MODELLING AND SIMULATION 2011

THE EUROPEAN SIMULATION

AND

MODELLING CONFERENCE

2011

ESM®'2011

EDITED BY

Paulo Novais
José Machado
Cesar Analide
António Abelha

OCTOBER 24-26, 2011

GUIMARÃES, PORTUGAL

A Publication of EUROSIS-ETI
The European Simulation and Modelling Conference 2011

GUIMARÃES, PORTUGAL

OCTOBER 24-26, 2011

Organised by
ETI - The European Technology Institute

Sponsored by
EUROSIS
Universidade do Minho
Co-Sponsored by
Ghent University
The University of Skovde
The Higher Technology Institute
and
BITE

Hosted by

Hotel de Guimarães
Guimarães, Portugal
EXECUTIVE EDITOR

PHILIPPE GERIL
(BELGIUM)

EDITORS

General Conference Chairs
Paulo Novais, Universidade do Minho, Campus de Gualtar, Braga, Portugal
José Machado, Universidade do Minho, Campus de Gualtar, Braga, Portugal

General Programme Chairs
Cesar Analide, Universidade do Minho, Campus de Gualtar, Braga, Portugal
António Abelha, Universidade do Minho, Campus de Gualtar, Braga, Portugal

Past Conference Chair
Prof. Gerrit Janssens, Hasselt University, Hasselt, Belgium

Journal Publication Chairs
Yan Luo, NIST, Gaithersburg, USA
Peter Lawrence, Swinburne University, Australia

Local Committee Chairs
Ângelo Costa, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Davide Carneiro, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Miguel Miranda, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Júlio Duarte, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Guilherme Pereira, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Luís Dias, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Gabriel Pontes, Universidade do Minho, Campus de Gualtar, Braga, Portugal

INTERNATIONAL PROGRAMME COMMITTEE

Methodology and Tools
Jon Andersson, Chalmers Univ. of Techn, Gothenburg, Sweden
Thomas Hanne, Fraunhofer -ITWM, Kaiserslautern, Germany
Bjorn Johansson, Chalmers Univ. of Techn, Gothenburg, Sweden
Erik Lindskog, Chalmers Univ. of Techn, Gothenburg, Sweden
Moreno Marzolla, University of Bologna, Bologna, Italy
Laszlo Muka, Budapest, Hungary
J. Manuel Feliz Teixeira, University of Porto, Porto, Portugal
Bert van Beek, Eindhoven University of Technology, The Netherlands

Simulation and Artificial Intelligence
Helder Coelho, Fac Ciencias, Lisbon, Portugal
Paulo Cortez, University of Minho, Guimareas, Portugal
Adam Galuszka, Silesian Technical University, Gliwice, Poland
Martin Hruby, Brno University of Technology, Brno, Czech Republic
Vladimir Janousek, Brno University of Technology, Brno, Czech Republic
Esko Juuso, University of Oulu, Finland
Wolfgang Kretzter, Univ. of Canterbury, Christchurch, New Zealand
Leon Rothkrantz, TU Delft, The Netherlands
Jim Torresen, University of Oslo, Norway
Morched Zeghal, Nat. Res.Council Canada, Ottawa, Canada
INTERNATIONAL PROGRAMME COMMITTEE

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

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

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

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

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

Cosmological Simulation
Philippe Geril, ETI Bvba, Ostend, Belgium

Discrete Simulation Modeling Techniques and Tools
Guilherme Pereira, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Luís Dias, Universidade do Minho, Campus de Gualtar, Braga, Portugal

Logistics and Simulation
Guilherme Pereira, Universidade do Minho, Braga, Portugal
Maria do Sameiro Carvalho, Universidade do Minho, Braga, Portugal

Simulation and Optimization
José António Oliveira, Universidade do Minho, Campus de Gualtar, Braga, Portugal
Sameiro Carvalho, Universidade do Minho, Campus de Gualtar, Braga, Portugal
INTERNATIONAL PROGRAMME COMMITTEE

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

Intelligent Systems
José Machado, Universidade do Minho, Braga, Portugal
Manuel Filipe Santos, Universidade do Minho, Guimarães, Portugal
Victor Alves, Universidade do Minho, Guimarães, Portugal

Real-time GPS Simulation and Service Applications
Matthias Klumpp, FOM University of Applied Sciences, Essen, Germany

Simulation with Petri Nets
Pascal Berruet, Universite Bretegne Sud, Lorient, France
Mauro Iacono, University of Naples II, Italy
Olivier Grunder, UTBM, Belfort, France
Stefano Marrone, Seconda Universita degli Studi di Napoli, Naples, Italy
Alexandre Nketsa, LAAS-CNRS, Toulouse, France
Mario Paludetto, LAAS-CNRS, Toulouse, France
Jean-Claude Pascal, LAAS-CNRS, Toulouse, France
Ivo Vondrak, Technical University of Ostrava, Czech Republic

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

DEVS
Fabrice Bernardi, University of Corsica, Corte, France
Dirk Brade, FOI, Stockholm, Sweden
Adriano Carvalho, FEUP, University of Porto, Portugal
Alexandre Muzy, Universite de Corse, Corte, France
Fernando Tricas, Universidad de Zaragoza, Spain

Fluid Flow Simulation
Diganta Bhusan Das, Loughborough University, United Kingdom
H.A.Nour Eldin, University of Wuppertal, Germany
Markus Fiedler, Blekinge Institute of Technology, Sweden
Hai Xiang Lin, TU Delft, The Netherlands
EUROPEAN SIMULATION AND MODELLING CONFERENCE 2011
Preface

Welcome to the 25th European Simulation and Modelling Conference (ESM 2011) organized by EUROsis. As in previous editions, this international conference presents the advances and promotes the research in the fields of simulation and modelling, allowing for the scientific exchange among researchers, practitioners, scientists and engineers. The conference is devoted to the core areas of simulation, including methodologies, models, languages, tools and computer programs, bringing to the event people from several areas, such as economy, management, health care, transportation, logistics, industry, environment, education and computer science, just to refer a few.

Welcome also to Guimarães and to Portugal. The city of Guimarães is located in the North of Portugal, in the Minho region, 350 km north of the capital, Lisbon, and about 50 km from the second Portuguese largest city, Oporto. Guimarães has its origin in times previous to the Foundation of the Portuguese nationality, and it is the place where Portugal was born in the twelfth century. It is proudly referred by its people as the Cradle of the Nation. The historical centre of Guimarães is classified as a UNESCO World Cultural Heritage. In 2012, Guimarães will be the European capital of Culture and more than 500 cultural events and 200 training sessions are expected to take place.

The Minho region borders on Galicia (Spain), in the north, and the Atlantic Ocean, in the west. Several attributes are associated to this beautiful region, such as the green of its natural scenery, the abundance of water, the mountainous terrain, the great biodiversity, the beaches along the coastline, villages with granite structures and the small fields, vineyards and corn fields. The Minho hospitality is worldwide recognized, as well as the rich culinary tradition and the green wine, the artistic expression with a religious mark and the popular festivals and celebrations.

The University of Minho is currently among the most prestigious institutions of higher education in the country, and it has also gradually come to assert itself on the international context. The University is renowned for the quality of its teaching, the quality of its students, the public recognition given to its Alumni and for its intervention and strong links with the local community and the surrounding region. Founded in 1973, the University has two major poles: the campus of Gualtar in Braga, and the campus of Azurém in Guimarães and it has a student population of 16,000, out of which 2,000 are postgraduate students.

The co-editors of the conference proceedings are members of the Computer Science and Technology Centre (CCTC), the University of Minho research unit in Informatics, which was associated to the event. We will like to highlight the work of the International Programme Committee, which contributes to maintain the technical excellence of the conference. We will like to thank the keynote speakers, the workshop and tutorial chairs, the presenters, the session chairs and the co-authors. Special thanks to the Hotel of Guimarães where the event will take place. The final thank is for Mr Philippe Geril. He accustomed us to an efficient and effective management and planning.

Paulo Novais, José Machado, Cesar Analide and António Abelha
(Conference and Program Chairs)
CONTENTS

Preface ........................................................................................................ IX
Scientific Programme .................................................................................. 1
Author Listing .............................................................................................. 445

DISCRETE SIMULATION METHODOLOGY

An Easy Extendable Modeling Framework for Discrete Event Simulation Models and their Visualization
Hendrik Renken, Christoph Laroque and Matthias Fischer ......................... 5

Regression Based Metamodelling Approach for Nonstationary Simulation Output Analysis
Rita Marques Brandão and Acácio M. O. Porta Nova ............................... 14

Comparison of Monte Carlo Methods through a Case Study
Megdouda Ourbih-Tari, Abdelnasser Dahmani and Latifa Ourbih-Baghdali .... 20

MULTI-AGENT SIMULATION

A Survey on Complexity Sciences and their Modeling Techniques
Morteza Mahmoudzadeh and Javad Jassbi ..................................................... 27

Multi-Hierachical-Agent Based Modelling Simulation Framework
Carlos Bragança de Oliveira and António Carvalho Brito ......................... 33

Cartography of Multi-Agent Model Parameter Space through a reactive Dicotomous Approach
Patrick Taillardt and Frédéric Amblard ......................................................... 38

SOFTWARE DEVELOPMENT TOOLS

Basic Constructions for the Definition of the Software Development Process with Formal Methods
Jan Kožusznik and Svatopluk Štolfa ............................................................. 45

System Dynamic Simulation Approach of Software Development Process in Haskell
Jan Kožusznik .................................................................................................. 48

Simulation as a decision Support Tool in Maintenance Float Systems – The Automatic Generation of Simulation Programs
Francisco Peito, Guilherme Pereira, Armando Leitão and Luis Dias .............. 53
CONTENTS

ECLIPSE-DDML: A Graphical Modelling Tool for Dynamic System Simulation
Ufuoma Bright Igboroje and Mamadou Kaba Traoré ..............................................61

DISTRIBUTED PARALLEL SIMULATION

HLA High Performance and Real-Time Simulation Studies with CERTI
Jean-Baptiste Chaudron, Martin Adelantado, Eric Noulard and Pierre Siron ....69

Warp-Level Parallelism: Enabling Multiple Replications In Parallel on GPU
Jonathan Passerat-Palmbach, Jonathan Caux, Pridi Siregar, Claude Mazel and David R.C. Hill.................................................................76

Simulation Based Evaluation of QOS Guided Scheduling for Distributed Computing in Large Environments
Florin Pop........................................................................................................84

A Generic Methodology to DEVS Parallel and Distributed Simulation
Adedoyin Adegoke and Mamadou K. Traoré ..........................................................91

BUSINESS SIMULATION

Storyboards Method and Structured Shot
David Ježek and Svatopluk Štolfa........................................................................99

Trust Model for Social Network
Arnostka Netrvalova and Jiri Safarik ...................................................................102

Competency Based Human Resources Allocation in Business Process Simulations
Štěpán Kuchař, Jan Martinovič and Tomáš Kocyan .............................................108

Market Segmentation through Conjoint Analysis using Latent Class Models
Liberato Camilleri and Lara Marie Azzopardi ....................................................112

Using Data Mining for Bank Direct Marketing: An Application of the Crisp DM Methodology
 Sérgio Moro, Raul M. S. Laureano and Paulo Cortez .......................................117

INDUSTRIAL DECISION MAKING

A Simulation Study of a Multi-Site Production Plant using ARENA
António Carvalho Brito, Carlos Bragança de Oliveira and Paulo Sá Marques ......................................................................................125
MDE in Action: First Steps towards a Crop Model Factory
Guillaume Barbier, François Pinet and David R.C.Hill ...........................................130

Use of the Belief Theory to formalize Agent Decision-Making Processes: Application to cropping Plan Decision-Making
Patrick Taillardier and Olivier Therond .................................................................138

INDUSTRIAL OPTIMIZATION

Investigating the Perfect Separation on Binary Outcomes coded in Decision Making Patterns for Pace Rating
Peeradaech Suwittayaruk and Dirk Van Goubergen .............................................145

Modeling, Simulation and Optimization in a Textile Production System
Sérgio Oliveira, José A. Oliveira, Luís S. Dias and Guilherme Pereira .....................150

A Simulation of Allocation of the Power Reserve in the Power System and Microgrid containing Renewables
Eugeniusz M. Sroczan ..........................................................................................155

ENGINEERING SIMULATION

Simulation and Experimental Techniques for the Analysis of localized Incremental Forging Operations
Ana M. Camacho, Marta M. Marín, Claudio Bernal and Miguel A Sebastián ....161

Real-Time Simulation of Automated Mechanical Transmissions
Marius Bataus, Daniela Vasiliiu and Nicolae Vasiliu .............................................168

ELECTRONICS SIMULATION

Modeling of the Advanced Spin Transfer Torque Memory: Macro and Micromagnetic Simulations
Alexander Makarov, Viktor Sverdlov, Dmitry Osintsev, Josef Weinbub and Siegfried Selberherr .................................................................177

A Simulation Study of Data Processing Activities in CMS Physics Analysis
Ciprian Dobre ...........................................................................................................182

NS-3 Simulation and Management of WiMAX and LTE Networks with NSDL
Eduardo M.D. Marques, Jorge J.F. Sousa and Paulo N.M. Sampaio .................190
CONTENTS

TRANSPORT SIMULATION

A Thirty-Two Thousand Variable Nonlinear Transportation Problem
William Conley .................................................................197

A Simulation Approach to Support the Design of Flexible Transport Systems
Ana Dias, José Telhada and Maria Sameiro Carvalho .........................201

Mathematical Model and Industrial Application of Four Index Transportation Problem
Michel Gourgand, Tuyet-Hoa Pham and Alain Tanguy ........................209

VEHICLE TRAFFIC SIMULATION

Autonomous Vehicles Simulation: A Comprehensive review
José L. F. Pereira and Rosaldo J. F. Rossetti ..............................................217

Utilisation of Computer Simulation for testing additional Support for Dispatching Rail Traffic
Jan Fikejz and Antonin Kavička ..........................................................225

GPS BASED SIMULATION

GPS Based Real-Time Transport Control for Production Network Scheduling Simulation
Matthias Klumpp and Christof Kandel ....................................................235

Shapley Value Simulation for Allocating GHG Emission Savings due to Logistics Pooling within ECR Cooperations
Christian Wick, Matthias Klumpp and Christof Kandel ........................240

LOGISTICS PROBLEMS SIMULATION

A Computer Assisted Study applied to the Analysis of the Production Set Efficiency
Magdalena Hajdasz and Adam Marlewski ..............................................247

An Empirical Evaluation of Martins’ Algorithm for the Multi-Objective Shortest Path Problem
Gerrit K. Janssens and Jose Maria Pangilinan ........................................252

An exact Algorithm for the Full Truckload Pick-Up and Delivery Problem with Time Windows: Concept and Implementation Details
Gerrit K. Janssens and Kris Braekers ..................................................257
Green Bullwhip Effect Simulation Concept
Matthias Klumpp .................................................................263

Goal-Based Agents in Material Flow Simulations –Integration of an Agent Programming Framework in the Discrete Event Simulator D³FACT
Alexander Klaas, Christoph Laroque, Hendrik Renken and Wilhelm Dangelmaier .................................................................266

WAREHOUSE SIMULATION

Simulation in the Warehouse Design and Management Context: A Survey
Carla A. S. Geraldes, Maria Sameiro Carvalho and Guilherme A B Pereira .....275

An Integrated Approach for Warehouse Design and Planning
Carla A. S. Geraldes, Maria Sameiro Carvalho and Guilherme A. B. Pereira ...281

Humanitarian Logistics Depot Location Model
Christof Kandel, Hella Abidi and Matthias Klumpp .....................................288

INVENTORY LOGISTICS

Workload Control Methods Stability
Nuno O. Fernandes and S. Carmo-Silva .................................................................297

Simulation as Tool to estimate the Homogenization Efficiency of Bulk Material Handling
Mark B. Duinkerken, Dingena L. Schott, Jos T. van der Geest, Paul Stoop and Gabriel Lodewijks .........................................................300

FLUID FLOW SIMULATION

CFD Modeling of the SDS2 Injection System for CANDU6 Nuclear Reactor
I. Prisecaru, S. Nicolici, D. Dupleac and I. P. Niţă ......................................................309

Hybrid modeling and simulation of a multi-product reactive distillation column
Ramon R.H. Schifflers, Edwin Zondervan, Jacobus E. Rooda and André B. de Haan ............................................................314

Modeling, Simulation and Dynamic Identification of the Electro Hydraulic Proportional Valve
Bogdan Mihalescu, Constantin Călinoiu and Nicolae Vasiliu .............................320
CONTENTS

Uni-axial Compressive Strength Prediction of Jet Grouting Columns using Support Vector Machines
Joaquim Tinoco, António Gomes Correia and Paulo Cortez.........................326

WASTEWATER SIMULATION

Comparative Modeling of Hydraulic Load of Water and Wastewater Networks
Jan Studzinski........................................................................................................333

Modeling and simulation of technological wastewater neutralization system in the brewing industry
Andrzej Urbaniak, Wojciech Góra and Rafał Brodziak........................................338

DISASTER MANAGEMENT

Momentum Probability and Staffing Calculations for Comet, Meteor and Asteroid Defense
William Conley ........................................................................................................343

Simulating Medical Disasters with Realistic Victim Profiles
Christophe Ullrich, Filip Van Utterbeeck, Erwin Dhondt and Michel Debacker..349

Climate Change Vulnerability Assessment with constrained Design of Experiments, using a Model Driven Engineering Approach
Romain Lardy, Gianni Bellocchi, Bruno Bachelet and David R.C. Hill ...............354

MEDICAL SIMULATION

Spatial Simulation of a (H1N1) Influenza Virus Spreading Comparison with Regional Data
Luc Touraille, Amandine Letagneaux, Damien Mouly, Alain Leroy, Delphine Bas, Julien Keruzec, Claude Mazel, Alexandre Muzy and David R.C. Hill.................365

Latest advances in the simulation of the removal of blood clots by a newly developed mechanical thrombectomy device
G. Romero, I. Moreno, M.L. Martinez, G. Pearce and N.D. Perkinson ...............373

Simulation Approach for Systems Radiation Biology
David Sarramia, Lydia Maigne, Vincent Breton, Laurent Deroussi and Benjamin Louvet .................................................................................................................381
MEDICAL DECISION SUPPORT SYSTEMS

A Moral Decision Support System in Medicine
Gabriel Pontes, Ana Duarte, David Cuevas, Maria Salazar, Miguel Miranda, António Abella and José Machado ..........................................................387

Digital Clinical Guidelines Modelling
Tiago Oliveira, Ângelo Costa, João Neves and Paulo Novais.........................392

AASYS - Appointment Alert System: An Open Source-Based Software to improve Show Rates in a Health Care Unit
Hugo Peixoto, José Machado, António Abelha, José Neves, António Correia and Manuel F.Santos.................................................................399

Performance Measuring of a Paediatric Operating Theatre: A Discrete Event Simulation Approach
Latifa Dekhici, Alain Tanguy and Khaled Belkadi.........................................404

Object trajectory simulation - An evolutionary approach
Pedro Goncalves, Luciano Alves, Tiago Sá, Cesar Quintas, Miguel Miranda, António Abelha and José Machado ........................................409

Towards Intelligent Drug Electronic Prescription
Júlio Duarte, João Neves, Alexandra Cabral, Marco Gomes, Vitor Marques, Manuel Filipe Santos, António Abelha and José Machado .................414

Enabling Real-time Intelligent Decision Support in Intensive Care
Filipe Portela, Manuel Filipe Santos, Pedro Gago, Álvaro Silva, Fernando Rua, António Abelha, José Machado and José Neves...............419

LATE PAPERS

ActivityDiagram2PetriNet: Transformation-Based Model in Accordance with the OMG SYML Specifications
Damien Foures, Vincent Albert and Jean-Claude Pascal ..........................429

An Objective Driven Framework for Discrete Event Simulation of Continuous Systems
V.Albert and A.Nketsa........................................................................434
SCIENTIFIC PROGRAMME
DISCRETE SIMULATION METHODOLOGY
AN EASY-EXTENDABLE MODELING FRAMEWORK FOR DISCRETE EVENT SIMULATION MODELS AND THEIR VISUALIZATION

Hendrik Renken
Christoph Laroque
Business Computing, especially CIM
Heinz Nixdorf Institute
University of Paderborn
Fürstenallee 11
D-33102 Paderborn, GERMANY
E-mail: {funsheep|laro}@hni.upb.de

Matthias Fischer
Algorithms and Complexity
Heinz Nixdorf Institute and
Department of Computer Science
University of Paderborn
Fürstenallee 11
D-33102 Paderborn, GERMANY
E-mail: mafi@upb.de

KEYWORDS
Simulation interfaces, Object-oriented, Discrete simulation, Combined simulation, Man-machine interfaces

Abstract

Today’s simulation software covers a variety of different problem areas, e.g., production, transportation, logistics, and physics. Therefore the manufacturers of simulation software use a generic simulation engine equipped with problem specific libraries to offer specialized products. These products are supplemented by user-programmed models and building blocks. However, the combination of components from different problem areas often lead to software architectural problems in terms of extensibility and reusability. Especially for the simulation area, we offer an architectural concept that provides the ability to simply combine components from different problem areas and the ability to make future extensions. Classic user interfaces are unable to handle this flexibility in simulation modeling. Therefore, we offer an architectural concept connecting simulation data with its visualization. We implemented our concepts in a prototype system and demonstrate them on the basis of the two problem areas “Production” and “Logistics”.

INTRODUCTION

Simulating a system to understand its behavior for certain inputs is a well established scientific method and is broadly used in research as well as in the industry. Therefore today’s commercial simulation software has to handle a wide range of problem areas, e.g., “production”, “logistics”, “transportation”, and “physics”, etc. Supporting all these areas in one software product bores the entire product with functions, building block libraries and tools. To help a user to keep track of his model and the problem he wants to solve, the simulation software is often adapted to each problem area.

Typically, the simulation core remains the same, but the building block library for creating the simulation model and the user interface, shipped with the software, suite a specific problem area. While the generic simulation core helps the software manufacturer to adapt his software to new problem areas, the high integration of the specialized building block library and graphical interface eases the modeling for the user. This makes each adaptation of its own software product, e.g., see Enterprise Dynamics (Solutions 2011b). The amount of specialized products and libraries grows with the adaption to more and more different problem areas. Also some manufacturers offer to modify the source code of their building blocks, enabling users to develop their own individual library versions.

This diversity leads to the following two problems: On one hand, the development of several variants of a building block, e.g. by different teams of a company, typically leads to incompatible building block variants and thus also to incompatible module libraries. On the other hand, problems are caused by updates of the manufacturer’s building block library. It is difficult to adjust the user’s variant with the updated building blocks of the manufacturer. The reason for these problems is the static inheritance hierarchy with which building block libraries are typically designed. While being backed by excellent build-in support from high-level programming languages such as C++ and Java, static type hierarchies often suffer from well known problems such as the “bubble up” effect or the “deadly diamond” (see Gregory (2009)). These problems make the extension and coupling of different building block libraries (based on the same simulation engine) difficult.

In this paper we introduce a model architecture that is based on property composition instead of static type hierarchies. This modeling technique provides an easy way to combine building blocks from different problem areas or to extend a model with components from external suppliers.

Another software part, specialized in a specific problem area is the user interface or visualization. It provides
tools and views highly depending on the problem area. E.g., the visualization of a manufacturing plant in 3D space is different from the visualization of physical or logistical data. Combining different problem areas in one simulation model leads to several visualizations capable of interacting with parts of the model. Extending the approach introduced for the combination of simulation models to visualizations, we are able to easily create a model describing the simulated data. Separating the simulation data from the visualization data can provide a flexible and extendable user interface.

In the next section we introduce our concepts. After that we provide an overview over related work. In Chapter “Model Object Architecture Based on Composition” details about our model architecture are given and the visualization model is presented afterwards in the Chapter “Visualizing the Simulation”. We implemented our concept in our prototype “d³fact” (Renken 2011) and show details and an evaluation of this system in Chapter “Evaluation”. We conclude our paper with a forward look on future steps.

CONCEPTS

Now we briefly discuss details of the problem and introduce our approach to overcome the problems afterwards. For this, we use the example shown in Figure 1. It is based on a static inheritance type hierarchy. The example shows two different products for two problem areas, namely “Inter-Company Logistics” and “Production”. Every problem area and building block library has its own base type directly derived from a common type Entity. The base type contains information (and functions) generally available throughout objects of the particular problem domain. This type belongs to the simulation core and provides an easy access to it, e.g. to access an event-scheduler.

Assume you want to optimize your production system, in terms of cost efficiency, and therefore you would like to create a simulation model of your production facilities. To optimize the whole system you would also want to model the “Inter-Company Logistics” to simulate the dependencies between the facilities and to estimate the impact of your local changes on a global scale. This raises the problem that the production part of your model only works with elements from the “Production” problem area, while the logistics can only be modeled using objects from the “Logistics” problem area. They won’t work out of the box with the production’s building blocks. A typical solution to this problem is to create a new “bridge” type, that can exchange information between building blocks of the different areas. This process is error-prone, time consuming and not suitable for the average user.

Figure 1: Two simulation products based on the same core. Users often create their own variants by manipulating the building block libraries of the products.

Model Object Architecture

We propose a property-centric approach that enables us to create new types by combining properties from different problem areas. E.g. to solve the previously described problem we just combine a “Freight”-property from the logistics area with a “Token”-property from the production area in a new object type. The combination of properties works without any additional lines of code. While “Production” and “Logistics” are areas that are somehow related to each other, this concept also works for completely unrelated areas: E.g. you can add some physical properties to your production tokens, so that they gain weight and a centre of gravity with which the simulation can be made much more realistic.

Another problem is the steady change of requirements in complex projects: We can design software to handle the upcoming requirements for the next years, but at some point the requirements will change so much, that most software needs to be completely rewritten. Let’s assume you already have a simulation model that was successfully used to plan a new production facility. Now your
supervisor suggests to simulate a variant of your model using machines from another manufacturer. Luckily this manufacturer provides you with a library where he already implemented building blocks representing his machines, but because you had to modify the source code of the standard building blocks, the provided library won’t work within your model without modification. Eventually the manufacturer provides you with the source code, then you still have to understand and correctly modify the external code, which is a very time consuming and difficult task.

Therefore we use the aggregation and composition pattern to build our building block library. With these two patterns we can simplify or completely remove the type hierarchy making the building block library easy to extend and the code highly reusable (see Chapter “Model Object Architecture Based on Composition”).

Visualization Models

Gaining such flexibility within the building block library can only be used in its full potential when the graphical interface passes this flexibility on to the user. As stated before, in today’s simulation software the building block library and the user interface are well integrated as well as deeply rooted in the implemented code, making it hard for the graphical interface to offer this flexibility. Consider the extension of a building block library with new components from a completely new problem area. Then you would also have to extend the software with new modeling tools, visualization methodologies and analysis capabilities to support the user in solving his problem in the new problem area. This is often impossible because of monolithic architectures.

We overcome this problem by applying a slightly modified “Presentation Model” (PM) pattern (Fowler 2011) to our simulation software, resulting in a split of the simulation part of the building block library from its visualization representation (cp. Figure 2).

This two-staged architecture allows us to use several visualizations with one simulation model, making it easy to view specific parts of the simulation model using different approaches. Also this enables us not only to provide different visualization representations for the simulation data (e.g. 2D, 3D, diagrams, etc.), but also provides different editing views for different problem areas, without the rebuild of the simulation software or model. For our example, extending our production facility model with “Inter-Company Logistics” (see Figure 1), we create two different views: one view for the production part of our model, giving a good overview over the interior of the facility and a second view to manage and view the “Inter-Company Logistics”. This decoupling provides the following advantages: First, the user is able to utilize the different visualization representations and to edit specific parts of the model in a convenient way. Second, several users can work together on the same model: E.g. while you are using a “Production” related view to create or modify the model part representing the factory, your colleague uses a different view to integrate the “Inter-Company Logistics” into the same model. Third, it is possible to create different views for different hardware platforms: We can create a minimalistic (2D) view for a browser based platform and a complex 3D view for a workstation.

To show the advantages of our proposed model architecture and the usage of the PM pattern in simulation software, we implemented both concepts in a prototype which will be outlined at the end of the paper.

RELATED WORK

Current Simulation Software

Today’s simulation software packages offer very powerful solutions for the modeling, simulation and analysis of systems. Often coupled with a graphical user interface and a dynamic process animation, the user is allowed to model and analyze via drag & drop (Siemens 2011, Rockwell Automation 2011, Company 2011). Most simulation packages offer more or less domain-specific building block libraries and aim at a single problem domain and its experimentation. In some cases multiple problem areas can be integrated through additional building block libraries. We discuss the general approach of most professional and commercial packages using the example of the simulation software suite “Enterprise Dynamics”, which is built by INCONTROL (Solutions 2011a). Enterprise Dynamics uses the concept of atoms. An atom is primarily an unspecified object with certain properties, which can react on events occurring during a simulation run. Each simulation model, as well as every object within this model is formally an atom. Based on this concept, INCONTROL offers several simulation suites that handle various problem areas, e.g. “Logistics”, “Production”, “Pedestrian Simulation”, “Warehousing” (for a detailed overview see (Solutions 2011b)). In each suite, there are pre-modeled atoms, which have some specific properties, ordered in a hierarchical manner, but using a flat hierarchy of objects. Transferring it to the example from Figure 1 would create a hierarchy like the one shown in Figure 3.

The user can manually implement additional aspects, which allow for some additional flexibility in building problem-specific solutions. But this approach might
Figure 3: The static type hierarchy of the example from Figure 1 build with the *atom* concept.

also lead to modeling problems or non-optimal solutions, since the user is limited to the basic building blocks and scripting language of the tool. The application of a specific tool might even limit future applications and reuse of the model components in other projects, since the user cannot integrate new demands and problem domains into the tool-framework in retrospect. A major advantage of the commercial tools for understanding and communication purposes is their ability to visualize specific simulation runs as well as the simulation results. But this is also a limitation. Even if the building blocks of the existing tool libraries offer possibilities for exploitation, the visualization possibilities are fixed and limited within the tool. Here, the user has no possibility to adjust the visualization for specific demands, e.g. for a specific group of applicants.

More flexible simulation language frameworks, e.g. Simula or GPSS (Nygaard and Dahl 1978, Ingolf Stähl 2011), offer a wider flexibility, since the model libraries are defined on a less detailed level and are in most cases not bound to a specific visualization method. Since the programmer is aware of the entire programming interface (API), he would be able to implement a visualization by himself. Nevertheless, those frameworks are in most cases based on a specific hierarchical library approach for the model’s elements.

There have been some efforts to automatically generate simulation code from abstract model descriptions (Filho and Hirata 2004, Dias et al. 2006). These approaches focus on the processes within the model, while our concept deals with the data representation.

Runtime Object Architectures

There are several object architectures that can be used to create and maintain a building block hierarchy. Some of them are widely known and available as standard programming patterns, while others only apply in specific situations and therefore are mostly unknown. In this chapter we discuss the different approaches, their advantages as well as their drawbacks.

Static Type Hierarchies Static inheritance type hierarchies are simple to understand since it is natural for us humans to arrange objects in a taxonomy (Sommerville 2004, Shaw and Garlan 1996). However, these type hierarchies can become hard to maintain because of different side effects, discussed below. This approach is based on the idea to have abstract types that represent certain kinds of objects, containing common functions and information for that specific kind. Derived child types inherit these functions and information, making it easy to create new types.

However, using this software pattern has some drawbacks, especially when the hierarchy should be extended in a yet unknown way - as it is common in simulation software (Shaw and Garlan 1996). Consider the small Example-Hierarchy from Figure 3.

Suppose we want to add a “3D position” to the two types *Freight* and *Token*. We’d have two options: We could add this function to both types individually or add the function to the lowest shared parent in the hierarchy (here *atom*). The first approach has the drawback of the function-enabling code having to be copied into both types. This example is too small to become problematic. However, in a continuously changing and growing hierarchy the copying of code would be performed a lot more often. It would be hard to keep track of the copied code parts and therefore it would get very difficult to maintain the whole structure. The second approach unwillingly introduces new functionalities into types that never should have had them (here *Truck* and *Machine*). This problem is commonly called the bubble up effect and (often) solved using a boolean flag, indicating whether or not a specific type has a specific functionality (Gregory 2009). Through this process, the higher, more abstract types can gain a lot of functionality over time, making the code hard to maintain and difficult to extend.

Multiple Inheritance Some high level programming languages allow multiple inheritance. C++ is a prominent representative of such languages. Colloquial called the deadly diamond. This concept is very error-prone because at some point, especially in bigger hierarchies, the overview on which type implements which variables and methods, is lost. A lot of problems arise when two different types implement variables or methods with the same name but different meanings. Normally this concept is avoided in all larger projects (Gregory 2009).

Mix-In Classes Another approach to make a static type hierarchy more flexible are Mix-In classes or simply Mixins (Smaragdakis and Batory 2001, Gregory 2009). In this approach we do allow multiple inheritance but with one restriction: “An object type can have multiple parents, but only one grandparent”. A type does never inherit two classes which have the same grandparent. This enables us to inject new functionalities through additional types into a monolithic type hierar-
chy. However, multiple inheritance is still not easy to handle and some languages (e.g. Java) do not support this approach. We are unable to derive from Token and Freight to create a new type that works in both problem areas.

Composition and Aggregation With the two patterns “composition” and “aggregation” a type hierarchy can be simplified or even completely removed (Gregory 2009, Deacon 2005). Here we use the definition partially introduced with UML. While a hierarchy is based on the concept of “is a”, a composition on the other hand on “has a”.

Composition This is the “strong” ownership. A composed object is bound to the life cycle of the owning object. Lets assume we have a building with a room in it. The building owns the room. The room cannot exist without the building and when the building is destroyed, so is the room.

Aggregation This is often called “the lesser composition”. Aggregation does not imply an ownership. Again, we have the building from earlier with the room and now the room contains a table. If the building and the room are destroyed, the table may exist afterwards. Composition and aggregation at first is a formal concept. Every simple variable in a type can be declared to be “composed”. The real benefit of this concept is exposed when it is used to combine different subsystems into one object, while these subsystems were never meant to be combined.

“Model View Controller” Software Pattern

The “Model View Controller” or MVC paradigm was initially developed for user interfaces written in Smalltalk-80. It describes a decoupling of the user interface from the data that is shown in the interface (Krasner and Pope 1988, Sommerville 2004, Deacon 2005). Three different roles are described: The model is the central structure that contains the data of an application (or simulation). The view displays the data to the user. Several views can exist for one model, each displaying the data in a different way. A controller is used to convey between a view and a model. It receives user input from the view (e.g. a mouse click), interprets the input and changes the model in an appropriate way. The decoupling of the data and its presentation makes a software way more maintainable and structured.

MODEL OBJECT ARCHITECTURE BASED ON COMPOSITION

Todays commercial simulation software uses an object-oriented, on a static inheritance type hierarchy based concept as their model architecture. As stated in the Introduction and in the Chapter Concepts these hierarchies often become problematic in terms of reusability. Therefore we use the property centric approach called “Composition and Aggregation”, introduced earlier. Using this approach an object in a simulation model is no longer an instance of a static type. Instead the object is represented by a dynamic set of “properties”. Here a “property” is defined as a [key, value]-pair, where the key is a unique identifier for the value. You can think of the key as a variable-name which is used to access the property value. While there are very different approaches to identify a set of properties, we use a generic container concept to sustain the object-oriented approach common in current simulation model architectures. The container type provides methods to manage its properties (add, delete, get by key, etc.). Consider the following example of the “Machine” type from Figure 1.

![Diagram](image)

(a) Standard Object Type. (b) Container Concept.

Figure 4: This Figure shows the differences between a static type (left) and our container concept (right). The properties are indicated by rounded boxes.

In Figure 4a it is displayed as when implemented as a normal static type. This implementation can be found in almost every high level programming language. Variables, indicated by their name and methods to trigger the logic/behavior are combined in a type. In Figure 4b however, our property-based approach is used to implement the type. All variables become properties, accessible through their name. To preserve the behavior of the type, the logic must be added as a property as well. As the term composition and aggregation indicates, there are two different ownerships defined. Roughly they apply to two different types of property, namely whether a property is active or passive in terms of the simulation. Simple properties like numerical values or strings are passive, meaning they do not react on state changes and also do not cause them. These properties are aggregated (the weak ownership): The container object owns them but they are not bound to the lifecycle of the container object. In the example above everything but the “logic” property is passive. The “logic”
instead, is an active property, because it does react on state changes. E.g., when a token arrives at the input for processing, the “logic” initiates the processing of the token by the machine. Composition strongly binds a property to a container, which means the property is bound to the container’s lifecycle and also receives a reference to it. Through this reference the property can access the container, the simulation core, the model and other objects within the model.

In Chapter “Concepts” we described an example where we want to use tokens as freight and vice versa. In a static type hierarchy, as indicated in Figure 5a, we would have to derive a new type FreightToken from Freight and Token.

(a) Extending the static hierarchy is difficult.

(b) Our concept is very flexible. Creating new types is very easy.

Figure 5: Extending a static type hierarchy can be difficult, but is easy when using the “composition” concept.

Additionally we would have to change the source code of all building blocks to use the newly created type. Using our approach however, we can easily add the token- and freight-property to a container. Now we have one object representing both, a freight in the logistics library and a token in the production library. One big benefit from our approach is, that creating such a new type can be achieved without coding a single line of code.

**VISUALIZING THE SIMULATION**

In the previous chapter we described a flexible simulation model architecture based on composing objects from [key, value]-properties. Passing this flexibility on to the user through a static user interface, based on the MVC pattern is nearly impossible. The interface would be bloated with functions and generic information about simulation objects since definite object types are not available and therefore it is not defined how an object should be displayed. Static user interfaces suffer from another problem: Because of the new flexibility you would want to add specialized modeling tools for newly composed object types which can get very difficult or is even impossible.

To provide a user friendly experience within our user interface we must overcome the restrictions introduced by static user interfaces and the MVC pattern. Our approach uses a two-staged concept to provide the same flexibility for the visualization as for the simulation model. It is based on the Presentation Model concept, that originated from Martin Fowler (2011). Instead of using the Presentation Model as a data layer for the view component, making it hard to extend the visualization with new types, we use the Presentation Model in a more general way. This decouples the components even more, allowing us to introduce new visualization concepts without modifying the view components. We call this decoupled Presentation Model simply Visualization Model. To display the data the user interface must create a drawable data structure from the visualization model (e.g., a scene graph). Figure 6a shows the normal MVC pattern and Figure 6b our two-staged concept. The characters in the boxes describe the role of the different components.

(a) Static user interfaces.  (b) Our concept.

Figure 6: This Figure shows the decoupled software parts and their role according to the MVC pattern. The arrows indicate the information flow.

Using static user interfaces in simulation software, the View interprets the data of the simulation model and decides how they should be displayed by accessing specialized variables or methods. Because of this static referencing between the simulation model and the view, it is almost impossible to provide different perspectives for a simulation model at runtime. Because the Controller is located in the visualization model we can easily add new modeling tools to support newly composed object types in the simulation model. We defined three principles for visualization models, enabling us to load several visualization models at runtime:

1. The simulation model does only contain data relevant for the simulation result.

2. The simulation model does not know how to display its state and certainly has no reference to a visualization model.
3. The visualization model has no impact on the simulation results. The simulation results are always the same, with and without a visualization model.

As already discussed earlier this can be useful to provide the user with specialized views on the simulation data, enabling a faster understanding of the processes in the model. Also it can be used to react on hardware restrictions like the connection bandwidth or the device used: E.g. a simpler (2D) visualization model can be loaded on a device like a smartphone and a more complex (3D) view provides a satisfying experience on a workstation computer. Figure 7 illustrates this concept.

![Figure 7: Example of an overlapping between a simulation model (solid line) and three visualization models (dotted lines).](image)

It shows a simulation model and several visualization models, where every visualization model displays different pieces of information from the simulation model. Some models may overlap or display additional information, not found in the simulation model, e.g. decorative content like a tree outside a simulated factory.

**EVALUATION**

To show the possibilities of the discussed concepts we implemented them in a prototype called “d³fact” (Renken 2011). In this chapter we want to present the prototype, show the currently existing examples and some performance measurements we did for a better evaluation of the concepts.

d³facts is written in Java with a client-server architecture to fully utilize the potential of the visualization models (see Figure 8). Currently two clients are implemented: A simple Java Client, capable of displaying 2D and 3D graphics, and a powerful C++ Client which can handle huge CAD model based 3D scenes (Eikel et al. 2011).

The latter recently gained low end hardware (netbooks and smartphones) support (Eikel et al. 2010) which we support through our visualization models.

Through some student theses we already gained support for the following problem areas: “Warehousing”, “Production”, “Agent-based Simulation”, “Transportation” and “Rigid Body Physics”. Currently we have models that combine “Warehousing” and “Production” and “Transportation” and “Production”. For the other problem areas there currently exist only small example models. Some are shown in Figure 10a, 10b and 11.

![Figure 8: Architecture overview of the prototype. The simulation server can provide several visualization models to a simulation model, helping to adapt the visualization to different client-concepts.](image)

Example: Combining Production and Logistics

In the following example, shown in Figure 9 we combined a “production” part and a “logistics” part into one model. We extended the scenario from (Fischer et al. 2010) with a “Inter-Company Logistics” part. It enables the truck to transport its freight between several production plants.

![Figure 9: Example where a fork lift from the “production” part of the model transports a package into a truck, located in the “logistics” part.](image)

Example: Visualization Model

As described earlier the visualization models can be used to display the simulation data in different ways. To illustrate this we implemented two different views, shown in Figure 10a and 10b. The simulation model uses physics to simulate the collision of at least 200 balls, every ball with different parameters. A ball is modeled as a container, having properties like “weight”, “friction”, etc. The visualization model from Figure 10a contains for each ball a visualization node having a sphere description with a random color, resulting in a natural view on the simulation data. An extension to this view is shown in Figure 10b where this visualization model records the trajectory of every ball and displays them as lines.

Another advantage is the possibility to load different visualizations according to the hardware used to display the information. Because of the third principle from Chapter “Visualizing the Simulation”, the different visualization models and therefore the different hardware has no impact on the simulation. In Figure 11 we show a simple 2D view of the simulation data already used for Figure 10. This visualization model can be used with simpler hardware like a smaller network connection or a smartphone, because this view needs lesser data to be
transferred and displayed.

Figure 11: A screenshot sequence of a 2D view of the simulation data already used in Figure 10.

The biggest advantage in terms of flexibility, availability and reusability is, that we could have made all these extensions and changes to the original models years after creating them, and most importantly, without much work.

Performance

We did some performance benchmarks on a reasonable up-to-date computer (A notebook with a 2.4 GHz Intel Core 2 Duo processor and 4 GB of RAM.) to test how much, in terms of simulation speed and memory consumption, the flexibility costs. The first benchmark (Figure 12a) shows the impact of our container approach on the overall simulation time of a simple production model where a large amount of container objects is created. The second benchmark, shown in Figure 12b, provides a detailed view on the resources needed to create a defined amount of container objects in a model.

The model of the first benchmark consists of a source and several machines, modeling a process within a saw mill. It was modeled with a time resolution of seconds and implemented using both, the “static type” approach and our container concept. We simulated an hour, a day, a week and a year while we measured the real time needed to complete the simulation. The benchmark shows that for this model we are about a constant factor of two slower than the static type approach.

During the second benchmark we simply recorded the amount of resources needed to create a specific amount of model objects. The results show that our initial container implementation, labeled ModelContainer, consumes a lot of memory and eventually causes the Java Garbage Collector to collect freed memory. This process may have a huge impact on the runtime and occurs when a lot of simple container objects, e.g. the “tokens” in a production model have to be created. This problem led to the creation of the RuntimeContainer, an implementation of the container concept, especially designed for simple, passive objects like “tokens”. In this implementation the reference to the model was removed, making the internal data management easier and therefore consuming less time and memory (As shown in Figure 12b). As result of the removal of the model reference, the RuntimeContainer can only be used for passive properties and therefore only for the modeling of simple container types.

CONCLUSION

In this paper, we report on our prototype “d³fact” of a discrete, event based simulation system, which uses the architectural concept of composition and aggregation.
This provides the user with the ability to easily combine different model components from different problem areas and allows extensions to simulation models, which might not be known at the projects start. The prototype system is demonstrated by combining a “Production” simulation case with some “Inter-Company Logistiques” simulation additions. In order to give the user the best possible chances to use the strengths of this gain of flexibility with our concept, we use a software design, similar to the Presentation Model software pattern for the visualization integration. Thereby, we are able to decouple the model description and behavioral description of a model component from its visualization. We offer some standard visualization patterns, but with this approach, the user may define a use-case-specific or even multiple visualization models for one simulation model and is thereby able to create visualizations for different hardware settings (smartphone, notebook, powerwall, etc.) as well as different communication targets (management presentation, engineers discussion, detailed modeling by the user himself, etc.).

In a next step, we are currently working on the development and implementation of a graphical user interface for the modeling of these simulation models as well as visualization models, in order to enhance the speed of modeling by the simulation expert.

References


REGRESSION-BASED METAMODELING APPROACH FOR NONSTATIONARY SIMULATION OUTPUT ANALYSIS

Rita Marques Brandão
Departamento de Matemática
Universidade dos Açores and CEG-IST
Ponta Delgada, Portugal
email: rita@uac.pt

Acácio M. O. Porta Nova
Departamento de Engenharia e Gestão
Instituto Superior Técnico
Lisboa, Portugal
e-mail: apnova@ist.utl.pt

KEYWORDS
Nonstationary simulation, output analysis, regression metamodels, time series models.

ABSTRACT

In this article, we present our approach for analyzing the output of a wide variety of discrete-event nonstationary simulations based on regression metamodels. We use classical regression techniques to uncover the deterministic components of the metamodels, while time series models are used for the residuals. Our approach is illustrated using a set of case studies taken from the simulation literature. Some conclusions and suggestions for further work are stated.

INTRODUCTION

In most simulation studies, the output analysis is focused on stationary situations. In these cases, we assume that the selected performance measures, or responses, possess steady-state distributions, with well defined expected values, and we perform point and confidence interval estimation of those responses of the target system. This is a very active research area and different procedures have been developed for that purpose. On the other hand, the behavior of real-systems in many areas, like traffic engineering, banking, inventory management and manufacturing, has to be more realistically considered as nonstationary, and the corresponding output analysis is basically a natural extension of the stationary case. Thus, we can only hope that those expected values will follow some smooth, predictable evolution in time. In these cases, the best that we can do is to try to explain this evolution, by fitting a simple expression, or metamodel, to the output produced by the simulation model. Our contention is that the expected value of the system response of interest will have a fundamentally deterministic, slow-varying behavior, that can be explained by an underlying metamodel. Superimposed on this, there will be highly autocorrelated high-frequency stochastic perturbations, that hide that underlying behavior, and that we will try to filter out in our approach. We will use classical regression techniques to uncover the deterministic component of the metamodel and we rely on classical (Box and Jenkins) time series models to express the random perturbations, or noise. When the responses being studied present some kind of periodic behavior, Fourier analysis/series may be required to obtain an adequate metamodel.

In our previous research work, we had already studied continuous-time responses in the context of nonstationary simulations. However, we mainly followed a mechanistic perspective, assuming autocorrelated responses, fitting Box-Jenkins models and relying on the corresponding forecast function for building the simulation metamodels; see Brandão and Porta Nova (2009). The results that were then obtained were certainly very interesting and motivating, especially because not too much material is available about nonstationary simulation output analysis. But we also identified some drawbacks and limitations of the approach. On one hand, discrete-time responses could not be addressed, and only continuous-time responses that were made stationary by successive differentiation could be handled by the proposed procedure.

Regression analysis is a widely used causal approach for metamodeling and it was our obvious choice for overcoming the limitations of the previous time series based approach. We first used it to stationaryize the time series of a non-terminating cyclical response, by removing the trend component, illustrating that approach on a known traffic example; see Brandão and Porta Nova (2004). Later, we also used regression metamodels in the development of a procedure for coping with the initial bias in stationary simulations, with very promising results in two case studies, one of them using analytical results; see Brandão and Porta Nova (2010). In this article, we complement our previous work, showing that regression metamodels can be used in a wide variety of nonstationary situations, to describe the behavior of both continuous time and discrete-time simulation responses.

This article is organized as follows. First, we present our methodology for analyzing the output of general nonstationary discrete-event simulations, based on the use of linear and nonlinear regression metamodels. Then, we discuss an experimentation of the procedure that was
performed on a selected set of case studies taken from the simulation literature. We conclude with a summary and some recommendations for further work in this area.

**METHODOLOGY**

Since our main interest is to analyze the output produced by nonstationary or transient simulations, the metamodels must be able to capture the evolution in time (or in terms of the arriving customer’s index) of the expected value of the system responses. Since the output produced by discrete event simulation models is highly autocorrelated, the random error patterns, on the majority of cases, are also usually autocorrelated. Therefore, in this work, we will consider metamodels represented by

$$Y_t = f(X; \beta) + u_t,$$

where $t$ represents the continuous time index (replaced by the arriving customer’s index, $j$, for discrete-time responses), $f$ is an unknown function, $X = (X_1, \ldots, X_m)^T$ is the vector of explanatory variables (eventually including time), $\beta = (\beta_1, \ldots, \beta_p)^T$ is a vector of unknown metamodel parameters (to be estimated), and $u_t$ represents the stochastic component of the metamodel (that may be described by a time series model). Although we are especially interested in estimating the metamodel deterministic component $f(X; \beta)$, a proper identification and characterization of the stochastic error component, $u_t$, is quite important for validating the estimated metamodel.

The general approach that we propose for analyzing nonstationary simulation responses consists of the following steps.

1. **Construction of an averaged time-series.** There is a wide range of situations that can be analyzed: unbounded or cyclical responses, or the transient behavior of stationary responses. Sometimes, the simulation duration is pre-determined, other times it needs to be decided after analyzing a long pilot run. With respect to the number of independent replications, we have found out that 30 runs was a good choice; see Brandão and Porta Nova (2009).

   For discrete-time responses (for instance, system or queue waiting times) observations are simply collected according to the customer’s arrival order. On the other hand, if the response is time-persistent (number of entities in the system, or waiting for service) the observations have to be sampled at a convenient discrete-time interval, $\Delta_t$, that has to be tentatively selected. Either way, the corresponding observations across runs are averaged, and it is this averaged time series that is used in the remaining steps of our procedure.

   2. **Identification, estimation, and validation of the metamodels.** The selection of the type of function to try for the underlying metamodel is done by looking at the representation of the averaged time-series with respect to time (or the arrival index). In order to facilitate the identification of an adequate relationship, until not long ago we had to rely on a catalog of different functional relationships, with their graphical representations. Today, there is software to assess the analyst in this task, like the one that we tried for this work: LAB Fit (http://zeus.df.ufcg.edu.br/labfit/).

   The metamodel parameters are then estimated in three steps: (i) assuming that the residuals are uncorrelated, preliminary estimates are obtained using the function gls (or gnlm if the model is nonlinear) of the package nlme for the the R Language—see R Development Core Team (2009); (ii) candidate ARMA models for the metamodel residuals are then identified in an automated way, using the function auto.arima of the forecast package—this function uses Akaike’s information criterion (AIC) to choose the best ARMA(p,q) model up to predefined orders $p$ and $q$—see Akaike (1974); (iii) the preliminary metamodel is updated using the function update of the package nlme, incorporating the autocorrelation structure of the regression residuals. This new metamodel assumes that the residuals are independent and identically distributed (i.i.d.) with zero mean and finite variance.

   The diagnostic checking of the metamodel is done by testing the significance of the coefficients ($t$ test), testing the normality of the residuals (Shapiro-Wilk test), and/or testing the hypotheses of uncorrelated residuals (Ljung-Box test).

   If the metamodel fails the validation process, this step is repeated. Otherwise, the metamodel can be used for evaluating the system under analysis and answer many “what-if” questions, without further simulation.

   3. **Confidence interval estimation.** When generalizing the output analysis for the case of nonstationary simulations, this is a topic that arises naturally, since a fitted simulation metamodel for a given response is simply the extension of the point estimator of this response in the stationary case. Thus, in this work, taking advantage of the abundant data available, we present alternative ways for constructing confidence intervals for specific values of the system responses throughout time (or the arrival index). First, the most obvious one, due to the availability of several independent replications, is based on the application of the method of independent replications to the values observed in each instant (or index value). However, we also looked
for an easier and more elegant way for building confidence intervals for the responses, based on a new metamodel fitted to the series of the variances that were calculated, in each instant of time, using the sample of 30 independent replications.

Confidence interval based on the 30 independent replications:

\[ \bar{Y}_t \pm t_{r-1,1-\alpha/2} \hat{\sigma}_{Y_t}/\sqrt{r}, \]

where \( \bar{Y}_t = r^{-1} \sum_{i=1}^{r} Y_{i,t} \), \( Y_{i,t} \) is the observation at time (or index) \( t \) in replication \( i \), \( t_{r-1,1-\alpha/2} \) denotes the \( 1 - \alpha/2 \) quantile of the Student’s \( t \)-distribution with \( r - 1 \) degrees of freedom, \( \hat{\sigma}_{Y_t} = \left[ (r-1)^{-1} \sum_{i=1}^{r} (Y_{i,t} - \bar{Y}_t)^2 \right]^{1/2} \) and \( r \) is the number of independent replications.

Confidence interval resulting from the estimation of the two metamodels (averaged and variance series):

\[ \bar{Y}_t \pm z_{1-\alpha/2} \sqrt{\frac{\hat{V}_t}{n}}, \]

where \( \bar{Y}_t = f(X_1; \hat{\beta}_1) \) is the predicted value from the metamodel fitted to the averaged series, \( z_{1-\alpha/2} \) denotes the \( 1 - \alpha/2 \) quantile of the standard normal distribution and \( \hat{V}_t = g(X_2; \hat{\beta}_2) \) is the predicted value from the metamodel fitted to the series of variances of the averaged series.

It should be noticed that this is an approximated confidence interval, since we are considering the predicted values from two metamodels.

**EXPERIMENTATION**

In this section, we illustrate how our procedure can be applied to a broad variety of situations and we experimentally evaluate it on a set of case studies: stationary and nonstationary \( M/M/s \) queues, and a network of queues with feedback.

**A Transient, Stationary \( M/M/1 \) Queue**

We analyze the number of entities in the system and the time spent in the system of a stationary \( M/M/1 \) queue, with a utilization factor of \( \rho = 0.9 \) (\( \lambda = 0.9 \) and \( \mu = 1 \)). We chose two different initial conditions, in order to induce distinct initial transients: an empty and idle system (no customers waiting, \( a = 0 \)); and 113 customers already waiting at time zero (\( a = 113 \)). For both cases, we performed 30 independent replications of the corresponding simulation models and collected 3000 observations of each response. The number of entities in the system was collected at 1 time unit intervals, starting at instant 0.5, and the time spent in the system was collected according to the customer’s arrival order.

Then, the corresponding observations across runs were averaged and the sample variance was also calculated. When \( a = 0 \), the metamodels that showed best results for the number of entities in the system and for the time spent in the system were, respectively:

\[ \overline{N}_t = \frac{\hat{\beta}_1}{1 + \rho} + u_t, \]

\[ \overline{T}_t = \hat{\beta}_0 + \hat{\beta}_1 \tanh(\hat{\beta}_2 t) + u_t, \]

with \( u_t \) an ARMA(\( p,q \)) satisfying

\[ u_t = \phi_1 u_{t-1} + \cdots + \phi_p u_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q}, \]

and where we assume that \( \{ \varepsilon_t \} \) is a sequence of independent, normally distributed, random variables with zero mean and variance \( \sigma^2 \), that is, Gaussian white noise.

We applied the Shapiro-Wilk test, the Ljung-Box test for 10 lags and the \( t \) test, in order to validate the estimated metamodels. All the coefficients were significant and no evidence was found to reject the hypotheses of normally distributed, uncorrelated residuals.

In Figure 1, we represent the original averaged series (black curve) and the fitted deterministic component of the metamodel (blue curve) for the time spent in the system. Although the estimated deterministic component of the metamodel readily captures the underlying behavior of the averaged series, both slightly overestimate the true asymptotic value (magenta dashed curve). This emphasizes the importance of a carefully chosen simulation duration and the handling of the initial bias—we addressed the latter problem in Brandão and Porta Nova (2010).

**Figure 1:** Original Data and Fitted Metamodel (M/M/1 Queue, with \( \rho = 0.9 \) and \( a=0 \))

The same analysis was performed for \( a = 113 \). This time, for the number of entities in the system, we fitted the metamodel given by (3) and for the time spent in the system the metamodel:

\[ \overline{T}_t = \beta_1 e^{\beta_2 t} + \beta_3 t + \beta_4 + u_t, \]

In both cases, all the estimated coefficients were significant, but we could not update the preliminary metamodel, incorporating the autocorrelation structure of the regression residuals.
Having estimated the metamodel for the averaged series, to build confidence intervals for the response, we need to estimate a metamodel for the sample variance. To illustrate this step, we chose the sample variance of the number of entities in the system for \( a = 113 \). After several attempts to find an appropriate metamodel, we found out the following:

\[
V_t = \beta_1 t^{\beta_2} + \beta_3 t + u_j, \tag{5}
\]

Again, only the preliminary metamodel was estimated. The curves represented in Figure 2(a) give us a general idea of the behavior of the metamodel for the average number of entities in the system and the corresponding limits of the proposed confidence intervals. They are the averaged series (black curve), the metamodel (blue curve), the limits of CI1 (green curves) and the limits of CI2 (red curves). As can be seen, the fitted metamodel keenly captures the underlying behavior of the response and the two confidence intervals produce similar results.

The sample variance (black curve) and respective fitted metamodel (blue curve) are represented in Figure 2(b).

We observe, in this case, that the variance has not yet stabilized. We recall that the metamodel for the variance is based on a single replication.

![Figure 2: M/M/1 Queue, with \( \rho = 0.9 \) and \( a = 113 \)](image)

The metamodels that were fitted to the exploding \( M/M/1 \) queue (\( \rho = 2 \)) for the number of entities in the system and for the time spent in the system were, respectively:

\[
\bar{N}_t = \beta_1 t + u_t \tag{8}
\]

\[
\bar{T}_t = \beta_0 + \beta_1 t + u_t \tag{9}
\]

Again, in order to construct confidence intervals for the responses, metamodels had to be fitted to the corresponding sample variances. In both cases, we fitted a polynomial of degree 2:

\[
V_t = \beta_1 t + \beta_2 t^2 + u_t \tag{10}
\]

Figure 4 shows the same curves as Figure 2(a), but now for the responses: average number of entities in the system (\( M/M/1 \) with \( \rho = 1 \)) and average sojourn time (\( M/M/1 \) with \( \rho = 2 \)).

![Figure 4: CI for the M/M/1 Queue (\( \rho = 1 \) and 2)](image)
In order to analyze the performance of the proposed confidence intervals, we carried out an experiment consisting of 3000 independent replications of the simulation model of the $M/M/1$ queue, for the utilization factors 1 and 2. Then, we computed 100 replications of the averaged series, and corresponding series of the sample variances, for the number of entities in the system. The metamodels (7) and (10) were fitted to the averaged series and to the sample variance series, respectively. Finally, we calculated the two confidence intervals for each instant. The performance measures used were: the coverage rate, the relative precision (the average CI half-length divided by the absolute value of the average CI midpoint), the mean, and the sample variance of the half-lengths of the 95% confidence intervals. The coverage rate was determined using the analytical results presented in Bailey (1964; 1957). The results that were obtained at some selected instants are reported in Table 1.

Table 1: 95% CI for the Average Number of Entities in the System ($M/M/1$ Queue, $\rho = 1$ and $\rho = 2$)

<table>
<thead>
<tr>
<th>$\rho$</th>
<th>Performance</th>
<th>Measure</th>
<th>Instants</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coverage rate</td>
<td>$12.5$</td>
<td>$67.5$</td>
</tr>
<tr>
<td>1</td>
<td>96</td>
<td>95</td>
<td>93</td>
</tr>
<tr>
<td></td>
<td>Relative precision</td>
<td>0.30</td>
<td>0.30</td>
</tr>
<tr>
<td></td>
<td>Avg. CI half-length</td>
<td>1.06</td>
<td>2.58</td>
</tr>
<tr>
<td></td>
<td>Var. CI half-length</td>
<td>0.04</td>
<td>0.17</td>
</tr>
<tr>
<td>C12</td>
<td>Coverage rate</td>
<td>95</td>
<td>99</td>
</tr>
<tr>
<td></td>
<td>Relative precision</td>
<td>0.30</td>
<td>0.28</td>
</tr>
<tr>
<td></td>
<td>Avg. CI half-length</td>
<td>1.05</td>
<td>2.45</td>
</tr>
<tr>
<td></td>
<td>Var. CI half-length</td>
<td>0.02</td>
<td>0.08</td>
</tr>
<tr>
<td>2</td>
<td>Coverage rate</td>
<td>95</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>Relative precision</td>
<td>0.15</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td>Avg. CI half-length</td>
<td>2.05</td>
<td>5.11</td>
</tr>
<tr>
<td></td>
<td>Var. CI half-length</td>
<td>0.07</td>
<td>0.39</td>
</tr>
<tr>
<td>C12</td>
<td>Coverage rate</td>
<td>91</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>Relative precision</td>
<td>0.16</td>
<td>0.07</td>
</tr>
<tr>
<td></td>
<td>Avg. CI half-length</td>
<td>2.11</td>
<td>4.93</td>
</tr>
<tr>
<td></td>
<td>Var. CI half-length</td>
<td>0.07</td>
<td>0.27</td>
</tr>
</tbody>
</table>

We can observe that for $\rho = 1$, the results from both confidence intervals are reasonable, since the relative precisions are around 0.3. In contrast, for $\rho = 2$, there is an improvement on the values of relative precision. With respect to the coverage rates, there is some undercoverage for independent replications, and for C12 there are instants with undercoverage and others with overcoverage.

Bounded and Unbounded Cyclical Responses

These cases were analyzed in Brandão and Porta Nova (2004), for a slightly modified version of the Single Lane Traffic Analysis classical example; see Nozari et al. (1984). In a two-lane road, one needs to be repaired and traffic lights are placed at the limits of the closed section. A full cycle of the lights consists of green in direction 1, both red, green in direction 2, and, finally, both red. In order to illustrate the two types of system behavior, we present the results for the queue lengths of direction 2 in two distinct situations: normal traffic conditions, and one-hour rush period. As before, we performed 30 independent replications and collected 360 observations of the response at intervals of width $\Delta t = 10$.

The metamodels fitted to the average queue length of direction 2 with normal and congested traffic were, respectively:

$$\begin{align}
\hat{N}_{2t} &= f(t; \beta) + u_t \\
\hat{N}_{2t}^\gamma &= \beta_0 + \beta_0 t + f(t; \beta) + u_t
\end{align}$$

with

$$f(t; \beta) = \frac{\beta_1}{2} + \frac{2\beta_1 \beta_2^2}{(\beta_2^2 - 1) \pi^2} \left\{ \sum_{n=1}^{18} \cos \left[ \frac{(\beta_2 + 1)n \pi}{\beta_2} \right] - \frac{1}{n^2} \cos \left( \frac{n \pi t}{\beta_2} \right) \right\}$$

the fourier series, with 18 terms, of an asymmetric triangular wave. In both cases, SARIMA models were fitted to the residuals and it was not possible to update the preliminary metamodels. Nevertheless, the original and the metamodel curves are virtually indistinguishable, as can be observed in Figure 5(a) and (b).

![Figure 5: Original and Fitted Values: Average Queue Length, Direction 2](image)

A Network of Queues with Feedback

We conclude the experimentation with a case that is an exception to the other situations that were considered, Model 5: A Network of Queues from Schruben (1982). It is a network of three capacity limited $M/M/s$ queues with feedback (blocked customers must reenter the service queue just completed); see Figure 6. In this case, we analyze the total number of entities waiting in all queues and the time that the customers spent in the system. The experiments were performed under the same conditions as the nonstationary $M/M/s$ queues, but now we collected the time spent in the system according to
the departure index, since some customers fail to enter when the first queue is full.

For both responses, we fitted the logistic metamodel given by equation (2) to the averaged series. The Weibull metamodel was fitted to the sample variance of the response average sojourn time:

$$V_t = \beta_1 - \beta_2 + e^{-\beta_2 t^\beta_4} + u_t.$$  \hspace{1cm} (13)

As before, the averaged series, the metamodel values and the two confidence intervals are shown in Figure 7.

**CONCLUSIONS**

In this paper, we present a methodology for analyzing general nonstationary or transient discrete-event simulation responses, based on regression models for the deterministic components of the metamodels, and time series models for the residuals. We feel that the experimentation on the varied set of case studies from the simulation literature showed that our approach is rather effective. However, we also feel that more work is necessary to validate the proposed confidence interval based on the estimation of metamodels for both the averaged and variance series.

**AUTHOR BIOGRAPHY**

**RITA MARQUES BRANDÃO** is an Assistant Professor of the Department of Mathematics at the University of Azores and a member of CEG-IST. Her web address is <www.rbrandao.uac.pt> and e-mail address is <rita@uac.pt>.

**ACÁCIO M. O. PORTA NOVA** is an Associate Professor of the Department of Engineering and Management at the Superior Technical Institute (IST) of the Technical University of Lisbon. His e-mail address is <apnova@ist.utl.pt>.

**REFERENCES**


COMPARISON OF MONTE CARLO METHODS THROUGH A CASE STUDY

Megdouda OURBIH–TARI
Abdelnasser DAHMANI
Department of Mathematics
University A. Mira of Bejaia
Laboratory of Applied Mathematics
Algeria

Latifa OURBIH-BAGHDALI
Ecole Nationale Superieur des Travaux Publics (ENSTP), Algiers
E-mail: megtari | baghdalil | a_dahmany @yahoo.fr

KEYWORDS
Modelling, Activity Cycle Diagram, Discrete-Event Simulation, Refined Descriptive Sampling, Monte Carlo.

ABSTRACT
This work investigates how a three phase discrete-event simulation system PSim: Turbo Pascal Three Phase Simulation Routines and its libraries using Refined Descriptive Sampling (RDS) and Random Sampling (RS), can be used in planning of aggregates production at a minimum cost. A model is developed using the activity cycle diagram and the performance measures of the system are evaluated using the developed software "Carrier Simulator". Inefficiencies of the studied problem are identified and improvement is proposed for the carrier managers by the determination of the optimal strategy. Furthermore, this paper focuses on the statistical comparison of both sampling methods and the obtained results prove once more the efficiency of RDS over RS.

INTRODUCTION
Surveys of modelling practice demonstrate that simulation is one of the most widely used modelling techniques of systems performance measures evaluation. It involves the development of an imitation on a computer of the system under study, followed by experimentation to understand and investigate improvements to the system. Discrete-event simulation can help to identify inefficiencies and examine how these may be addressed. This paper examines Civil Engineering problem throughout a three phase discrete-event simulation systems and focuses on a case study of aggregates quarry Adhrar Oufella-ALGRAN of Bejaia in Algeria at the operational level, considering the random behaviour, the resource characteristics and dynamic interactions during operations. A logical model for the career aggregates using an expanded Activity Cycle Diagram (ACD) is built from which we have simulated the behaviour of the quarry on a computer. The ACD expresses the logic of complex simulation models effectively and is used as a vehicle for experimentation. So, experiments are carried out on the model built and unknown parameters of the output random variables of interest denoted as the unknown but observable transformation of X through the model built are estimated. Hence, Monte Carlo methods are usually used for such problems.

Since the simulation results may be difficult to compute for each new value of X, it is important to pick a sampling scheme that allows us to estimate the simulation results well while keeping M to a minimum. A lot of statistical methods for choosing input random samples exist. The simplest is Random Sampling (RS) (Fishman 1997) which is used to generate M identically independent distributed (iid) random vectors with the distribution of X using some Random Number Generators (RNGs). RNG provides sequences of real numbers in the interval [0,1]. These numbers are viewed as realizations of iid U(0,1) random variables. This statistical RS method is commonly used to simulate systems containing stochastic or probabilistic situations (Andersson 2001). It might also be considered to solve some deterministic problems that cannot be solved analytically (Robert and Casella 2004).

Saliba (Saliba 1990) proposed an alternative sampling method of generating input random samples that he calls Descriptive Sampling (DS) based on a deterministic selection of the input sample values and their random permutation. It controls completely the set variability but keeps the sequence effect so the resulting estimates are more accurate than those of RS and an input random variable is then well represented. For its use, DS requires known input sample size in advance. Another problem was emerged using this method is the possibility of producing biased results through simulation. Both problems were discussed by some authors e.g. (Pidd 2004). Later on, these problems were considered by analyzing the response surface (Tari and Dahmani 2005a). Some models for the response surface have been proposed and only one of them produces high bias for the simulation estimates. Once the worst case model was found, it was then isolated and a new approach called: Refined Descriptive Sampling (RDS) was proposed (Tari and Dahmani 2006) to make DS safe, efficient and convenient. It has already been explored how RDS behaves on a manufacturing production system handled by the three phase approach in discrete-event simulation. We have seen how it can reduce significantly the sampling bias of the output performance measures compared to DS and RS methods. The production problem
chosen has a theoretical solution obtained from a stochastic
Petri network that was used for comparison with all cited sampling methods (Tari and Dahmani 2005b). RDS was also tested on a flow shop problem. A unidirectional flow production system of flow shop type has been analyzed
using the different sampling methods. Then, its performance measures were established using activity
approach in the discrete-event simulation. The obtained results strongly support the efficiency of RDS (Tari and Dahmani 2005c).

This paper investigates how RDS method can be outperformed on Civil Engineering problems. Then, our sample system is simulated by PSim: Turbo Pascal Three Phase Simulation Routines and its libraries using RDS and RS with a focus on the statistical comparison of both sampling methods and the obtained results prove once more the efficiency of RDS over RS on a branch of engineering. The purpose of the simulation is to improve the system being studied by determining the optimal strategy.

The three phase simulation system is based on specific libraries of the three phase approach; these libraries are available in different programming languages, Turbo Pascal, C, C++ and Visual Basic v3. The PSim software, related to the three phase approach is available in Turbo Pascal programming language and carried out by Pidd in the 5th version of 2004. It has been selected for a better simulation programming of the aggregates quarry problem. The selected three phase approach in discrete-event simulation fits well our requirements; nevertheless we can find other methods like the Activity based approach, the Event based approach and the Process interaction approach.

Section 2 and 3 are respectively devoted to describing RDS method and the studied problem. By then, a discrete-
event simulation model is given and the performance measures of the aggregates career system are established in section 5. Finally, the working and the environment of our Simulator is described.

REFINED DESCRIPTIVE SAMPLING

To reduce the risk of bias, RDS procedure was proposed as an alternative approach to Monte Carlo Simulation. This method is mainly concerned with a block of prime numbers which must be situated inside a generator aiming to distribute regular samples of prime size when required by the simulation. Compared to DS this approach removes the need to determine in advance the sample size.

Let \( p_q = 1, 2, 3, \ldots \) be distinct prime numbers and a simulation experiment of \( M \) replicated runs, terminates when \( n_q \) prime numbers have been used. In this procedure, we present regular samples from \( p_q \) then \( p_{q+1} \) and so on for any \( q \) in random order as required by the simulation. We terminate when the simulation terminates. Using RDS, subset values for the input random variable \( X \) are generated as required by the simulation.

The general method of the inverse transform produces regular subset values given by

\[
x_i = F^{-1} (r_i) \text{ for } i = 1, 2, \ldots, p_q, \quad q = 1, 2, \ldots, m \text{ and } j = 1, 2, \ldots, M
\]

where

\( F^{-1}(r), r \in [0,1] \) is the inverse cumulative input distribution and

\( r_i = (i-0.5)/p_q \) with \( i = 1, 2, \ldots, p_q \), \( q = 1, 2, \ldots, m \) and \( j = 1, 2, \ldots, M \)

and the sequence of each regular subset \{ \( r_i, i = 1, 2, \ldots, p_q \) \} is randomised.

DESCRIPTION OF THE PROBLEM

In this section, we describe the system being studied, the career of Adrar Oufella-Bejaia which specializes in operating underground mines and open pit for the production of non-ferrous ores and useful non-metallic substances. The system being studied has five (5) classes of entity: Driller, Crusher, Drilling Team, Truck and Loader identified respectively by two characters Dr, Cr, DT, Tr and Lo. The carrier problem considers 15 input random variables which are: The different service times of loading, transport, unloading, crushing, truck return, drilling, mining, driller team shelter, drilling team return, loader shelter, loader return, driller shelter, driller return, firing and inspection. It observes 7 output variables through simulation. The occupancy rates of different entities have all one parameter, the mean to be respectively estimated by DrRate, CrRate, DTRate, TrRate, LoRate. The cost and the required time to produce a scheduled aggregates volume are both computed for choosing the best strategy. We suppose that the different input processes follow all Poisson distributions with different rates \( \lambda_i, i = 1, \ldots, 15 \).

The manufacture of a certain volume of aggregates at Adrar Oufella quarry must go through a sequence of operations that are:

Drilling of boreholes, mining, firing, inspection, loading the rock into the truck, transporting the rock, unloading the truck at the site of crushing, return of the truck to the quarry site for possible reloading and finally crushing the rock leading to the production of aggregates.

The Adrar Oufella career operates in two shifts of five hours each, one morning and one afternoon daily with a lunch break of one hour making the working day to 10 hours and the day to 11 hours. In this case, drilling a round is done in approximately 3 days, the longest time of all activities.

As long as the system performance is the aggregates production, the operations of loading and transporting that provide the rock to the crusher never stop during working hours except in case of force majeure. Thus, the optimal strategy for the quarry's operation is to carry out firing only during the lunch break when the operations of loading and transport stop anyway. The various activities that begin to approach the break are not allowed unless it remains at least 20 mm for the end of a shift, this offset adjusts operations in the breaks.

For that firing can take place during the break, all conditions must be met 20 mm before the break. The decision and the announcement of the firing cannot take place if there is not at least 30 mm before the break. Firing during working hours can only occur when there is not enough rock to be loaded and transported, and there are 60 holes already loaded with explosives. In this case, it is unnecessary to await the next break to fire, because all
operations are already stopped (with the exception of
drilling extra holes).

MODELLING THE QUARRY

The four main approaches to discrete event simulation
differ somewhat in their word-view of the system being
simulated. The basic building block of the activity approach
is the activities, these cause events to occur and follow from
events, somewhat analogous to network methods.

Modelling is representing the important interactions in a
model which has a form useful for simulation. The career
aggregates defined in section 3 is represented by its ACD in
figure 1.

Figure 1: The ACD representing the Adhur Oufella-
ALGRAN Carrier

THE THREE PHASE PROGRAM STRUCTURE

In this section, we present the software "Carrier
Simulator" developed for the simulation of ALGRAN
career problem which evaluates the performance measures
of the system in the three phase discrete event simulation
using different sampling methods for generating input
variables. The "Carrier Simulator" software is realized
under Turbo Pascal 7 (TP7) and implemented under the
Windows environment using the source code of the
program Turbo Pascal PSim Three Phase Simulation
Routines and its libraries. TP7 is a powerful programming
language integrating Pascal object programming and the
environment of programs conception visual and event
programming. After building the ACD, this information,
together with activity durations, attributes, branching
conditions and any other additional data that might be
necessary is fed into the PSim environment. Its
implementation requires the generation of artificial samples
of known distribution variables using random number
generators according to the sampling method.

B and C Activities

Two types of activity are identified, B and C activities. B
for Bound activities are executed whenever the time at
which they are scheduled is reached. C for Conditional
activities whose initiation depends on either the co-
operation of different entities or the satisfaction of certain
specified conditions, or both.

One or several activities of either type will be initiated
whenever the system changes state, i.e. whenever an event
occurs. The system of the quarry may change state at any of
the following points: Beginning and end of drilling, mining,
drilling team shelter, driller shelter, loader shelter, firing,
inspection, drilling team return, driller return, loader return,
loading, transport, unloading, truck return and crushing.

Simulation Executive

In a three phase simulation, it is normal to have an
executive or control program to ensure that the entities are
properly scheduled for future activities (known as Bs) and
also to ensure that current activity (often known as Cs) are
properly sequenced. In the actual case, this is managed by
representing the entities in the following simple form like in
PSim.

The Details array is examined during the A Phase, so as
to find the time of the next event. This involves a simple
search for the minimum TimeCell, disregarding any rows in
which Avail is True. The row numbers, referred to as the
time ID, of entities with minimum TimeCells are placed in
the CurrEntArray.

During the B Phase, the focus is on the CurrEntArray.
This is worked through, one element at a time and the B
Phase procedure takes the entity ID and then executes the B
indicated by the entity's NextAct field in the Details array.

The Main Units of the Software

The Routines Provided

PSim or the main program "PasExec" contains the 3
phase executive and other procedures needed for the
processing and is based around the following, each of
which represents a Turbo Pascal unit. "GenLib" contains
useful functions for frequently needed tasks such as
input/output. "ExecVars" contains global variables needed
by "PasExec", "ExecUtils" and "model". "ExecUtils"
contains functions that are used mainly by the simulation
executive and the "model". Finally, the "model" contains
the simulation model. The run-time program is produced by
compiling and linking these files within the Turbo Pascal
Interactive Development Environment.

The Proposed Routines

For better programming, some changes were made to
"PasExec" unit. Stopping the simulation is not supported by
the duration of simulation but by the volume of aggregates to produce, this parameter is fixed by the user. Execution of "PasExec" was extended to N scenarios and M replicated simulation runs for each scenario. The number N and M are also fixed by the user. After the run-time of all scenarios, a call to a new procedure entitled "Finalisation2" is executed.

The latter procedure defined in "Model" unit allows the computation of means, variances and finding the best scenario according to averages of the production cost. The unit "Model" contains the constants, types and variables necessary for processing the program, as well as procedures such as, all B and C activities procedures, initialization, finalization and finalisation2.

Experiments and software interface

**EXPERIMENTS**

It is important to remember that the purpose of RDS is to reduce the variability of simulation estimates. Therefore, a single run cannot show this benefit, because only one estimate is produced for each parameter under study. Therefore, we carried out N=6 different scenarios of different number of trucks and loaders, for the production of 31500m³ volume of aggregates, each scenario with M=30 replicated simulation runs for each sampling method, using the same following simulation input parameters.

- The cost of mining a round is assumed to 7372 DA. The conversion between the Algerian Dinar (DA) and the Euro is approx. € 1 ≈ 120 DA.
- A round provides an average volume of rock, after firing, equal to 7920 m³.
- The cost of the drilling team in activity is assumed to 400 DA/h. Indeed, we assumed a team of 4 people where each person is paid 1100 DA/ working day. Therefore, the awaiting cost of the team is estimated at 400 DA/ working day i. e. 36.36 DA/h.
- The cost of the driller in activity and awaiting is respectively assumed to 9171 DA/h and 7492 DA/h.
- The cost of the crusher in activity and awaiting is respectively assumed to 4087 DA/h and 1000 DA/h.
- The cost of the truck in activity and awaiting is respectively assumed to 1328 DA/h and 548 DA / h. We assumed that all trucks have the same volume.
- The cost of the loader in activity and awaiting is respectively assumed to 4734 DA/h and 1000 DA/h.

All these parameters are given by ALGRAN Carrier and any others could be used to predict the future behavior of the system or to simulate another optimization carrier problem.

**Data Inputs**

For the first window, we kept the external appearance of the program Psim. So, when running the program PasExec, a first window appears prompting the user to enter system data, in the following order: The number of scenarios, replicated simulation runs, loaders and trucks, the smallest capacity of trucks, the capacity of the crusher, the average amount of rock spilled after firing, the volume of aggregates to produce, the mean time of the different input variables. See figure 2.

![Figure 2: The 1st window summarizing the input data of the quarry simulation](image)

After that, a second window appears and invites the user to enter the costs per hour of activity and awaiting of different entities, the cost of mining, the delay for a possible stop of the simulation duration and an invitation to keep or not the trace of the execution. If yes, please give a name to the file. See figure 3.

![Figure 3: The 2nd window summarizing the input data of the quarry simulation](image)

From the second scenario, the program will require just the number of loaders, trucks and display delay since the rest is unchanged.

**Results Output**

For each scenario, the following results for both sampling methods will be displayed: The number of trucks and loaders of the current scenario which does not change throughout the execution of the M replicated simulation runs, the number of the current scenario, the number of the current simulation run, the volume in cubic meter of available rock, the current simulation duration in minutes, the use, shelter and return number of the driller, the mining, shelter, firing, inspection and return number done by the drilling team, the use, shelter and return number of loaders, the use number of crusher, the transport, unloading and return number of the trucks, the volume of the achieved aggregates given in m³. All these results change throughout the simulation experiment. Once, all M simulation runs are carried out, the cost of the current scenario is compared to the previous scenario's cost and the one with a minimum cost is kept, after that, another scenario will start. Once all simulation runs and all scenarios are carried out, the best strategy to produce the scheduled volume of aggregates is displayed, indicating the number of trucks, the number of
loaders, the cost and time required for this production and the mean and variance of occupancy rates of different entities for the best strategy. These last simulation results are used for the comparison of both sampling methods using the variance as a statistical criterion since both sampling methods produce unbiased estimators through simulation. See figure 4 and 5.

Figure 4: Simulation results related to the carrier system using RS

Figure 5: Simulation results related to the carrier system using RDS

CONCLUSION

According to the different scenarios carried out, inefficiencies of the studied problem were identified and it has been shown through the three phase discrete-event simulation and whatever the sampling method used, that the carrier system can be improved to 2 loaders and 6 trucks for the production of 31,500 m³ of aggregates at a minimum cost. Therefore, the sampling method does not influence the choice of the best strategy. But the use of RDS, influence a lot over RS, when the concern is the minimization of the cost to produce a scheduled volume of aggregates as it provides a smaller cost than RS for the same and best strategy. As a result, RDS represents better the input random variables than RS.

On the other hand, the best performance of the carrier is demonstrated using RDS through the mean occupancy rates of different entities and their variance values strongly support the efficiency of RDS over RS on a civil engineering problem. We conclude that RDS procedure offers a better alternative to RS.

The design of the software "Carrier simulator" is special in many ways. First, it affords the comparison of both sampling methods and is useful for ALGRAN manager as all parameters are fixed by the user. The "Carrier Simulator" is designed to support a carrier of aggregates production with any number of loaders and trucks. The software affords a multitude of simulation scenarios or experiments. This provides an opportunity for user communities to have access to solve carrier problems by simulation.

REFERENCES


MULTI-AGENT SIMULATION
A Survey on Complexity Sciences and their Modeling Techniques

Morteza Mahmoudzadeh and Javad Jassbi
Department of Industrial Management,
Science and Research Branch,
Islamic Azad University, Tehran,
Iran
Email: {m.mahmoudzadeh, jassbi}@srbiu.ac.ir

KEYWORDS
System Thinking, Complexity Theory, Modeling Techniques

ABSTRACT

In recent years there is a tendency for using new modeling techniques such as agent based modeling in physical and social sciences rather than previous variable based modeling techniques. Researchers who use these techniques justify that they could model emergence, self-organization and adaptation which are properties of complexity theory. These properties are share with system thinking but with different definitions. In most of the problems which are modeled by agent based modeling, these differences have been ignored. Hence these applications have been restricted to model, previously modeled problems only in a new manner and ignoring the potential capabilities of this technique in modeling issues such as creativity and novelty. This paper based on differences of complexity theory and system thinking and capabilities of modeling techniques of complexity sciences, proposes a new map that guides researchers for using the best modeling technique based on problem’s properties.

INTRODUCTION

There is no doubt that today’s dominant discourse on the modeling of physical and social phenomena has been built on two strands of thinking developed during the course of the twentieth century. The first is Newtonian mechanistic view, and the second is system thinking. In mechanistic view the universe was understood to function in a mechanistic clockwork manner. To identify any phenomenon- in this method- it is reduced to parts, and the whole is simply the sum of the parts. But in Systems thinking the whole came to be thought of as a system and the parts as subsystems within it. The parts aren't simply additive and they affect each other. The focus of attention shifted from understanding the parts, or entities of which the whole was composed, to the interaction of subsystems to form a system. (Stacey & et al 2000). They also argue that system thinking is based on Kant’s static resolution of paradox (thesis and antithesis) in a “both…and” paradigm. They named this form of causality, formative causality in which the internal dynamics (form of the system) play a major role in determining its behavior. On the other hand internal dynamics is given, and the behavior does not cause the form. This means that in this manner the role of the parts is restricted to assembling into the whole.

An alternative notion of the whole is that, a whole is never complete and it is under perpetual construction. Stacey & et al (2000) argue that this idea has been built on Hegel’s thought in which parts form the whole while being formed by the whole in paradoxical movement(based on Hegel's dialectic -thesis, antithesis and synthesis.

Another issue in system thinking is the definition of self-organization and moving toward a knowable future. System thinking assumes that there is a knowable equilibrium state in future which system moves toward it. In this manner self-organization is a process of unfolding an enfolded form.

An alternative notion of this issue is the definition of self-organization as a process of interaction characterized in an essential way by paradox and the emergence of the truly unknowable. Stacey & et al (2000) named this form of causality, transformative and adaptionist causality. They also argue that these forms of causality are the basis for complexity sciences which and cause novelty and creativity in human organizations. Phelan (1999) distinguishes between system thinking and complexity theory based on Cohen and Cyert (1961) definition of confirmatory and exploratory analysis.

**Confirmatory analysis:** "... aims at understanding the operating characteristics of a total system when the behavior of the component parts is known with a high degree of accuracy."

**Exploratory modeling:** aims "... to derive a set of component relations that will lead to a total system exhibiting the observed characteristics of behavior".

He argues that Systems theory predominantly focuses on confirmatory analysis. It seeks to identify relationships between elements in a system and then to optimize some objective functions and it depends on feedback loops to characterize most of the relationships and interactions between system elements. He points out three points of departure between complexity and systems theory and argues that complexity theory tends (i) to focus on exploratory analysis, (ii) to use agent-based modeling, and (iii) to maintain that complexity arises from the interaction of agents following simple rules.
SYSTEM THINKING METHODS

The systems theories have been developed along three pathways over much the same period of time: general systems theory, cybernetics, systems dynamics. Figure 1 depicts the historical development of system thinking methods and their expansions.

General Systems Theory
The central concept in general systems theory is homeostasis, which means that systems have a strong tendency to move toward a state of order and stability, or adapted equilibrium.

Cybernetic System
Cybernetic systems are self-regulating and goal-directed systems adapting to their environment. These systems are working through a process of negative feedback operation.

System Dynamics
In systems dynamics, mathematical models of a system are constructed, consisting of recursive, nonlinear equations that specify how the system changes states over time. One important difference from the other two systems theories is the recognition of amplifying, or positive, feedback as well as negative feedback. Another difference is the introduction of nonlinear responses into a chain of circular causality that could lead to unexpected and unintended outcomes, which means that it can no longer be assumed that the system will move to equilibrium. In the theories of cybernetics and general systems there is also linear causality of an efficient kind. For both, if there is a change in the environment, then the system will adapt. It is a gap between environment and internal state that triggers a change back to an equilibrium state in a straightforward linear manner. History here is unimportant since change toward equilibrium is triggered only by the current gap between environmental conditions and internal state. In the systems dynamics strand, as well as formative causality, there is also efficient causality but this time it takes a circular nonlinear form. The behavior produced by a system in one period of time feeds back through the system to determine behavior in the next time period. In systems dynamics, history does, therefore, play a part.

Some researchers have criticized some aspects of system thinking and tried to solve these challenges by new method. Two of these challenges and response to them has been mentioned in below:

Problems with system boundaries
In systems thinking there is always an observer, that is, someone who delineates a system of interacting parts and identifies or designs rules of interaction for those parts. So the patterns of behavior they will produce are already there, in the identified or designed rules of interaction. The system, in systems thinking, can only do what it is designed to do.

Participation
Another criticism on systems theories is that they imply that organizations are physical entities like organisms with clear boundaries, structures and functions. Hence systems theories present individuals as deterministic and thinking machines and ignore the emotion, conflict, politics and cultural aspects of organizational life.

Responses to critiques
In response to these critiques the system thinkers have expanded it in two dimensions. The first is vertical expansion of system definition to include higher and higher cognitive levels, including more and more aspects of human cognition. As we go up in each level there is always a higher order cybernetic system that controls and changes. Second order Cybernetics is one of these models (Foerster 1974,1981). The second response is a horizontal widening of the boundary to include cultural and political systems in the definition of the relevant system. Checkland (1981) advocate an interpretive approach to systems thinking in which account is taken, presumably by the designer of the system, of the social rules and practices of participants in the system He defines a model, a learning cycle with a number of steps, that constitutes the Soft Systems Methodology. However, there has to be a limit to deepening and widening the boundaries because the former runs into infinite regress and the latter into a degree of complexity that cannot be dealt with (Flood, 1990). The major criticism of this move is that in this extension at each stage in deepening and widening the boundary of the system, the source of novel change is located in the heads of the designers and in the other hands outside of the system.

COMPLEXITY THEORY

Complexity Sciences based on Weaver’s (1948) classification of complexity are Organized-Complexity Systems. In the organized-complexity system, the typical form for living systems, only a finite but large number of components will define the system (Skyttner, 2005). Based on Ackoff’s (1971) classification of systems, these systems
are goal changing systems. McKelvey (2004) identifies two schools in complexity science: European and American. He argues that the European group such as dissipative structure Theory, focus on physical phenomena and they are math intensive. American group such as Chaos Theory and theory of Complex Adaptive Systems study complexity with heterogeneous agents and the rules that govern their behavior and interactions in search space. He also argues that the precursor of complexity theory is research into the phenomenon of chaos. Stacey & et al (2000) argue that there is no single science of complexity but, rather, a number of different strands comprising what might be called the complexity sciences. They also argue that complexity scientists usually draw on concepts to be found in one or more of three of these strands – namely, chaos theory, dissipative structure theory, and the theory of complex adaptive systems.

**Chaos Theory**

Chaos theory provides an explanation of the behavior of a system that can be modeled by deterministic nonlinear equations in which the output of one calculation is taken as the input of the next. Chaos theory shows how particular control parameters, determined outside the system, cause its behavior to move toward a particular state space called an attractor. Such systems have the potential to move to one of a number of different attractors, depending upon the parameter values. Attractors describe global patterns of behavior displayed by a system. At some critical level of the control parameter, between levels that lead to equilibrium attractors and those that lead to instability, behavior is drawn to a strange attractor. Strange attractors are paradoxically regular and irregular, stable and unstable, at the same time. Chaos theory, then, produces a rather clear conclusion. Any system governed by recursively applied nonlinear laws may display behavior of the strange attractor type at certain parameter values. When it follows a strange attractor its behavior is predictable at global, macro levels of description, but only in qualitative terms. At the specific micro level, predictability is confined to short-term local occurrences, leaving the specific long-term trajectory unpredictable due to the inability of humans to measure with infinite accuracy (Gleick, 1988).

What is distinctive about chaos theory, compared to the other systems theories, however, is the clear identification of the limits to predictability. From a chaos perspective, this move toward the simplicity of equilibrium could be interpreted as a move toward failure, where health and success are strange attractors in which long-term predictability of specific trajectories is impossible. For systems dynamics thinkers, the aim is to identify leverage points for interventions that will enable them to identify where, when and how to initiate change. (Stacey & et al, 2000)

**Dissipative Structure Theory**

Prigogine and Stengers (1984) identify a dynamical pattern of change for chaotic systems in the form of following statements:

- A system, is held far from equilibrium by some environmental constraint.
- In this condition, small fluctuations are amplified to break the microscopic symmetry of the entities comprising it.
- At a critical level of environmental constraint the system reaches a bifurcation point. This is a point at which the system becomes unstable and has the possibility of developing along a number of different pathways.
- At this bifurcation the whole ensemble of entities spontaneously self-organizes, in effect “choosing” a pathway, one of which could produce a new pattern. In other words, long-range correlations form between the entities and a new coherent pattern suddenly emerges without any blueprint, one that cannot be explained by, or reduced to, or predicted from, the nature of the system’s component entities.
- That pattern is a dissipative structure, that is, one that dissipates energy or information imported from the environment, so continuously renewing itself. The structure is an evolving interactive process that temporarily manifests in globally stable states taking the form of irregular patterns, and it is essentially a contradiction or paradox: symmetry and uniformity of pattern are being lost but there is still a structure; disorder is used to create new structure.

There is an important difference between chaos theory and the theory of dissipative structures. A chaos model of a system cannot explain how a system might move from one pattern of behavior, one attractor, to another. When the system is close to equilibrium, fluctuations are unimportant because they are rapidly damped away by the system’s movement to equilibrium. However, far from equilibrium, the dynamic is such that fluctuations are amplified to the point where symmetry is broken. In other words, the dynamics of instability break existing patterns so that the system approaches a bifurcation point at which behavior becomes highly unstable. At this bifurcation point, a number of different patterns of behavior, different pathways for future development, become possible and the system, in a sense, “chooses” one of these. In other words, the system displays the capacity to move from one attractor to another. While the macroscopic equations describing the system specify the different pathways, there is nothing in those equations that determines the “choice”. It is a spontaneous movement of the system that depends upon the micro detail of the fluctuations at that particular point in time (incorporated in the equations as a randomizing factor), and that movement may be the emergence of a new complex order. Central to Prigogine’s approach, at all levels, is the distinction between individual entities and populations, or ensembles, consisting of those entities. He then argues that individual trajectories cannot be specified for complex
systems, not simply because humans are unable to measure with infinite precision, as in chaos theory, but for intrinsic reasons as follows:

Since individual trajectories cannot be identified for intrinsic reasons, Prigogine takes the ensemble as fundamental and argues that change in whole ensembles emerges over long periods through the amplification of slight variations in individual entities; that is, the variability of individuals in the case of organisms or microscopic collisions in the case of matter.

**Complex Adaptive Systems (CAS)**

A complex adaptive system consists of a large number of agents, each of which behaves according to its own principles of local interaction. No individual agent, or group of agents, determines the patterns of behavior that the system as a whole displays, or how those patterns evolve.

<table>
<thead>
<tr>
<th>Path</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>View</td>
<td>Information and connections</td>
</tr>
<tr>
<td>Intention</td>
<td>Goals</td>
</tr>
<tr>
<td>Speech</td>
<td>Communication among the agents</td>
</tr>
<tr>
<td>Action</td>
<td>Interaction</td>
</tr>
<tr>
<td>Livelihood</td>
<td>Payoffs</td>
</tr>
<tr>
<td>Mindfulness</td>
<td>Cognition</td>
</tr>
<tr>
<td>Concentration</td>
<td>Model Focus and heterogeneity</td>
</tr>
</tbody>
</table>

Here self-organization means agents interacting locally according to their own principles, or “intentions,” in the absence of an overall blueprint for the system. A CAS not only self-organizes, but can direct its activity towards its own optimization. Grimm and Railsback (2005) argue that adaptive in CAS refers to the entities making up the system, not the system itself, because there is no clear concept of fitness at the system level. Dooley (2002), identifies three principles that typify complex adaptive systems. First, control and order are emergent, not predetermined. Second, a complex adaptive system’s history is irreversible. Third, a system’s future is often unpredictable. Fryer defines CAS as a system in which the agents as well as the system are adaptive: the system is self-similar. A CAS is a complex, self-similar collectivity of interacting adaptive agents. Other important properties are communication, cooperation, specialization, spatial and temporal organization, and of course reproduction. Communication and cooperation take place on all levels, from the agent to the system level. The forces driving co-operation between agents in such a system can be analyzed with game theory.

Murray (1998) defines these properties for Complexity sciences:

**Emergence:** Rather than being planned or controlled the agents in the system interact in apparently random ways.

**Co-evolution:** All systems exist within their own environment and they are also part of that environment.

**Connectivity:** The ways in which the agents in a system connect and relate to one another is critical to the survival of the system.

**Simple Rules:** Complex adaptive systems are not complicated.

**Iteration (Butterfly Effect):** Small changes in the initial conditions of the system can have significant effects after they have passed through the emergence - feedback loop a few times.

**Self-Organizing:** There is no hierarchy of command and control in a complex adaptive system.

Grimm & Railsback (2005) argue that emergence, adaptation, fitness, prediction, interaction and collectiveness are the main properties of a CAS. Miller and Page (2007) suggest an eightfold path (which is borrowed from Buddhism) that elements of it can be mapped to key modeling issues in a CAS (Table 1).

**MODELING TOOLS OF COMPLEXITY THEORY**

In literature of modeling there is variety of techniques for modeling complexity sciences (CS). Boccarda (2004) suggests differential equation modeling, Cellular Automata (CA) and Networks for modeling CS. Hofbauer and Sigmund (1998) suggest evolutionary game theory for modeling CS problems. Ihlinski (2001) suggest CA as modeling tool for discrete complex systems. Sterman (2000) argues that system dynamics is a method to enhance learning in complex systems. Figure 2 shows the historical development of modeling techniques.

**Differential Equations**

The study of dynamical models formulated in terms of ordinary differential equations began with Newton’s attempts to explain the motion of bodies in the solar system (Boccarda, 2004). In differential equations modeling a target system, with its properties and dynamics, is described using a system of equations which derive the future state of the target system from its actual state (Borrelli, Coleman, 2004).

**System Dynamics**

System dynamics has its roots in systems of difference and differential equations. Thus system dynamics -which first introduced by Forrester in 1960s- differ from systems of differential equations mostly in two technical aspects: discrete time is used as a coarse approximation for
continuous time to achieve numerical solutions; and functions of all kinds, not just continuous functions can be used (Gilbert 2005). In the system dynamics approach to modeling, one creates a model that expresses the temporal cause-and-effect relationships between variables, but individuals are not represented directly. System dynamics as its name implies, models systems of interacting variables and is able to handle direct causal links, such as growth in population leading to increased depletion of resources, and feedback loops, as when population growth depend on the food supply, but food supply depends on the level of the population (Sterman, 2000).

Cellular Automata
The origins of Cellular Automata can be traced back to mathematician John Von Neumann’s attempt to create a self-replicating machine (Schiff, 2008). A CA has the following features (Gilbert, Troitzsch 2005): It consists of a number of identical cells arranged in a regular grid. Each cell can be in one of a few states. Time advances through the simulation in steps. The states of a cell after anytime step is determined by a set of rules which specify how that state depends on the previous state of that cell and the states of cell’s immediate neighbors.

Because the rules only make reference to the states of other cells in a cell’s neighborhood, cellular automat are best used to model situations where the interactions are local. CA are prototypical models for complex systems and process consisting of a large number of identical, simple, locally interacting components. Most CA models usually possess these five generic characteristics (Ilachinski, 2001):

- **Discrete Lattice of cells:** the system substrate consists of a one-, two- or three-dimensional lattice of cells.
- **Homogeneity:** all cells are equivalent.
- **Discrete States:** each cell takes on one of a finite number of possible discrete states.
- **Local interactions:** each cell interacts only with cells that are in its local neighborhood.
- **Discrete Dynamics:** at each discrete unit each cell updates its current state according to transition rule taking into account the states of cells in its neighborhood.

Evolutionary Game Theory
John Maynard-Smith applied game theory to biology, and thus invented evolutionary game theory, he relaxed the premises behind GT Hofbauer and Sigmund (1998). Classical GT is a normative theory, in the sense that it expects players or agents to be perfectly rational and behave accordingly Tuyls and Parsons (2007). In classical game theory, interactions between rational agents are modeled as games of two or more players that can choose from a set of strategies and the corresponding preferences. Players in the classical setting have a perfect knowledge of the environment and the payoff tables, and try to maximize their individual payoff. However, under the biological circumstances considered by Maynard-Smith, it becomes impossible to judge what choices are the most rational. Instead of figuring out, a priori, how to optimize its actions, the question now facing a player becomes how to learn to optimize its behavior and maximize its return, and it does this based on local knowledge and through a process of trial and error. This learning process matches the concept of evolution in biology, and forms the basis of EGT. In contrast to classical GT, then, EGT is a descriptive theory, describing this process of learning, and does not need the assumption of complete knowledge and perfect rationality. GT assumes that players will compute Nash equilibrium and choose to play one such strategy. EGT assumes that players will gradually adjust their strategy over time in response to repeated observations of their own and others’ payoffs.

Agent Based Modeling
There is no general agreement on the definition of an agent. But the term is usually used to describe self-contained programs that can control their own actions based on their perceptions of their operating environments. Agent applications has been much influenced by work in artificial intelligence (AI), especially a subfield of AI called distributed artificial intelligence (DAI) which is concerned with the properties of and the design of networks of interacting agents (Gilbert 2005).

In agent-based modeling (ABM), a system is modeled as a collection of autonomous decision making entities called agents. It enables one to build models where individual entities and their interactions are directly represented. In comparison with variable-based approaches using structural equations, or system based approaches using differential equations, agent based simulation offers the possibility of modeling individual heterogeneity. It allows modelers to represent in a natural way multiplet scales of analysis, the emergence of structures at the macro level from individuals’ action, and various kind of adaptation and learning, none of which is easy to do with other modeling approaches (Gilbert 2008).

Bonabeau (2002) argues that the ABM has benefits over other modeling methods which could be captured in the following statements: (i) ABM captures emergent phenomena; (ii) ABM provides a natural description of a system; (iii) ABM is flexible. Grimm and Railsback (2005) argue that scientists who working in CS, attempt to understand the dynamics of systems of adaptive individuals, by using agent-based computer simulation.

CS PROPERTIES vs MODELING TOOLS CAPABILITIES
The properties of complexity theory and its differences from system thinking and also the capabilities of modeling tools was shown in the preceding sections. This paper based on these properties and their definition from the perspective of complexity sciences draws a map that shows which technique has potential capability to cover all aspect of CS.
Table 2. Shows this map in which the rows are the most important properties of complexity sciences and the columns are modeling tools. Each cell in table shows that a certain modeling tool could cover a property of CS or not? It is important to note that shared properties between two strands (system thinking and complexity sciences) like self-organization, emergence and adaption could be used in definition of complexity sciences in a tools when the tools covers other properties like as collection of agents or local interactions and diversity.

CONCLUSION

Based on Table 2., the agent-based modeling is the best tool that could cover all aspects of CS. This modeling technique could cover new definitions of self-organizing, emergence and evolution based on properties of complexity theory and so this tool has potential capability for studying creativity, novelty and moving toward new structures in the problem under study. There is poor literature in applications which use ABM in these areas and most of them have been used in situations which have some of CS properties that could be modeled by other techniques. If there is no diversity or chaotic behavior in a problem, the CA method is the second best way for modeling CS. In the case of lacking individual properties in system and when average property of population is enough for modeling the system, DE and SD methods are the best techniques. Iteration property in SD and DE is equivalent to path dependence and Butterfly effect. EGT however is a tool that supports collection of agents and bottom-up modeling but it couldn't cover other properties such as self-organization and emergence.

Table 2. modeling techniques vs CS properties

<table>
<thead>
<tr>
<th>CS Prop.</th>
<th>SD</th>
<th>DE</th>
<th>EGT</th>
<th>CA</th>
<th>ABM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emergence</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Collection of agents</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Self-Organization</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Adaptation</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Nonlinearity</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Simple Rules</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Diversity</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Evolution</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Iteration</td>
<td>√</td>
<td>√</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Local Interactions</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Bifurcation</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

REFERENCES


MULTI-HIERARCHICAL AGENT BASED MODELLING SIMULATION FRAMEWORK

Carlos Bragança de Oliveira
Antônio Carvalho Brito
UGEI
Faculdade de Engenharia
Universidade do Porto
Rua Dr. Roberto Frias; P-4200-465
Porto, PORTUGAL
E-mail: braganca@fe.up.pt
E-mail: acbrito@fe.up.pt

KEYWORDS
Simulation, Modelling, Manufacturing, ABMS.

ABSTRACT

The performance of a company depends on its strategic decisions but also on the efficiency of how those decisions are implemented by all of its sectors. In any company there are different sectors or groups, probably physically distributed, with different levels of autonomy, each one controlling local resources and interacting with the others. Many companies use simulation software to analyze critical sectors based in a number of scenarios, in order to estimate the efficiency of the different configurations.

The process of simulate alternative solutions can be time consuming. Depending on the software used it may be necessary to generate new simulation models for each solution, develop new applications or change existing ones.

This paper describes a simulation framework where each element is modelled using the discrete-event approach and interoperability between elements is achieved by the combination of web services and Agent-based modelling and simulation (ABMS).

The use of ABMS in this framework provides an easier way to evaluate alternative solutions. Each sector or group is an agent that can be added or removed from the model to build different configurations. A multi-hierarchical simulation executive is proposed in order to enable the use of different time intervals in each level. This increases the autonomy of the simulation agents and promotes the use of parallel or distributed solutions.

INTRODUCTION

A simple definition of Simulation is doing experimentation with a model that represents a system. Systems can be modelled as scaled models or as conceptual models (Teixeira 2006). A conceptual model can be implemented as continuous, discrete or even as partial discrete and partial continuous.

According to the implementation strategy, simulation can be classified as Continuous, Monte Carlo, discrete-event, hybrid, or agent-based (White and Ingalls 2009).

Computer simulation is widely used to analyze business activities in order to improve their efficiency. Each business has its own characteristics, but we can identify some common ones (White and Ingalls 2009):

- Multiple organisations involved in the process
- Physically distributed
- Different groups within organisations with different levels of autonomy
- Concurrency and interrelated tasks
- Dynamic and unpredictable tasks

The discrete-event simulation is the most used technique in parallel and distributed processing applications area (Nicol and Liu 2002).

Using a traditional discrete-event simulation approach the model is implemented in a single application. Modifying one of its elements requires the application to be changed, which can have impact in the other elements or in the logic associated with the business processes.

A business process is composed of a number of customer-supplier functional activities and tasks that need to be executed in a controlled and ordered way (N. R. Jennings et al. 2000). The customers and suppliers can be modelled in independent processes if properly synchronized. Furthermore, in most organisations, internal resources are grouped into business units with some autonomy that can also be modelled in sub-processes, synchronized inside the organization.

The synchronization schema can be synchronous or asynchronous. Synchronous simulation is characterized by the existence of one global clock and one event list queue, not appropriate to parallel or distributed situations. In asynchronous simulation each process maintains its own local clock and event list queue that are synchronized according to a conservative or optimistic mechanism. The conservative mechanism ensures that no message arrives in a time that is greater than the current time of the local clock. The optimistic technique must implement a roll back mechanism if the message arrives in a time that is less than the current time of the local clock. In this case, it is necessary to rewind the simulation to the time when the
message is received (Nicol and Liu 2002), (Niewiadomska-Szynkiewicz, Zmuda, and Malinowski 2003).

AGENT-BASED MODELING AND SIMULATION

ABMS (Agent-Based Modelling and Simulation) also known by ABM (Agent-Based Modeling), ABS (Agent-Based Systems or Simulation), or IBM (Individual-Based Modeling) is a new approach for simulation using agents to model human, social and organizational behaviour and individual decision-making (Macal and North 2009).

The use of the ABMS in the simulation process makes the task of defining alternative configurations more flexible because an agent can be changed or new agents can be added to the model without interfering with existing agents. The term Agent in the context of AMBS is not consensual and is the subject of many discussions. One definition that many authors tend to agree with (Macal and North 2009) (N Jennings 2000) is the following one:

An agent is an encapsulated computer system that is situated in some environment and that is capable of flexible, autonomous action in that environment in order to meet its design objectives.

The most important properties of the agents are:

- Autonomous and self-directed. They can be independent and can interact with other agents.
- Self-contained. They are identifiable with characteristics, attributes, behaviours and decision capability.
- Interactive. They have capabilities to communicate with other agents.
- Space limited. The interaction with other agents may be limited in space and time.
- Goals. They may have explicit goals like objectives to maximize or as criteria against which to assess the effectiveness of its decision and actions
- Ability to learn. In some situations an agent can learn as they react and/or interact with other agents.
- Recourse attributes. They have a diversity of resources that must control.

The term simulation in ABMS means that the agent performs its task over time as an entity in a simulation system.

The ABMS concept can be used in a similar way in different business processes modelling them as black boxes (see Figure 1). For example, in a model for simulating a supply chain system, the customer-supplier exchange activity of products and commercial messages may be represented as agents (Teixeira and Brito 2003).

computer, it must be ensured that all agents are properly synchronized over time. It also must be ensured that the events are processed in a correct order, as in a single simulation process (Fujimoto 2001).

SIMULATION MODEL FRAMEWORK

A simplified model of an industrial business environment can be described as shown in Figure 2. Interaction exists between the client and the factory, and between the factory and the supplier. Messages exchanged between the client and the factory are quite similar to the messages exchanged between the factory and the supplier.

The client and factory sends:

(i) messages to ask the list of products or services and supply conditions
(ii) trading messages
(iii) messages with orders.

The factory and supplier sends:

(i) messages with the list of products or services and supply conditions
(ii) trading messages
(iii) messages with product or service when the order is completed.

The simplified model a traditional discrete-event simulation system can be used to simulate each agent and to simulate the server communication manager, which controls the communications and event scheduling of the different agents.

![Figure 2: Simplified Business Model](image)

In a more complex model there are several clients and suppliers. Using the same type of agents of the simplified model and a communication manager server, it is possible to create an environment as shown in Figure 3.

![Figure 3: Independent Agents](image)
If the objective of the simulation is to improve the performance of a company it may not be necessary to model with detail all the clients and suppliers. The company external agents may be grouped by type and simulated as a whole, since they are not the main concern of the study.

The model can be developed varying from a simple to a complete representation of the company. Probably, neither of them will be the best choice, a simple model can produce poor results and a complete model can be time consuming in modelling, running and analysing the results.

One first simulation attempt may be a simple model of different independent sectors of the company modelled as agents, as represented in Figure 4. Almost every sector can be modelled as a group of agents with different levels of detail.

![Figure 4: Simple Model](image)

New simulations can be run by simply replacing one agent by a group of agents, within a nested level, without changing the rest of the model, as shown in Figure 5. Figure 5 shows the model of the factory with two agents in Sector A, instead of just one, increasing the degree of simulation detail. The factory sends and receives messages to/from clients and suppliers independently of the degree of detail used in the model (Figures 4 and 5). The factory and each of the sectors can be simulated in a similar way and this concept can be further extended in a multi-hierarchy of agents.

![Figure 5: Detailed Model](image)

Using this recursive approach it is possible to model a system with the required level of detail. It is possible to start with a high level solution and gradually increase the model detail in critical sectors.

**SCHEDULE**

The agents in the system must be synchronized in time when they send or receive a message. The simulation time of each individual agent can advance as long as there are no interactions with others agents and then pause, until all agents reach the same unit of time.

According to the way the agents communicate with each other the following types can be defined:
- A - Send and receive messages at fixed intervals
- B - Send messages at any time, but receives at fixed intervals
- C - Send and receive messages at any time

In synchronization the critical issue is receiving the messages. When an agent sends a message the time associated is the time of its internal clock. If the receiver agent is of type C when it receives a message the internal clock must be in the same time unit of the agent sending the message. If the receiver is of type B, the internal clock time must be greater than the time of the message that was send. The agent of type A is a particular case of the type B where sending messages are restricted to fixed intervals of time.

If all the agents in the system are of type B and the time interval of synchronization is multiple of the smaller time interval, it is easy to find a solution. Figure 6 shows two agents of type B, where the synchronization time is not aligned during the simulation. In this case the agent i, sends two messages (SM_{i1} and SM_{i2}) but the agent j, only receives and reply in the time t3. The messages sent by the agent j in the time 1 and 2 will only be processed by the agent i at the time 3.

![Figure 6: Receives Only in Fixed Periods](image)
synchronize is only necessary at the time when an agent is able to receive a message.

The Server Manager controls the main simulation Clock, stores the information of all the agents in the system and is the responsible for the communication between agents.

Figure 7: Agent Type B and C

**ARCHITECTURE**

In simulation history the use of programming languages has changed according to the software and computational power evolution. In the past, almost every language has been used, such as TURBOPASCAL, FORTRAN, SQPC, C++, SOAR, Z, DYNAMO, JAVA and other dedicated languages (Gilbert and Bankes 2002) (Teixeira 2006).

In the last decades the most used simulation tools were developed in C++, Java and dedicated languages. The advantages and disadvantages of those platforms like MASON, NetLogo, Repast, Swarm, can be found in the literature (Berryman 2008) (Railsback, Lytinen, and Jackson 2006) (Gilbert and Bankes 2002) (Castle 2006) (Tobias and Hofmann 2004). The main advantage of using C++ is speed. The main advantage of Java is the great number of Free Java-libraries that can be found.

The prototype presented here was developed in C# and uses Web Service concepts. C# is a simple object-oriented language and powerful when combined with the .NET Framework. Visual C# 2008 enables the creation of Windows applications, Web services, database tools etc. that can be compared with the functionality found in C++ and Java.

To develop an Agent-based modelling and simulation (ABMS) system, we need to develop the logic of each Agent (AGi), the Server Manager (SM) responsible for the communication and synchronization between agents, the Web Service (WS) to store information of the agents and simulations data, the Configuration Model (CM), the Display Model (DM) to display the progress of the simulation and a Statistical Output (SO) to interpret the output of the simulation (see Figure 8).

The Agents in this prototype are developed in C#, but can be developed in any language that supports TCP/IP, in order to communicate with the Server Manager and exchange XML messages with the others Agents.

Figure 8: Architecture

Figure 9 shows a simplified flowchart of the system. After the initial configuration, provided by the configuration Model, the Server manager starts and waits for the connection of all agents. If the connection is accepted the agent sends additional information to the server that is stored by the web service in the database to be used later on during the simulation and by CM, DM and SO modules.

When all the agents are connected the SM sends an initial synchronization message to all agents so they can start the simulation.

The synchronization messages are treated by the agent and by the server manager as usual events in a discrete-event simulator.

Figure 9: Simulation executive

The Event List of the agent contains three types of events

- Internal events that will be executed within the agent
- External events that will be send to the destination agent by the server Manager
• Synchronization events that will be send to the Server Manager

The Server Manager contains the information related to the synchronization time of all agents.

CONCLUSIONS

The simulation using the ABMS technique seems to be an adequate solution in modern business to analyse complex problems. The ability to model a system in components with a different level of detail, allows a more flexible and consistent analyses. The autonomy of some sectors of a company allows developing independent agents that can be reutilized in other studies. The proposed framework of agents can be useful because it allows the developing of models and the running of the simulation to be carried out physically distributed over a computer network. This can accelerate the developing task and be an incentive to collaborative work. An additional advantage can be the participation of local human resources that can be involved in the developing of the models that concerns their specific sectors.

The performance of the framework implementation depends on the number of interactions and information exchange between the agents. The framework must be implemented with the agent business affinity concept to minimize the number of interactions. Each agent only interacts with business related agents and not with all the agents in the system.

In future work it is intended to develop further the prototype, including more complex business components and semantic language definition for exchanging XML business messages.

REFERENCES


Castle, CIE. 2006. “Principles and concepts of agent-based modelling for developing geospatial simulations.” Centre for Advanced Spatial Analysis (University 44 (0).


doi:10.1080/088395100403379.


AUTHOR BIOGRAPHIES

ANTÓNIO CARVALHO BRITO was born in Porto in 1958, and went to Universidade do Porto, where he studied mechanical engineering and obtained his degree in 1981, staying in the same institution as Lecturer where he did his M.S. thesis in 1985. He obtained his Ph.D. in Simulation from the School of Management at the University of Cranfield (U.K.) in 1993. He is Assistant Professor at Universidade do Porto since 1993, where he leads research in the field of Visual Interactive Simulation. His e-mail address is: acbrito@fe.up.pt

CARLOS BRAGANÇA DE OLIVEIRA was born in Vila de Cucujães (Portugal) in 1958, and went to Universidade do Porto, where he studied mechanical engineering and obtained his degree in 1981, and his M.S. thesis in 1987 staying in the same institution as researcher until 1992. He was teacher in several Universities and is Invited Assistant Professor at Universidade do Porto since 2007, where he do research in the field of Agent Simulation. His e-mail address is: braganca@fe.up.pt
CARTOGRAPHY OF MULTI-AGENT MODEL PARAMETER SPACE THROUGH A REACTIVE DICOTOMOUS APPROACH

Patrick Taillandier
MTG Lab. UMR-IDEE 6228
1. rue Thomas Becket,
Mont-Saint-Aignan, France
E-mail: patrick.taillandier@univ-rouen.fr

Frédéric Amblard
IRIT - Université Toulouse 1 Capitole
2, rue du Doyen Gabriel Marty
Toulouse, France
E-mail: frederic.amblard@univ-tlse1.fr

KEYWORDS
Agent-based Simulation, Model Parameter Space
Cartography, Reactive Exploration, Dichotomous Approach

ABSTRACT

Parameter space exploration is a key issue in agent-based modeling. Many approaches were proposed concerning the optimization of a specific output of the model, but rare are the ones that aim at making the map of the parameter space. Yet, this map can bring very important information about a model. In this paper, we propose a new approach dedicated to this map making. Our approach is based on a reactive dichotomy of the parameter space following a criterion and on the use of random decompositions. We present two experiments that show that our approach allows, with the same number of simulations, to make a more relevant map than a uniform splitting.

INTRODUCTION

The agent-based modeling is now widely used to study complex systems. Its ability to represent several levels of interaction along a detailed environment representation favored such a development. There are nowadays numerous tools to help modelers to develop agent-based models. However analyzing such models can be very complex. Indeed, agent-based models can bring into plays numerous parameters that can each have an impact on the global dynamic of the system. Moreover, the stochastic nature of the agent-based models makes their analysis even more difficult.

The problem of the parameter space exploration is a classic problem in simulations. If numerous approaches were already proposed (e.g., (Gramacy et al. 2004; Lee et al. 2006)), very few specifically concern the agent-based simulations. Indeed, agent-based models are generally characterized by a large number of parameters that have a deep impact on the global dynamics of the system: a small modification of the value of one single parameter can lead to a radical modification of the dynamics of the whole system. Moreover, carrying out a simulation of an agent-based model is often very time-consuming. It is thus very important to limit as much as possible the number of simulations carried out.

Most of the existing approaches dedicated to agent-based models search to find the parameter values that allow to optimize a given fitness function (e.g., (Brueckner and Parunak 2003; Rogers and Tessin 2004; Calvez and Hutzler 2007)). While these studies are important for some application contexts (e.g., calibration of a model), they usually give rather few information on the system dynamics. In order to get a better view of the system dynamics, we propose to analyze the whole parameter space and not only to focus on values that optimizes a given output. The goal is to give the best approximation of the studied outputs according to the whole parameter space, i.e., to make a map of the model parameter space, while carrying out as few simulations as possible.

The mostly used approach to build such a map consists in splitting the parameter space into a uniform grid and to compute for each cell a representative output value. The problem of this approach comes from the uniform size of the cells: whatever the interest of the area covered by the cell, its size will remain the same. Thus, there is a risk of too many simulations that will be carried out in uninteresting areas (typically, areas where there are no variations of the output values), and too few in interesting ones (areas with important variations of the output values). Another problem comes from the size chosen for the cells: if too big, some important properties of the system dynamic can be missed; if too small, the computation times will be very high.

In order to tackle this issue, we propose a new approach based on a reactive dichotomy of the parameter space and on the use of random splitting. The next section describes the approach.

APPROACH PROPOSED

As mentioned in the previous section, our parameter space exploration approach is based on a reactive decomposition of the parameter space. It is based on the algorithm proposed in (Munos and Moore 2002), which is dedicated to the state abstraction for continuous time and space, deterministic dynamic control problems (e.g., reinforcement learning).

The principle of the approach is to start from a rough uniform grid and to split, in an iterative way, the most “promising” cells into sub-cells until a stopping criterion is checked. The choice of the most “promising” cells is made...
according to a criterion that characterized the quantity of variations of the output inside this cell. Higher the variations, more the cell has a chance to be split. The general algorithm is the following:

<table>
<thead>
<tr>
<th>Input values: S, K and R : integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialization stage: decomposition of the parameter space into a grid</td>
</tr>
<tr>
<td>While the number of sets of parameter values for which the output values was computed &lt; S:</td>
</tr>
<tr>
<td>Selection of the K most promising cells</td>
</tr>
<tr>
<td>For each selected cell:</td>
</tr>
<tr>
<td>Splitting of the cell</td>
</tr>
<tr>
<td>End for</td>
</tr>
<tr>
<td>For R points built randomly</td>
</tr>
<tr>
<td>Splitting of the cell that contains the point</td>
</tr>
<tr>
<td>End for</td>
</tr>
<tr>
<td>End while</td>
</tr>
</tbody>
</table>

Three parameters have to be defined:
- S: number of sets of parameter values for which the output value was computed during the exploration
- K: number of “promising” cells that will be split at each iteration
- R: number of random points that will be drawn at each iteration

Our approach is composed of three important stages:
- Building of the initial grid.
- Selection and splitting of the most promising cells.
- Drawing of random points and splitting of the cells containing these points.

These stages are described in the following sections.

Initialization

The first step of our approach consists in splitting the model parameter space into a uniform grid. Figure 1 gives an example of initial grid that can be built for a parameter space composed of two parameters x and y which values are real between respectively [0,1] and [0,6].

![Figure 1: Example of grid of dimension 2 (x,y)](image)

Each cell will be composed of $2^N$ points with N, the dimension of the model parameter space. Each point corresponds to a vector of parameter value. Figure 2 gives an example of a grid for a parameter space of dimension 3.

For each point, the value of the studied output will be computed. In order to cope with the stochastic nature of the agent-based models, we propose, to determine the output value of a point, to carry out several simulations and to compute their average output value. We propose as well to define the global output value of a cell as the average output value of each point composing it. Thus, for example, in the case of the cell shown Figure 2, the global output value will be equal to the average output value of $P_1$, $P_2$, $P_3$, $P_4$, $P_5$, $P_6$, $P_7$ and $P_8$.

![Figure 2: Cell of dimension 3 (x,y,z)](image)

Selection and splitting of the most promising cells

This stage consists in selecting and splitting the most “promising” cells, i.e., the ones for which the value of the output studied varies the most.

Munos and Moore (Munos and Moore 2002) present several criteria that can be used for the selection. In our context, where the goal is to give a global view of the model parameter space and not only to focus on optimal parameter values, the two most relevant ones are:

- **Average corner-value difference (ACVD):** average of the absolute difference of the output value between two points of the cell along a direction.
  - For example, in the case of the cell shown in Figure 1, the value of this criterion for the direction x, y and z are computed as follows:
    
    $\text{Val}_{ACVD}^{x} = \frac{1}{4} \left( |\text{val}(P_1) - \text{val}(P_2)| + |\text{val}(P_2) - \text{val}(P_3)| + |\text{val}(P_3) - \text{val}(P_4)| + |\text{val}(P_4) - \text{val}(P_1)| \right)$
    
    $\text{Val}_{ACVD}^{y} = \frac{1}{4} \left( |\text{val}(P_1) - \text{val}(P_5)| + |\text{val}(P_5) - \text{val}(P_6)| + |\text{val}(P_6) - \text{val}(P_7)| + |\text{val}(P_7) - \text{val}(P_1)| \right)$
    
    $\text{Val}_{ACVD}^{z} = \frac{1}{4} \left( |\text{val}(P_1) - \text{val}(P_9)| + |\text{val}(P_9) - \text{val}(P_{10})| + |\text{val}(P_{10}) - \text{val}(P_{11})| + |\text{val}(P_{11}) - \text{val}(P_1)| \right)$

- **Value non-linearity (VNL):** variance of the absolute increase of the output values between two points of the cell along a direction. The computation of this criterion is similar than the one based on the average, except that here the variance is used instead of the average.
Thus, for each cell, the criterion value will be computed for each direction, but only the direction that maximizes the criterion value will be kept.

Once each cell characterized by a criterion value and a direction, we propose to select the \( K \) cells that maximize the criterion value. The main interest to select \( K \) cells at the same iteration instead of just one is to allow the distribution of simulations. Indeed, each splitting can be computed independently on different processors or computers to minimize the computational time of the parameter space exploration process. Several frameworks specially dedicated to this task were proposed (Riou et al. 2008; Abramson et al. 2009; Lorca et al. 2011).

Remarks that other selection methods such as selection by roulette, by tournament (see (Mitchell, 1996) for a description), and others, can be used at this step of our approach. The conclusion of the paper comes back on this point.

The \( K \) cells selected are then split in two sub-cells of same size along the direction chosen. Figure 3 shows the three possible decompositions for a cell of dimension 3.

![Figure 3: Possible decompositions for a cell of dimension 3 (x,y,z)](image)

**Drawing of random points and splitting of the cells containing these points.**

The initial decomposition of the model parameter space into a grid can have a deep impact on the final result. As the cell selection method we propose in the previous section is purely greedy, i.e., it selects only the cells with the highest criterion value, a risk is to miss very local behavior of the system dynamic. Figure 4 illustrates this problem: in this example, cell 3 that hides a very interesting local phenomenon has little chance to be selected as the variation of the output value between the points \( P_2 \) and \( P_3 \) is very small.

A solution to this issue consists in using a more precise initial decomposition (with smaller cells), but it will increase the number of simulations that will be carried out during the initializing and thus decrease the number of simulations dedicated to the next stages of the exploration process.

![Figure 4: Example of initial decomposition for a parameter space of dimension 1 (x)](image)

Another solution that we propose to apply in our approach is to draw random points in the parameter space to carry out random splitting of the cells.

Thus, \( R \) points will be drawn in the model parameter space. For each point, the cell containing this point will be split according to this point in two sub-cells along a random direction. Figure 5 shows the three possible decompositions for a cell of dimension 3 according to a given point.

![Figure 5: Possible random decompositions for a cell of dimension 3 (x,y,z)](image)

\( R \) is a parameter of our approach. If \( R = 0 \), the approach will be fully greedy. A high value of \( R \) will favor the diversification of the search process over its intensification on the “interesting” cells detected.

The next section presents two experiments we carried out to evaluate our approach.

**EXPERIMENTS**

In order to test our approach, we propose two case-studies. Each of these case-studies consists in comparing the results obtained with a uniform decomposition of the model parameter space into a grid and with our approach for the same number of simulations. The goal is to evaluate the representativeness of the map built with these two approaches.

Our approach was implemented within the GAMA simulation platform (Taillard et al. 2010; GAMA 2011). This platform provides a complete modelling and simulation
development environment for building spatially explicit multi-agent simulations. Its main advantage comes from the simplicity to define a model with it. Indeed, GAMA provides a rich modelling language, GAML, for easily modelling agents and environments. Moreover, this platform provides a batch mode that allows to run sets of simulations with different parameter values. At last, this platform, which is implemented in Java, is easily extensible.

The first case-study we propose consists in testing our approach for a mathematical function. The goal is to have first evaluation of our approach for a very simple deterministic model.

The function chosen is:

$$f(x, y) = \frac{2 \cdot x^2 + y^2}{1 + x^2}$$

We propose to make the map of this function for:

- $x \in [-10,10]$
- $y \in [-10,10]$

Figure 6: a) Precise map of the considered function; b) Approximate map of the considered function obtained with a uniform grid (20x20); c) Approximate map of the considered function obtained with our approach (initial grid 10x10, $K = 20$, $R = 2$, $S = 441$)

Figure 6a shows the precise map of this function. The darker, the higher the value of the function is. Figure 6b shows the map obtained with a uniform decomposition (grid $20 \times 20$, thus 441 simulations) and Figure 6c with our approach (we used the “Average corner-value difference” criterion, an initial grid of $10 \times 10$, $K = 20$, $R = 2$, $S = 441$). As shown by the figures, our approach allowed to make a more relevant map than the uniform decomposition. Indeed, the map made by our approach presents a more precise frontier between the black and the gray areas. Moreover, it allows to visualize the white area (minimum of the function) at the center of the map.

The second case-study we propose consists in testing our approach for a classic model: the ant foraging model (Resnick 2000). The goal is to have first evaluation of our approach for a stochastic model.

In this model, a colony of ants forages for food. When an ant finds a piece of food, it carries the food back to the nest and drops pheromone as it moves. When an ant smells the pheromone, it follows the pheromone.

Figure 7: a) Precise map of the ant model; b) Approximate map of the ant model obtained with a uniform grid (10x10); c) Approximate map of the ant model obtained with our approach (initial grid 3x3, $K = 5$, $R = 1$, $S = 121$)

Two parameters are defined in this model:

- diffusion rate $\in [0,1]$: diffusion rate of the pheromone
• evaporation rate $\in [0,1]$: evaporation rate of the pheromone

We propose to study the influence of the diffusion and evaporation rates on the time necessary for the ant colony to carry all the food in the nest. We then make the map of the function $Time(diffusion\_rate, evaporation\_rate)$. As the model is stochastic (the movement of the ants, when no pheromone is detected, is random), we propose to carry 5 simulations and to compute the average time obtained to determine the global time of a set of parameter values.

Figure 7a shows the precise map obtained for the ant model. The darker, the higher is the time necessary to carry all the food to the nest. Figure 7b shows the map obtained with a uniform decomposition (grid $10 \times 10$, thus $121 \times 5$ simulations) and Figure 7c with our approach (we used the “Average corner-value difference” criterion, an initial grid of $3 \times 3$, $K = 5$, $R = 1$, $S = 121$). As shown by the figures, our approach allowed to make a more relevant map than the uniform decomposition. Indeed, the map made by our approach presents a far more precise frontier between the black and the gray areas in the right part of the map.

CONCLUSION AND FUTURE WORK

In this paper, we proposed a new approach to make the map of an agent-based model parameter space. Our approach is based on a reactive decomposition of the parameter space. We presented two case studies that show that our approach can allow to obtain, with the same number of simulations carried out, a more representative map than with a uniform decomposition.

The first experiments carried out concerned simple models. A first perspective is to apply our approach in the context of real models integrating a great number of parameters.

In the experiments, we used a simple criterion for the choice of the most “promising” cells. We would like to study and test more complex criteria, in particular criteria that take into account expert knowledge. Indeed, domain-experts that study a complex system often have knowledge about its dynamic and the impact of the different parameters. We propose to use this knowledge to define more relevant criteria.

As mentioned in a previous section, we would like to test other cell selection strategies. In particular, strategies that are not purely greedy. We can base these strategies on classic strategies used by optimization algorithms such as Simulated Annealing (Kirkpatrick 1983).

At last, we would like to work on the formalization of the output function. Indeed, in some context, the formalization of what the modeller wants to observe can be complex: for example, formalising the concept of “line of ants” in the ant foraging model is not straightforward. Thus, we propose to develop a new method based on the work we carried out concerning the definition of objective functions for optimisation problems (Taillard and Gaffuri 2009). This approach is based on man-machine dialogue: several solutions (that can be simulation results) are presented to the user and are commented by him/her. The analysis of the comments allows to learn a function that translates the user needs.

REFERENCES


AUTHOR BIOGRAPHY

PATRICK TAILLANDIER graduated in artificial intelligence from the University of Lyon 1 (France) in 2005 and received his PhD degree in 2008 at the University Paris Est (COGIT lab – IGN). After working two years for the MSI research team (IFI – Hanoi, Vietnam) and one year for the SMAC research team (IRIT – Toulouse, France), he was recruited in 2011 as an associate professor by the University of Rouen.
SOFTWARE DEVELOPMENT TOOLS
Basic Constructions for the Definition of the Software Development Process with Formal Methods

Jan Kožusznik, Svatopluk Štolfá
Department of Computer Science
VSB - Technical University of Ostrava
17.listopadu 15, Ostrava-Poruba
Czech Republic
E-mail: jan.kozusznik, svatopluk.stolfa @vsb.cz

KEYWORDS
model, software development process, set, relation, knowledgebase, ontology

ABSTRACT: The formal specification of the software development process is the necessary step for the knowledgebase support of the process. The problem is the selection of the appropriate specification language. One way how to support a selection of the formal specification language is the knowledge what type of constructions should be supported. In this paper we focus on the modeling constructions that are used during the modeling of the software development process. These constructions should be supported then by the selected formal language.

INTRODUCTION

Modeling the process is always driven by the specific goal. The goal has to be selected before the modeling is performed, because the modeling approach depends on the desired point of view and that point of view depends on the goal. Our intention is to develop a modeling approach that can be used, at least at the beginning of the modeling, without the knowledge of the real goal. We have decided to use an ontology based approach that could fulfill some necessary properties: iterativeness – the model can be modeled from the abstract viewpoint and then refined; transformation between different types of model approaches; integration of approaches – avoiding duplicities when modeling by one approach and then switching to another.
We have already discussed benefits of knowledge modeling (Kožusznik et al. 2011). In this paper, we present which basic model constructions are used for the modeling of the static structure of the software process. Description of the dynamic property of the process will be inspected in the future – we are planning to integrate it with some process algebra (Milner 1999; Hoare 1985) or another approach as is Petri-net (Petri 1962). Integration of a different formal specification languages is described in (Mossakowski, Maeder, and Luttich 2007)

The paper is organized as follows: Section 1 contains State-of-the-art; Section 2 describes the reference modeling example and Section 3 introduces the concept of modeling constructions. Finally, concluding Section 4 provides a summary and discusses future research.

STATE-OF-THE-ART

Modeling is often used in other disciplines and it is beneficial for the software development process. The term model means a representation of one system – modeled system – by another system. The modeled system often comes from reality or another artificial complex system and the model is its simplification – abstraction. Three functions of abstraction for modeling during database system development are defined (Smith and Smith 1977; Machado, Caetano Traina, and Araujo 2000) – see Figure 1.

There exist many modeling techniques for process modeling but the benefits of the knowledge-based approach to modeling and simulation (KBS) compared to a Discrete-Event Simulation and System Dynamics are discussed in (Scacchi 1999).

EXAMPLE OF SOFTWARE PROCESS

Our research of software development process modeling is tested on generally used example defined in (Kellner et al. 1990) – ISPW-6. Many authors use this example for the comparison of different modeling approaches - (Raffo 1996; Martin 2002) etc. There are defined main activities: Develop Change and test Unit, Schedule and Assign Tasks, Modify Design, Review Design, Modify Code, Modify Test Plans, Modify Unit Test Package, Test Unit, Monitor Progress. Activity “Develop Change and Test Unit” represent the whole process and is composed of others activities. Every activity of this process is also described by input and output work products; a description; a responsible role and constraints.

Figure 1 Functions of abstraction
MODELING CONSTRUCTIONS

Every model in every modeling language must be described by some constructions. These constructions typically should include explicit declaration of groups of elements, description of group relations, some specific conditions, definition of individuals etc. Every language has its own grammar and expression capability. Therefore, the description of necessary constructions could vary from the simple definition to the complicated description of the same construction element.

The selection of necessary modeling constructions always depends on the domain that we would like to model and on the goal of the modeling. The domain and the goal specify the point of view that we use for the modeling of the reality.

Our intention is to model the software process. Since the static structure is the most common part for all points of view of the processes (other points of view use the elements definitions etc.), we have tried to define constructions that are necessary and simplify the modeling of the static process view. These constructions were derived and defined by the modeling of the process static structure in different languages. We were able to define five basic categories of constructions that are, according to us, useful for the modeling of the static structure of the process.

Five basic categories are:

- Definition of separated theories (models, ontologies);
- Definition of classes (sets);
- Definition of relations;
- Constraint definition;
- Macro or user function definition

Discussion for every category follows.

Definition of separated specifications

Specification represents, in our sense, defined model by specified elements, constraints and relations between them. We don’t distinguish between theory and ontology, metamodel or model.

We have discussed the importance of separate specifications existence in (Koziusznik and Svatopluk 2011). Possibility of the importing of other specification is also involved. It enables us to define separate meta-model and use it for the definition of many independent models afterwards. Every rule in imported specification should be valid in importing theory. On the other hand, rules in the independent specification can be in conflict. This construction is presented in OWL or in Common algebraic specification language (CASL). Example in CASL looks like:

```plaintext
// following theory defines meta-model of software process; it contains definition of required basic
// constructions
spec process_domain_metamodel
then
... // definition of this theory
end

// following theory represents model of example process - it is named ISPW-6 in literature (Raffo 1996)
```
ModifyDesign,
("This step involves the modifications of the design for the code ...")

Constraint definition

Some specific relations and classes have to satisfy some constraints. Example of such constraint could be the definition that every "WorkProduct" should be in the relation "WorkProduct.out" and with at least one "WorkUnit":

\[ WorkProduct.out \subseteq WorkProduct \times WorkUnit \]

\[ \forall d \in WorkProduct \Rightarrow (\exists r \in WorkUnit \land (d, r) \in WorkProduct.out) \]

Existential and general quantifiers and logical operators are used for such kind of constraints.

Macro or user function definition

Sometimes more complicated expressions are repetitively used to express relatively simple combinations of basic constructions. The constraint defined previously is an example. It can be required for another relations and it is unpleasant to define it in above presented way. We propose to use macros or user function definitions to simplify the expressions and enable comfortable construction and usage of complicated expressions:

\[ \text{someItemOnLeft} (\text{relation}) \equiv \text{relation} \in \text{Domain} \times \text{Range} \]

\[ \Rightarrow \forall d \in \text{Domain} \Rightarrow (\exists r \in \text{Range} \land (d, r) \in \text{relation}) \]

Then only this fact is defined during the modeling:

\[ \text{someItemOnLeft}(\text{WorkProduct.out}). \]

CONCLUSION

Presented paper explores required constructions used during the formal definition of the software development process. It was limited only for static structure of software process – dynamic structure will be examined in the future. The paper also includes overview of additional features of the modeling language. We would like to evaluate available formal languages and choose the most appropriate for formal representation of software development process in the future. This paper will be used as source for the inspection and measurement of the features.

Acknowledgements. This research has been supported by the internal grant agency of VSB-TU of Ostrava - SP2011/56 Knowledge approach to the modeling, simulation and visualization of software processes.

REFERENCES


System Dynamic Simulation Approach of Software Development Process in Haskell
Jan Kožusznik
Department of Computer Science
VSB - Technical University of Ostrava
17.listopadu 15, Ostrava-Poruba
Czech Republic
E-mail: {jan.kozusznik}@vsb.cz

KEYWORDS
System dynamics, Haskell, Aivika, software development process,

ABSTRACT
The paper inspects possibility of the programming language Haskell for a creation of a simulation model. A paradigm of the system dynamic simulation is used. It mainly presents our experiences with a creation of a model of a software process. The simulation of the process model should asset benefits of reusing implemented features during system development.

INTRODUCTION
The software process is quite specific (Raffo 1996) and it has been characterized as “the most complex endeavor humankind has ever attempted” (Brooks 1987). However, software process could be modeled formally (Curtis, Kellner, and Over 1992). A simulation is a specific objective for the modeling and it has been used in the last two decades for the software process (Abdel-Hamid and Madnick 1991). Simulation helps to achieve the optimal balance among quality, budget and duration (Rus, Collofello, and Lacey 1999). Simulation helps forecast and quantify process trajectories in respect to their actual performance (Scacchi 1999).
Continuous simulation System Dynamics (Raffo, Vandel, and Martin 1999; Ruiz, Ramos, and Toro 2004) is one of the leading simulation paradigms. It is controlled by continuous time and change of parameters in process is modeled as system of differential equations. This paper examines using of this paradigm for modeling of software product line engineering (Pohl, Böckle, and Linden 2010). Many mature simulation tools for system dynamics exists: ExtendSim (ExtendSim Simulation Software by Imagine That Inc), Vensim (Software |Vensim), Anylogic (Simulation Software Tool - AnyLogic), Powrsim (Powrsim Studio 8 | Powrsim Software). The tools are commercial and require additional costs, on the other hand. The aim of this paper is to describe our experience with environment based on programming language Haskell (The Haskell Programming Language) – library Aivika (HackageDB: aivika-0.1) that supports also discrete-event (Banks et al. 2009) and multi-agent (Gilbert 2007) simulation paradigm. It was originally implemented in F# but it is also ported into Haskell language – it is available as package on Haskell Platform. This environment is free, offers some specific features but is less comfortable. We will demonstrate possibilities on a simulation model of a specific software process. On the other hand, we would like to discuss the simulation environment and not the simulation model or its result.
This paper is organized as follows: section System dynamics provides brief introduction to simulation paradigm; section Simulated process describes software process that was modeled; section Basis of the simulation environment shows how construct simulation model in Haskell with Aivika package; following section depicts our experiences with simulation of the presented process; Conclusion summarizes positives and negatives of the simulation environment.

SYSTEM DYNAMICS
System dynamics model software process as a closed-loop version of a system (Figure 1). Input represents requirements, specifications and resources; otherwise output represents artifacts. The input is transformed to output and the operation is influenced by a controller that changes the behavior of the system.
System dynamic simulation approach was developed by Jay Wright Forrester – published in (Forrester 1961). It was adopted for the software process modeling purpose by Tarek Abdel-Hamid (Abdel-Hamid and Madnick 1991). This approach is also named continuous system modeling because the model is computed in respect to continuous time. It is the main difference from discrete-event modeling.
System dynamics model is composed from elements (adopted from (Madachy 2008)) defined in Table 1.

<table>
<thead>
<tr>
<th>Element</th>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td><img src="image" alt="Tasks" /></td>
<td>It represents an accumulation over time – also called a stock or state variable.</td>
</tr>
<tr>
<td>Source/Sink</td>
<td><img src="image" alt="Flow" /></td>
<td>It indicates that flows come from or go to somewhere external to the process.</td>
</tr>
</tbody>
</table>
Rate

It is also called flow and it effect changes in levels.

Auxiliary

Only label

It is converter of input to outputs, it helps elaborate the detail of level and rate structures.

Information Links

It is used to represent the information flow that could influence value of some auxiliary or rate element.

The system is modeled as a structure of levels connected by flows and information links. Flows between levels represent process entities that are conserved within flow chains. Information links only provide data from auxiliary, flow or level to another auxiliary or flow (Figure 2). Individual events are not tracked; tasks are modeled by levels and the system could be described by system differential equations.

![Figure 1 Closed-loop system](image1)

![Figure 2 Code construction](image2)

It has a very simple parallel to the physical system of water basins (level) connected with valued pipes (rates). Pipes could be controlled based on the level of other water basins or other computed factors (auxiliary). Information about control is represented by an information link. The amount of level in a specific time $T$ is equal to:

$$\int_{0}^{T} \left( \sum_{i \in \text{input}} f_i(t) - \sum_{o \in \text{output}} f_o(t) \right) dt$$

Functions $f_i(t)$ represent input flows to a level, while $f_o(t)$ represents output flows.

**SIMULATED PROCESS**

Inspected simulation process example demonstrates cost of reusing implemented features in system. It will be used as initial analysis for introduction of a product line development (Pohl, Böckle, and Linden 2010).

The simulation model contains two process branches:

- a first simulates development of requested features that were not developed yet;
- a second simulates reuse of requested features that are already developed.

Every implemented feature is integrated into library and should be reused (adapted) afterwards. Software for library management should be implemented before using this library – it depicts flow LibraryBuildingRT. An overview of the simulation model is (diagram of system dynamic approach) in the Figure 3.

A rate of requested features is constant in the model, but a probability that requested feature is already in library is changing. We use learning curve mentioned in (Madachy 2008):

$$(1 + \text{featureIntegratedToLib}) \log_2(1 - \text{slopeCoeff})$$

The coefficient `libraryFeature` contains a number of implemented features that could be reused. The coefficient `slopeCoeff` defines a slope for a curve of the probability function.

![Figure 3 An overview of modeled process](image3)

**BASIS OF THE SIMULATION ENVIRONMENT**

**Aivika model structure**

Created simulation models are created directly in Haskell programming language. Simulation constructions from library Aivika are used. This library is available as part of Haskell platform. A simple but comprehensive introduction to Haskell is available in (Lipovaca 2011) that is also available online for free (Learn You a Haskell for Great Good!).

The simplest version of model in Aivika has structure:

```haskell
sampleModel =
do
--computational steps
return totalFeatures
```
There are defined computational steps after keyword “do”. Specific constructions are discussed in following sections. Simulation model is performed by function that looks likes:

```haskell
specs = Specs { spcStartTime = 0, 
spcStopTime = 10000, 
spcDT = 0.5, 
spcMethod = Euler }
runModel = do a <- runDynamics reuseModel specs 
print a
```

This code prints array of values of a variable “TotalFeatures” during simulation. One item of array exists for one simulation step. Better result presentation will be presented later.

Variable “spec” contains parameters for simulation execution: start time, stop time, size of step and integration method.

It is also possible put more variables on the output. Similar construction is used:

```haskell
sampleModel = do 
--computational steps 
return ( sequence [time, totalFeatures] )
```

Output variable “a” – defined in previous function - contains array of result variables (array of arrays) now.

**Basic constructions**

There is no difference between rates and auxiliary variables in created models. A flow “RequestingRT” is defined as:

```haskell
let 
requestingRT = c_request / t_impl
```

“c_request” and “t_impl” are variables – coefficients. Construct of level is defined as:

```haskell
featuresForRequest <- newInteg n_requestedFeat
```

Number “n_requestedFeat” defines initial value of level in this construction. Term “newInteg” is something as constructor.

Input and output flows for specific level are defined by integration:

```haskell
integDiff implementedFeatures ( implementationRT - integrationIntoLibraryRT )
```

Term integDiff is special function, “ImplementedFeatures” is level and “ImplementationRT”, “IntegrationIntoLibraryRT” are flows.

If we want to use a value of some level for computing of an auxiliary variable, a rate or as return value we have to use function “integValue”:

```haskell
let 
totalFeatures = ( integValue implementedFeatures ) 
+ ( integValue featIntegratedToLib )
+ ( integValue adaptedFeatures )
---
return ( sequence [time, totalFeatures, integDiff totalEffort] )
```

**Structure of whole model looks like:**

```haskell
reuseModel = do 
--definition of levels 
featuresForRequest <- newInteg n_requestedFeat
--
let 
--definition of rates and auxiliaries 
requestingRT = c_request / t_impl
--
--integrations 
integDiff implementedFeatures 
( implementationRT - IntegrationIntoLibraryRT )
return ( sequence [time, totalFeatures, 
integValue totalEffort] )
```

**More complicated expressions**

It is also possible to define functions that encapsulate some general constructions. It increases level of abstraction during modeling. A modeling of constant flows that have zero value of rate if their input is equal or less to zero is typical common construction in system dynamics. Haskell offers possibility to create function that expresses it:

```haskell
limitedInpRt inputLevel rate = do 
act <- inputlevel 
if (act > 0) then rate else 0
```

**Definition of a flow named “ImplementationRT” looks:**

```haskell
implementationRT = 
limitedInpRt (newlyRequestedFeatures) (1/t_impl)
```

The flow is equal to \( \frac{1}{t_{impl}} \) iff the level “NewlyRequestedFeatures” is greater than 0. Otherwise, it is equal to zero.

**Result presentation**

Previously discussed model puts simulation result on standard console. On the other hand, it is common to use charts for result presentation or comparison. It exists Haskell package “Chart” that provides required capability (HackageDB: Chart-0.14). Basic charts are provided by functions in module “Graphics.Rendering.Chart.Simple”:

- `plotWindow` – creates line chart to output graphics window;
- `plotPDF` – creates line chart to the specified pdf file.

General syntax for drawing of chart with two curves is:

```haskell
plotWindow ar_x ar_y_1 “curve_1” ar_y_2 “curve_2”
```

Expression `ar_x` represents array of x values. Expressions `ar_y_1` respectively `ar_y_2` represents array of y-values for 1th respectively 2nd curve. Values “curve_1” and “curve_2” define captions of curves.

Described functions can be used for a presentation of results of a simulation model. A function that draws chart representing dependency of TotalFeatures and TotalEffort
base on time during simulation is (result chart is in the Figure 4):

```haskell
compute_1 = do
  a <- runDynamics reuseModel specs
  plotWindow
  (map (\x -> x!!0) a)
  (map (\x -> x!!1) a) "Total features"
  (map (\x -> x!!2) a) "Total effort"
```

Expression:
```
(map (\x -> x!!0) a)
```

extracts from results (represented by array of arrays) a single array that contains values located on index 0 in the array. It is necessary because structure of results is:
```haskell
[[val1, val2, ..., valn1],
 [val1, val2, ..., valn2],
 ...
 [val1, val2, ..., valn]]
```

Function “map” is standard Haskell function. It has on its input an array (second parameter) and a function (first parameter). It applies the function on every item of the input array and produces new array that contains result values of the function. The operator “!!” returns specific item from an array.

Previous expression is useful and we defined functions based on them:

```
ar !!? o = map (\x -> x!!o) ar
```

Displaying of results is much easier with that:

```haskell
comp = do
  a <- runDynamics plModel specs
  plotWindow (a!!0)
  (a!!1) "Total features"
  (a!!2) "Total effort"
```

EXPERIENCES WITH AIVIKA

Experiments with the presented simulation model mainly involved an efficiency comparison of the simulated process model (named reuse model) with process where no implemented feature is used (named simple model). Two separated processes were created for both of them. A dependency chart of a number of implemented features on time was created (Figure 5). Number of implemented features is sum of newly implemented features (integrated into library and also not integrated into library yet) and adapted features from library into developed system. Second interesting chart depicts dependency of a total effort on time (Figure 6).

Influence of input parameters on the result of model was also inspected. Interesting result is ratio between effort spent in reuse model and effort spent in simple model. Running of many simulations take its place and a benefit of a programming environment is used. More simulations with different input parameters will be executed automatically. A function for simulation execution with modified parameter is defined:

```
computation val = do
  a <- runDynamics1 (reuseModel
  simSpec{c_featForLibrary = val}) specs
  b <- runDynamics1 simpleModel specs
  return ((a!!2)/(b!!2))
```

Presented code is quite complex and use additional functions and construction. A function “runDynamics1” executes simulation but returns only one value – there is only last value of a total effort required. A construction `simSpec{c_featForLibrary = val}` modifies a value of a parameter “c_featForLibrary” in argument “simSpec”. The argument is instance of a complex structure that defines values of model input parameters. Model function “reuseModel” actually has one input argument of that types – it enable simply modify its input parameters. Simulation is executed for values of parameter c_featuresOfLibrary in interval (0,1) by a function:

```haskell
showEffortRatio = do
  let
```
arr_x = [0.05..1]
arr_y <- (sequence (map computation arr_x))
plotWindow arr_x arr_y "Ratio of total effort"

We omit detailed description of that block of the previous code – function “sequence” or definition of array. It is out of scope of the paper. Result chart is in the Figure 7. It shows if a parameter `featuresOfLibrary` is greater than 0.3 then reuse model is less effective than simple model. It means that reuse model is profitable if number of requested feature is greater than 30% in comparison to number of features required during development of software for library management.

![Figure 7 A ratio of a total effort](image)

CONCLUSION

The paper presented possibility of Haskell language and its library Aivika for system dynamics simulation. Constructions presented in this paper are simplified. Our constructed model enables: define priorities for an activity and solve simple resource allocation, simply change structure etc. However, presentation of the entire model is out of scope of the paper.

It demonstrates that is quite simple. Main advantage is that the environment is free. It also benefits from its ability to create more complex constructions and it also enables to customize a computation. It is suitable when more computations for different values of input parameters are required. Haskell also provide possibility to present result values in result charts.

On the other hand, this tool lacks possibility to visualize structure of a created model as other professional tools. During construction, we created overview of model in a graphical tool and then we implemented it. However, we would like to create tool that will visualize structure of process model defined by the presented environment in the future.

Acknowledgements. This research has been supported by the internal grant agency of VSB-TU of Ostrava - SP2011/56 Knowledge approach to the modeling, simulation and visualization of software processes.

REFERENCES


Forrester, Jay Wright. 1961. *Industrial Dynamics*; Pegasus Communications.


SIMULATION AS A DECISION SUPPORT TOOL IN MAINTENANCE FLOAT SYSTEMS – The Automatic Generation of Simulation Programs

Francisco Peito
Polytechnic Institute of Bragança
Higher School of Technology and Management
Industrial Management Dpt.
Campus de Santa Apolónia
5301-857 Bragança
Portugal
pires@ipb.pt

Guilherme Pereira
University of Minho
School of Engineering
Production and Systems Dpt.
Campus de Gualtar
4710 057 Braga
Portugal
gui@dps.uminho.pt

Armando Leitão
Polytechnic Institute of Bragança
Higher School of Technology and Management
Industrial Management Dpt.
Campus de Santa Apolónia
5301-857 Bragança
Portugal
afleitao@ipb.pt

Luís Dias
University of Minho
School of Engineering
Production and Systems Dpt.
Campus de Gualtar
4710 057 Braga
Portugal
lsd@dps.uminho.pt

KEYWORDS
Simulation, Discrete Event Simulation, Maintenance, Preventive Maintenance, Queueing Theory, Float Systems.

ABSTRACT
This paper is concerned with the use of simulation as a decision support tool in maintenance systems, specifically in MFS (Maintenance Float Systems). For this purpose and due to its high complexity, in this paper the authors propose a flexible way to develop typical MFS models, for any number of machines in the workstation, spare machines and maintenance crews. Arena® simulation language is used to understand a specific MFS, create the corresponding MFS model and analyze most common performance measures.

INTRODUCTION
According to (Pegden et al. 1990), simulation can be understood as the process of construction of a real system representative model, as well as an experimental process aiming to a better understanding of their behavior and to assess the impact of alternative operations strategies. Thus, simulation may also be considered as a decision support tool that allows to predict and to analyze the performance of complex systems and processes as they are in many real systems. In addition, with the use of simulation we acquired a capacity to forecast and to achieve quickly the importance of taking some decisions about the system under analysis. In some real systems like production areas, services such as transport companies, health service systems and factories, the main goal is to achieve high levels of competitiveness and operational availability. In this environment the need for equipment to work continuously is very likely in order to maintain high levels of productivity. This is why MFS has an important role on equipment breakdown and production stoppage has a high and direct impact on production process efficiency and, as a consequence, on their operational results. Therefore, maintenance control and optimization of equipment utilization become not only an important aspect for the mentioned reasons, but also for personnel security matters and to prevent negative environmental impact.

In general, preventive maintenance implementation increases equipment control and avoids unexpected stoppages. However, these maintenance actions could make maintenance costs too high for a required availability.

In production systems involving identical equipments such as Float Systems it is an advantage to integrate maintenance management with materials and human resources. The existence of spare equipment to replace machines that fail or need overhaul is an example of this type of situation. Then, direct and indirect costs due to equipment stoppage are minimized and the level of production or service requirements fulfilled. Although the existence of spare equipment is important to maintain the production process working keeping the number of spare equipment at an optimum level is recommended.

Mainly due to the non-existence of a specific simulator for the maintenance field, we had a great difficulty in choosing an appropriate simulation tool. However, (Dias et al. 2005) had a definite contribution as far as the simulation tool decision is concerned.

In fact, the choice of Arena® as a simulation language was based on the fact that its hierarchical structure offers different levels of flexibility, thus allowing the construction of extremely complex models, allied to a strong visual component (Kelton 2004; Pidd 1989; Dias 2006 and Pidd 1993).

Having referred the importance of studying MFS, the next section of this paper will focus on the literature review on analytical models, but also on some type of simulation metamodels for this type of maintenance systems.

The following section describes new developments on a previous simulation model towards flexibility. In fact, the model presented in (Peito et al. 2011) will gain the capacity to automatically generate a specific simulation program for each specific MFS desired. The program will then be adapted for specific situations with no need of further coding effort. In fact the new proposed tool is intended exclusively to give a response to a type-standard configuration of MFS. Nevertheless, within this type-standard configuration, the user could easily evaluate different strategies under different number of resources available (active machines, maintenance crews and spare machines). This way, the resulting MFS model aims to fill a gap in terms of computer solutions currently existing for this specific type of maintenance systems. Conclusions and Future Developments are the closing sections for this paper.

Conclusions and Future Developments are the closing sections for this paper.
To conclude this introduction, once more, we must refer that the proposed tool is intended exclusively to give a response to a type-standard configuration of MFS. Nevertheless, within this type-standard configuration, the user could easily evaluate different strategies under different values for the number of active machines, the number of maintenance crews and the number of spare machines This way, the resulting MFS model aims to fill a gap in terms of computer solutions currently existing for this specific type of maintenance systems.

RESEARCH BACKGROUND

As far as float systems maintenance models is concerned, (Lopes 2007) refers some studies where simulation has been used to produce results based on specified parameters. Due to the fact that these simulation models were only concerned with the input/output process, without dealing with what is happening during the simulation data process, some metamodels have emerged (Madu and Kuei 1992a; Madu and Kuei 1992b; Madu and Lyeu. 1994; Kuei and Madu 1994; Madu 1999; Alam et al. 2003). The metamodels express the input/output relationship through a regression equation. These metamodels can also be based on taguchi methods (Madu and Kuei 1992a; Kuei and Madu 1994) or neural networks (Chen and Tseng 2003). These maintenance system models were also recently treated on an analytical basis by (Gupta and Rao 1996; Gupta 1997; Zeng and Zhang 1997; Shankar and Sahani 2003; Lopes 2007). However, the model proposed by (Lopes 2007) is the only one that deals, simultaneously, with three variables: number of maintenance crews, number of spare equipments and time between overhauls, aiming the optimization of a system composed by $M$ active and identical equipments. Although this proposed model already involves a certain amount of complexity it may become even more complex by adding new variables and factors such as: a) time spent on spare equipment transportation, b) time spent on spare equipment installation; c) the introduction of more or different ways of estimating efficient measures; d) allowing the system to work discontinuously; e) speed or efficiency of the repair and revision actions; f) taking into account restrictions on workers timetable to perform the repair and revision actions; g) taking into account the workers scheduling to perform the repair and revision actions; h) taking into account the possibility of spare equipment failure, etc. Anyway these mentioned approaches would aim at ending up with MFS models very close to real system configurations. In fact, the literature review showed that most of the works published, involving either analytical or simulation models, concentrate on a single maintenance crew, or on a single machine on the workstation or even considering an unlimited maintenance capacity – thus overcoming the real system complexity and therefore not quite responding to the real problem as it exists.

As far as the model presented by (Lopes et al. 2005; Lopes et al. 2006; Lopes 2007) is concerned it is assumed that systems work continuously, its availability is not calculated and the system optimization is only based on the total maintenance cost per time unit. Moreover, it considers that the total system maintenance cost is the same without taking into account the number of machines unavailable, which in many real situations is not the best option. Finally the referred analytical model only allows that its failures occur under a Homogeneous Poisson process (HPP).

Another important aspect on the companies management strategic definition is to have their tasks correctly planned. To help this planning procedure it is important to know different indicators such as: machine availability, equipment performance and maintenance costs, among others. Therefore one should consider new factors that affect these float systems indicators: possibility of some machine failure, efficiency, repair time.

Moreover, when preventive maintenance policy is used, the time for individual replacement is smaller than time for group replacement. It means that the latter situation requires more machines on the process to be stopped, and also implies an increase on the number of maintenance crews for certain time periods.

In general, companies policy lies on using economic models to define their best strategies. Profits maximization or costs minimization are the most frequent goals used. However, strictly from the maintenance point of view, availability is frequently used as an efficient measure of the system performance, and sometimes more important than the cost based process.

DESCRIPTION OF THE MFS

Our model represents a typical Maintenance Float System and it is composed of a workstation, a maintenance centre with a set of maintenance crews to perform overhauls and repair actions and a set of spare machines (Fig.1). The workstation consists of a set of identical machines and the repair centre of a limited number of maintenance crews and a limited number of spare machines. However, the model we have adopted, being a typical MFS, presents certain specificities both as far as the philosophy of the maintenance waiting queues are concerned, and related to the management of the maintenance crews.

![Typical Maintenance Float System](image)

Fig. 1 – Typical Maintenance Float System

This model follows the one proposed and developed by (Lopes 2005; Lopes et al. 2006; Lopes et al. 2007), considering $M$ active machines, $R$ independent and identical spare machines and $L$ maintenance crews. The active machines considered operate continuously. Machines that fail are taken from the workstation and sent
to the maintenance park waiting queue, where they will be assisted according to arrival time. Machines that reach their optimal overhaul time are kept in service until the end of a period $T$ without failures. However they will be also kept on a virtual queue to overhaul. If the number of failed machines plus the number of machines requiring overhaul is lower than the number of maintenance crews available, machines are replaced and repaired according to FIFO (First In First Out) rule. Otherwise if it exceeds the number of maintenance crews, the machines will either be replaced (while there are spare machines available) or will be sent to the maintenance queue. The machines that complete a duration period $T$ or time between overhauls in operation without failures are maintained active in the workstation, where they wait to be assisted, and they are replaced when they are removed from the workstation, to be submitted to a preventive action. Its replacement is assured by the machine that leaves the maintenance centre in the immediately previous instant. If an active machine happens to fail it awaits for the accomplishment of an overhaul, then it will be immediately replaced, if a spare machine is available or as soon as it is available.

In this version of our model it is assumed that the $M$ active machines of the workstation have a constant failure rate while the model runs.

Time between failures are assumed as independent and identically distributed following an Exponential Distribution for all machines (failures occur under a Homogeneous Poisson Process). However, during a simulation run, this value could be adjusted based on failure time between overhauls. Obviously a smaller time between overhauls implies greater time between failures.

As far as time to overhaul and time to repair are concerned, we have assumed the Erlang-2 distribution, even though considering overhaul time significantly lower than the repair time.

For our MFS, the variables used are the following:

1. Number of active machines ($M$);
2. Number of maintenance crews ($L$);
3. Number of spare machines ($R$);
4. Machine-Overhauls rate ($\lambda_{rev}$)*;
5. Machine-Initial Failures rate ($\lambda_I$)*;
6. Crews-Repair rate ($\mu_{rep}$)*;
7. Crews-Overhaul rate ($\mu_{rev}$)*;
8. Failure cost ($C_f$);
9. Repair cost ($C_{rep}$);
10. Overhaul cost ($C_{ov}$);
11. Replacement cost ($C_r$);
12. Cost due to loss production ($C_l$);
13. Holding cost per time unit ($h$);
14. Labour cost per time unit ($k$);
15. Time to convey and install spare machine ($T_{Convo}$).

(* This variable can be adjusted during the simulation run.)

The developed simulation model for our MFS allows us to estimate the following global efficiency measures:

a) Average system availability ($AvgSAv$);
b) Total maintenance cost per time unit ($AvgTCu$);

However, some other performance measures are also estimated, such as:

c) Average number of missing machines at the workstation ($AvgM_{sp}$),
d) Average number of machines in the maintenance waiting queue ($AvgLq$),
e) Average waiting time in the maintenance waiting queue ($AvgWt$),
f) Average operating cycle time ($AvgD$),
g) Probability of existing 1 or more idle Machines ($Prob_{id}$),
h) Probability of the system being fully active ($Prob_{f}$),

and still, some individual efficiency measures per machine or maintenance crew, i.e.,

i) Utilization rate per machine,
j) Utilization rate per maintenance crew;
k) Number of overhauls and repair actions performed per maintenance crew;
l) Average availability per machine.

INCREASING FLEXIBILITY OF THE SIMULATION MODEL

The Arena® simulation language environment, used in the previous development (see details on Peito et al. 2011), has been now revisited, aiming to give flexibility to the previous model. The user, now, would be able to automatically generate a simulation program according to specific characteristics of the MFS, namely varying the number of active machines ($M$), the number of maintenance crews ($L$) and the number of spare machines ($R$). However, the steps towards the development of the previous simulation model were all kept and are presented in figure 2, for a better understanding of the simulation model developed.

Fig. 2 - Steps for simulation model development
Figures 3 and 4 explicit the global logical simulation model before and after gaining flexibility, underlining its different developed components:

1. Active machines (workstation);
2. Statistics 1 (Recording Machines $T_{up}$);
3. Maintenance queue;
4. Machines’ transportation (by the maintenance crews);
5. Spare machine request;
6. Maintenance center (set of maintenance Stations);
7. Release machines to the set of spare machines;
8. Statistics 2 (Recording Machines $T_{up}$ and $T_{down}$);
9. Spare machines (in the start of the system).

This logical model configuration choice was kept identical for the MFS (Fig. 3 and 4), providing again a clear global visualization of the undergoing operations and a great simplicity to make changes in the model. In fact the logical model, after increasing flexibility, will appear even more simplified – see figure 4. The implementation of Arena resource sets, the inclusion of indexed variables and data arrays and also a set of control variables, replacing previous Arena internal variables, have definitely contributed to a simplified model.

The components 1 and 9 after increasing flexibility (Fig. 6) include now a generation and control system for all repair and overhaul requests of all machines, this was not the case in the previous model (Fig. 5). For this control system to be effective, it would also be necessary to guarantee absolute independence of each type of request for every machine. For this purpose, a mechanism for attribute identification was developed. With this mechanism, it is now possible to identify the state of each machine and the occurrence of every type of machine request (failure or overhaul), at any instant – entity number and color (see figure 6, zone A).

In Fig. 6 (Zone B), a small change has occurred. In fact, some Arena Blocks have been replaced by Arena Modules. This way, planned changes to some parameters are now easy to implement once Arena shows data in a simple table format.

Finally, figure 6, Zone C shows four ReadWrite Arena modules, allowing the registration, in an excel worksheet, of the failure instants and the number of failures for each machine.

Fig. 5 - Generation and control system for repair and overhaul requests before increasing flexibility

Fig. 6 - Generation and control system for repair and overhaul requests after increasing flexibility

The maintenance waiting queue is defined through a synchronization of events between the component 3 and 4. In the component 4 (figures 7 and 8) there is a "control mechanism", which only allows a request to proceed if there is a free maintenance crew. Component 4 will now include the use of an Arena Resource Set for the maintenance crews, selecting the available maintenance crew that has the least number of services allocated.

The rules for the maintenance queue management were all kept unchanged. In fact, FIFO (First In First Out) is the rule for the maintenance queue management, except for the case when the total number of maintenance requests (overhauls plus repair actions) exceed the number of maintenance crews available – in this case, machines requiring repair action have priority over machines requiring overhauls.
Fig. 7 - Maintenance waiting queue before increasing flexibility

Fig. 8 - Maintenance waiting queue after increasing flexibility

Component 4 (figures 7 and 8) has also been changed and now includes an Assign Module in Zone D. Besides the identification of the maintenance crew and the machine transport state (for a spare machine or a failed machine or even a machine needing overhaul), this Module also updates the number of maintenance crews that are free.

Fig. 9 - Request and activation of spare machines before increasing flexibility

Fig. 10 - Request and activation of spare machines after increasing flexibility

In component 5 (figures 9 and 10) that performs the request of a spare machine, performed by a maintenance crew, there is only a small change in Zone E, that is related with the demand with one free available machine. Now the model includes a Search Block that searches for a free machine.

In component 6 (Fig. 11), the change is in the structure of the component. In fact, the discrete variables are now indexed discrete variables – this way, it is possible to individually save a set of performance indicators for both types of maintenance operations.

Fig. 11 – Identification and statistics of the states of the maintenance crew

In component 7, responsible for releasing machines under maintenance crew actions whenever they finish their work, either repairing or performing overhauls, all Release Modules have been replaced by a single Release Block – this was possible once now only a single indexed discrete variable is capable of saving all the information related to each machine.

Fig. 12 – Record statistics

Components 2 and 8 (Fig. 12) which are responsible to record fundamental statistical data to calculate adequate efficiency measures, do not suffered any change.

Fig. 13 – Data input area sample screenshot before increasing flexibility

Fig. 14 – Data input area sample screenshot after increasing flexibility
This work, making previous simulation model gaining flexibility, allows the user to get a simulation model for any Maintenance Float System desired – regardless the number of active machines, the number of maintenance crews and the numbers of spare machines. After inputting these three values (Zone F, figures 13 and 14), the user will instantly get the appropriate simulation model automatically generated.

**Fig. 15 - Variables and graphics control**

The simulation model here presented, incorporating analysis of usual performance measures, also drives its concern towards new efficiency measures, enabling new trends for the analysis and discussion of the best decisions as far as a specific Maintenance Float System is concerned. Nevertheless the authors are now aiming to the development of an advanced simulation model, incorporating still more flexibility. This target would be reached by developing and incorporating new modules in our simulation tool, in order to also incorporate maintenance systems where failure rates would also vary while the model runs, i.e., where a Non Homogeneous Poisson Process (NHPP) is present. These mentioned future developments also intend to potentiate the known capability of simulation to efficiently communicate with managers and decision makers, even if they are not simulation experts.

**REFERENCES**


**CONCLUSIONS AND FUTURE DEVELOPMENTS**

This new development of our simulation model for our Maintenance Float System presents:

- More flexibility

  This was the main challenge for the work presented in this paper. The automatic generation of simulation models, depending on the three main maintenance system variables – $M$, number of active machines; $L$, number of maintenance crews; $R$, number of reserve machines. In fact, the user would just have to introduce $M$, $L$ and $R$ and, instantly, he will get the adequate simulation model to run and experiment.

- More interactivity

  Now the user has the possibility to interact with the simulation model during each simulation run. In fact the user can now modify some parameters of the maintenance system under analysis and can, therefore, evaluate system behavior under different maintenance strategies.

- Better information

  This model now offers much better maintenance information. Indeed, the strong visual aspect offered by the developed model clarifies the actual process inside the system. This allows a better understanding of the different interactions in the model and of the simulation results.


Rodrigues, Guimarães and Carvalho, Valério (1984), "CAPS - ECSL, Experiência de modelagem e simulação aplicada a um sistema de elevadores", Relatório Técnico, Universidade do Minho.


AUTHOR BIOGRAPHY

FRANCISCO PEITO was born in 1966 in Macedo de Cavaleiros, Portugal. He graduated in Mechanical Engineering-Management of Production in Polytechnic Institute of Porto. He holds an MSc degree Industrial Maintenance in University of Porto. His main research interests are Simulation and Maintenance.

GUILHERME A B PEREIRA was born in 1961 in Porto, Portugal. He graduated in Industrial Engineering and Management at the University of Minho, Portugal. He holds an MSc degree in Operational Research and a PhD degree in Manufacturing and Mechanical Engineering from the University of Birmingham, UK. His main research interests are Operational Research and Simulation.

ARMANDO LEITÃO was born in 1958 in Porto, Portugal. He graduated in Mechanical Engineering at University of Porto, Portugal. He holds an MSc degree in Production Engineering and a PhD degree in Production Engineering from the University of Birmingham, UK. His main research interests are Reliability and Quality Maintenance.

LUÍS M S DIAS was born in 1970 in Vila Nova de Foz Côa, Portugal. He graduated in Computer Science and Systems Engineering at the University of Minho, Portugal. He holds an MSc degree in Informatics Engineering and a PhD degree in Production and Systems Engineering from the University of Minho, Portugal. His main research interests are Simulation, Operational Research and Systems Visual Modeling.
ECLIPSE-DDML:
A GRAPHICAL MODELING TOOL FOR DYNAMIC SYSTEM SIMULATION

Ufuoma Bright Ighoroje
African University of Science and Technology
Abuja, Nigeria
E-mail: uighoroje@aust.edu.ng

Mamadou Kaba Traoré
LIMOS, CNRS UMR 6158
Blaise Pascal University, Clermont-Ferrand 2, France
Email: traore@isima.fr

KEYWORDS
Graphical Modeling, DDL, DEVS, XML, Eclipse.

ABSTRACT
This paper presents Eclipse-DDML, an Eclipse plugin for graphical modeling for simulation of dynamic systems. The graphical modeling notation adopted is the DEVES-Driven Modeling Language (DDML). DDML supports the specification, analysis, design, verification, and validation of a broad range of complex systems. It provides simple but expressive constructs based on very powerful principles and formalisms like DEVS (Discrete Event System Specification), UML, and State Charts to address the requirements of dynamic system modeling and simulation. Eclipse-DDML provides enhanced graphical editing of DDL models; further simplifying model construction and promoting good modeling practices. Models developed with this tool are stored in XML-based format making them platform-independent. Integration with Eclipse simplifies software development, installation and updates, while allowing extensibility.

INTRODUCTION
Building simulation models of complex systems have become a daunting task especially because of the difficulty in collaboration between simulation and domain experts. This is as a result of the variability in interests and expertise. Domain experts are concerned with system characteristics, problems and behavior. Simulation experts use mathematical formalisms, algorithms, and/or computer programs to develop abstractions of systems. These abstractions must be translated into a simulation semantic to investigate system properties. To bridge this gap, an intermediate framework should be adopted. The framework should be simple enough and generic to be used by both worlds, at the same time the relevant aspects of the system (namely behavioral and structural) should be correctly captured and specified. This is the case for the use of UML (OMG 2009) in developing models in software engineering. Employing similar constructs for simulation modeling would definitely facilitate the system development process. Graphical modeling, as promoted by the OMG, promises to be very effective and efficient, regarding editing, human reader perception of models, and maintaining models compared to traditional textual and mathematical representation. A unified and standardized graphical notation for systems modeling and simulation will facilitate the common understanding of models for parties involved in the development of systems. DEVS (Discrete Event System Specification) (Zeigler 1976) is an established universal formalism because other formalisms has been proven to have equivalent DEVS representation (Vangheluwe 2000). The DEVS-Driven Modeling Language – DDL (Traore 2008) is a proposed graphical modeling framework to facilitate collaboration and cooperation in building simulation models of dynamic systems. DDML refines the DEVS formalism and also adopts software engineering principles. It uses formal analysis as a compliment to simulation for validation, verification, and accreditation of models against system properties.

DEVS standardization has become a subject of interest. The idea of standardizing DEVS model is to allow a platform-independent DEVS model representation to be executed by any DEVS-based simulator. (Touraille et al. 2009), (Meseth et al. 2009), (Mittal et al. 2007), and (Hosking & Sahin 2009) proposed XML-based simulator-independent DEVS modeling representations. SimStudio (Traore 2008) is a modeling and simulation environment based on the DEVS formalism. Its architecture is based on a plugin system which will allow the addition of modules and integration of existing tools, via model transformations. The communications between these heterogeneous modules rely on XML based model representations thereby promoting a standardization of the DEVS modeling language. SimStudio aims to take advantage of visual modeling by adopting DDML as its graphical modeling notation.

We introduce Eclipse-DDML, a graphical editor for DDML. We take advantage of graphical modeling to simplify construction of DEVS models and facilitate cooperation among simulation and domain engineers. Our editor translates the graphically defined models into a
standard form of representation (XML-based). It is equipped with libraries of reusable modeling components and it allows modelers to share models thereby providing a spring-board for collaboration between modelers. Simulation can be done using the SimStudio kernel or any other DEVS-based simulator due to the generality of its model representation.

THE DEVS AND DDML FORMALISMS

DEVS (Zeigler 2000) has been established as an expressive and hierarchical formalism for modeling and simulation discrete event systems. Over the years, DEVS has been used in many applications for both discrete and continuous systems of a broad range of domains. DEVS has two fundamental variants: Classic DEVS (C-DEVS) and Parallel DEVS (P-DEVS); both describe the same class of systems. DEVS distinguish between atomic models and coupled models. DEVS separates concerns – the model, simulator, and experimental environment are separated.

An atomic model is modeled as a system with several possible states specified by a set of states (S), exhibiting autonomous behavior specified by an internal transition function ($\delta_{im}$) and time advance (\(\tau\)), reacting to external stimuli specified by an external transition function ($\delta_{ex}$), and generating output events (\(\lambda\)) as its state changes through time. The atomic model interacts with its environment through input (X) and output (Y) ports. A coupled model has input and output ports (X and Y) and subcomponents that can either be atomic or coupled models themselves. These components are coupled (Z) via their ports thereby enabling interchanges between them. These components can be partitioned into EIC (External Input Coupling), EOC (External Output Coupling), and IC (Internal Coupling).

DEVS modeling could be difficult. Due to its generality, it does not propose a means to specify systems in details. The mathematical foundations are defined but the specification details are left to the modeler. Hence, DDML (Traore 2008) has been developed to fill this void. DDML refined DEVS and it provides simple graphical notations to specify system models. The functional aspects are represented as business processes (corresponding to coupled model definitions). Dynamical aspects are represented using state/activity diagrams. The static aspects are represented using an abstract structure graph, and like UML, they can be derived from the flowcharts and state charts. DDML notations provide constructs for P-DEVS and C-DEVS. Some basic notations of DDML are shown in Fig 1.

Fig. 1(a) shows the DDML Coupled Network notations. A is a coupled model while B and C are atomic models. Models have input and output ports (corresponding to X and Y in DEVS) and they are coupled via EIC, EOC, and IC. In this diagram, we are interested in the composition of the system, the sub-components, their ports, and the interaction between these components. Ports are modeled with their types indicating the domain of the values passed through them. Fig. 1(b) shows some DDML state notations. The dynamics of the system are modeled using such state charts and lines with arrows are used to indicate transitions (input, output, and confluent). State partitions are created using state variables to create equivalence classes of states. DDML introduces the concepts of activity regions in states. A system performs some activities when in a given state. The state chart expresses five functions graphically (corresponding to $\delta_{in}$, $\delta_{ex}$, $\delta_{on}$, $\lambda$, and $\tau$ in DEVS). For other notations and detailed discussion, the reader should consult (Traore 2008).

ECLIPSE-DDML: DDML GRAPHICAL EDITOR

To facilitate model construction using DDML, it is imperative to use a graphical editor. This will also promote model reuse as models can be constructed, edited, saved, and shared among modelers. In this section, we present our DDML editor based on Eclipse. The editor has separate components for creating Coupled and Atomic Models.

DDML Coupled Model Editor

The DDML coupled model editor contains tools to define DDML coupled models and sub-models (coupled models or atomic models). Figure 2 below shows a snapshot of the DDML Coupled Model Editor. The DDML coupled model editor has menu and tool bars, a project explorer, an outline view, a properties view, a rich palette of tools, and a diagram workspace.
Using the editor to define simulation models is very intuitive. The Menu Bar has: File, Edit, Diagram, Navigate, Search, Project, Run, Window, and Help menus. The toolbar contains common tools for formatting the model diagram. The project explorer view provides a hierarchical view of the project and resources in the Workbench. The outline view shows a graphical outline of the workspace. The palette contains the tools for defining a model. Models are created in the drawing workspace in a drag and drop fashion. Tools (for coupled and atomic models, input and output ports, EIC, IC, and EOC couplings, and select flags) are picked from the palette to create the model or property desired. The Eclipse-DDML workbench provides a properties view that displays the detailed properties for the selected element. Some details about a model element which can be not shown in the drawing workspace are shown in the properties view.

**DDML Atomic Model Editor**

The DDML Atomic Model Editor (shown in Fig. 3) contains tools that can be used to define states and state transitions within a process model. Just like the Coupled Model Editor, the Atomic Model Editor has a menu bar and tool bar; a project explorer, a properties view, an outline view, a diagram workspace, and a palette. Apart from the palette, the other sections are very much similar to the Coupled Model Editor. This editor can be launched by simply double clicking on an atomic model within the DDML Coupled Model editor. The Passive, Transient, and Finite States contain compartments for defining State Variables (which can be picked from the palette), and Activities. The State Activity is defined within the body of the do {} in the properties view and this must be done in the predefined language. The Time Advance for the Passive State and Transient State is set to infinity and zero respectively. External and Internal Transitions can be modeled by using the Transition tools. This can be done by simply picking the tool and connecting two states. The output function (lambda) and computation must be defined for the internal transition while the input function (trigger) and the Computation must be defined for the external transition (this can be done either graphically or in the properties view).

**DDML Domain Property File**

One of the main advantages of using this editor is that the model properties are captured and stored in an XML-based file. The graphical properties (shapes, position, color, …) of the model are separated from the model-specific properties. The markup of this domain file is the DEVS Markup Language (DML) as proposed by (Hill 2000) and (Touraille et al. 2009). This file can be queried and Since XML is a standard platform-independent file format, the model developed can be simulated by transformation into any DEVS compliant simulator defined in any language like SimStudio, DEVSIJAVA (Sarjoughian and Zeigler 1998), DEVSC++ (Zeigler et al. 1996), JDEVS (Filippi and Bisgambiglia 2004) … The domain property file takes into account the model structure, i.e. state variables, input and output ports, and couplings; the atomic model dynamics described by the transition functions; model parameterization and initialization; and model behavior corresponding to simulation results represented by trajectories.
DESIGN OVERVIEW

The Eclipse-DDML is implemented as an Eclipse plug-in. Eclipse is an open source software development platform, widely available, simple and robust to support the integration of powerful tools. The Eclipse platform can be extended by creating plugins with its Plugin Development Environment (PDE), making it very extensible. We therefore identified Eclipse as a suitable platform for the development of our graphical editor. In order to implement the graphical editor, several graphics utilities were investigated. Some of the libraries we considered include the native Java Abstract Window Toolkit (AWT) and Swing Libraries. Eclipse also provides the Standard Widget Toolkit (SWT) and JFace libraries. These libraries are very useful for defining form windows but it is difficult to build graphical editors with them. We also considered the Eclipse Draw2D library, based on SWT and created particularly to handle 2D drawings like figures and shapes. Furthermore, the Graphical Editing Framework (GEF) based on Draw2D is specifically designed to building graphical editors. On the other hand, Eclipse provides the Eclipse Modeling Framework (EMF) (Budinsky et al. 2008), a Java framework and code generation facility for building tools and other applications based on a structured model. EMF uses Ecore meta-modeling language which uses XMI (XML Metadata Interchange) as its canonical form of a model definition. What we need to do was to glue GEF and EMF in order to build a model-based graphical editor. The Graphical Modeling Framework (GMF) (Gronback 2009) provides this facility.

GMF has a Model-Driven Architecture (MDA) that makes it easier to build graphical editors with GEF based on an underlying model defined in EMF. It provides a generative component and runtime infrastructure for developing graphical editors. GMF effectively implements the Model-View-Controller (MVC) design pattern, making it possible to define graphical components and model components separately. Hence, every GEF-based application uses a model to represent the state of the diagrams being created and edited. Using EMF models provides some advantages over using arbitrary objects. The MVC architecture used by GEF relies on controllers that listen for model changes and update the view in response. If you use an EMF model, notification of model change is already in place, as all EMF model objects notify change via EMF’s notification framework.

Fig. 4 illustrates the development cycle and methodology for the Eclipse-DDML editor based on GMF, EMF, and GEF frameworks.
The development process involves creating:
- the Ecore Meta-model for DDML (ddml.ecore) which describes the abstract syntax of the modeling language;
- the graphical definition model (ddml.gmfgraph) which describes the figures and connectors. GEF provides Java classes for the basic shapes, drawing canvas, and connectors which are extended to create the DDML notations;
- the tooling definition (ddml.gmftool) which describes the palette, tool bars, menu bars, the icons/buttons, and other periphery to create and work with diagram content.

These model files are fed into GMF to define the mapping model (ddml.gmfmap), which is perhaps the most important of all models in GMF. Here, the elements from the diagram definition model (nodes and connectors) are mapped to the domain model and assigned to tooling elements from the tooling definition. The mapping model represents the actual editor definition and is used to create the generator model (ddml.gmfgen). Typically, a one-to-one mapping exists among a mapping model, its generator model, and a particular diagram. The mapping model uses Object Constraint Language (OCL) in many ways, including adding initialization features for created elements, defining link and node constraints, and defining model audits and metrics. Audits identify problems in the structure and its underlying domain model instance, and metrics provide measures of diagram and domain model elements.

The generator model adds information used to generate code from the mapping model. The GMF generator model is more of a many-to-one model transformation than a decorator model. As a mapping model is transformed into a generator model, it loses knowledge of the graphical definition and gains knowledge of the runtime notation model. This minimizes the number of dependencies linked from the generator model and separates concern among the models. The code generated is a skeleton of the DDML editor which is tweaked to get the final editor.

**RELATED WORKS ON GRAPHICAL MODELING TOOLS FOR DEVS SIMULATION**

There have been several propositions on graphical editors for creating models for discrete event simulation. We cite some of them here.

CD++ Builder (Chidisic and Wainer 2007) uses a similar approach like Eclipse-DDML. It is an IDE based on Eclipse that integrates a modeler that provides a graphical editor for DEVS and Cell-DEVS. Models can be visualized and C++ codes can be generated for simulation. However, what is missing is a general format of model representation that can be transformed onto other DEVS compliant simulators.

DEVSimPy (Capocchi et al. 2011) is an advanced wxPython GUI for the M&S of systems based on the DEVS formalism. Features include a built-in editor for modeling and a DEVS simulator. The main goal is to facilitate the modeling of DEVS systems using the GUI library and the drag and drop approach and automated simulation. Moreover, it requires the user to be familiar with the Python language to define the dynamics of the system under
consideration and it does not map the model definition to a platform-independent format.

The Virtual Laboratory Environment (VLE) (Quesnel et al. 2009) provides a complete C++ API for DEVS based simulator and a GUI for the graphical specification of the structure of the model, definitions of the experimental frame, and visualization of results. It uses the Java Swing technology to enable construction of models in several languages (Java, Python...). However, the GMF framework offers more advantage over Swing because of its easy MDA approach and generative framework.

Eclipse-DDML differs from these tools in that it is platform-independent. Models are created with less emphasis on the language of the simulation platform. The models are stored in XML format. This allows it to be glued to any DEVS-based simulation environment via automated code synthesis. The Eclipse platform is beneficial in this aspect as it provides tools to automate code generation. At the moment, we are working on this for the SimStudio Platform.

CONCLUSION

In this paper, we presented Eclipse-DDML, a graphical editor for the DEVS-Driven Modeling Language (DDML). Our aim is to reduce the efforts in system development by:
- utilizing an intuitive and expressive graphical syntax promoting collaboration between domain engineer and simulation experts
- automating the generation of platform-independent models stored in a format that is generic enough to allow for automatic code synthesis for simulation in different environments

We also presented the architecture and implementation technologies based on the Graphical Editing Framework (GMF), an MDA framework for building graphical editors. It is our aim to integrate this editor into the modelables the addition of new featuring axis of SimStudio. SimStudio’s plug-in architecture enables the addition of new features and the integration of existing tools.

REFERENCES


DEVS Standardization Group, 2008. URL: http://csci-devs.sce.carleton.ca/devsgroup/

GEF: Graphical Editing Framework. URL: http://www.eclipse.org/gef/


DISTRIBUTED PARALLEL SIMULATION
HLA HIGH-PERFORMANCE AND REAL-TIME SIMULATION STUDIES WITH CERTI

Jean-Baptiste Chaudron, Martin Adelantado and Eric Noulard
Office National d'Etudes et de Recherches Aérospatiales
2 Avenue Edouard Belin
FR-31055 Toulouse Cedex 4, France
E-mail: name.lastname@onera.fr

Pierre Siron
Institut Supérieur de l'Aéronautique et de l'Espace
10 Avenue Edouard Belin
FR-31055 Toulouse Cedex 4, France
E-mail: name.lastname@issae.fr

KEYWORDS
Real-time simulation, HLA, RTI, embedded systems.

ABSTRACT

Our work takes place in the context of the HLA (High Level Architecture) standard and its application in real-time systems context. Indeed, current HLA standard is inadequate for taking into consideration the different constraints involved in real-time computer systems. Many works have been invested in order to provide real-time capabilities to RTIs (Run Time Infrastructures). This paper describes our approach focusing on achieving hard real-time properties for HLA federations through a complete state of the art on the related domain. Our paper also proposes a global bottom up approach from basic hardware and software requirements to experimental tests for validation of distributed real-time simulation with our own RTI called CERTI.

INTRODUCTION

Modern systems become more and more complex with an increasing number of both components and interactions between them. These different applications often require their services to be delivered within a given amount of time (deadline). This focus is the problematic of real-time system which are defined as those systems in which the correctness of the system not only depends on the logical results of computation, but also on the time at which these results are produced (Stankovic 1988). Real-time systems are broadly classified into two categories based on the nature of the deadline, namely, hard real-time systems, in which the consequences of not executing a task before its deadline may be catastrophic and soft real-time systems, in which the utility of results produced by a task with a soft deadline decreases over time after the deadline expires. Examples of typical hard real-time systems are flight control and nuclear power-plant control. Telephone switching system and image processing applications are examples of soft real-time systems.

Distributed computing paradigm proposes a high performance solution thanks to advances in network technologies. Different programs located on several computers interact all together in order to achieve a global common goal. However, designers and developers of distributed software applications have to face several problems such as heterogeneity of the various hardware components as well as both operating systems and communication protocols. Development of middleware standards like CORBA (Common Object Request Architecture) (OMG 2002) allows to consistently face these problems. The term middleware describes a software agent operating as an intermediary between distributed processes (Cf. Figure 1). This software must be considered in the domain of interoperability; it is a connectivity software which enables the execution of several interacting applications on one or more linked computers.

![Figure 1: Illustration of Middleware](image)

Indeed, real-time experts investigate distributed computing solutions to ensure real-time behavior for complex systems (Stankovic 1992). However, traditional distributed standards and middleware architectures could not yet take into account real-time constraints. Real-time aircraft software and hardware embedded components interconnected with middleware have led to some particular research projects like ARMADA (Abdelzaher et al. 1997) and MIDART (MIDdleware Architecture for distributed Real-Time systems) (Gonzalez et al. 1997) and also some advances in current standards to include real-times properties, like RT CORBA (Real-Time CORBA) (OMG 2005) or more recently DDS (Data Distribution Service) (OMG 2007). The main objective of our work is to use an HLA middleware, compliant with current HLA IEEE 1516-2010 standard (IEEE 2010a) (IEEE 2010b) (IEEE 2010c) to develop, interconnect and maintain real-time simulations of embedded system (hardware-in-the-loop system or fully simulated system). This article explains how we proceed to ensure real time behavior for our simulations. The use of a distributed simulation architecture to study distributed
embedded systems should provide a more natural and flexible framework for new researches in the domain.

The paper is structured as follows: Section 2 describes the problem statement. We present the targeted applications, a background on HLA use for real-time and we describe in detail the CERTI architecture. Section 3 outlines our global approach for real-time simulation purpose. We describe all the techniques and methods used to ensure the correct temporal behavior of the simulator. Different experimental results obtained on our specific platform are illustrated in Section 4. Finally, a discussion of results, as well as currently planned extensions of the infrastructure, is proposed in conclusion.

PROBLEM STATEMENT

Targeted applications

Our work takes place in a global project named PRISME (Plate-forme de Recherche et d'Ingénierie des Systèmes Embarqués). The main focus of this project is to study new embedded system concepts and techniques through a special hardware and software environment. All these simulations could also be Hardware-in-the-loop simulations by connecting real actors: actuators, sensors or real embedded computers in the simulation loop. Obviously, these simulations could also be Human-in-the-loop simulations but we are focusing here on real-time aspects.

A collaborative study between ONERA (Office National d’Études et de Recherches Aérospatiales) and CNES (Centre National d’Etudes Spatiales) laboratories gave first elements to understand the use of the HLA standard and CERTI runtime infrastructure for real-time simulations (Noulard et al. 2008). The case study, is a satellite formation flying simulation that is made up by four components that are embedded systems simulators for two satellites as depicted by figure 2: **Federate 1** is a simulator of the board computer on satellite 1; **Federate 2** is a simulator of the dynamics of the satellite 1; **Federate 3** is a simulator of the dynamics of the satellite 2 and finally **Federate 4** is a simulator of the board computer on satellite 2.

![Figure 2: CNES Satellites formation flying simulation](image)

**HLA real time background**

Simulation is a well established technique used in the man-machine system area for training, evaluation of performance and research. However, works to include real-time specifications and properties to HLA standard are less advanced than others ones (Zhao 2001). We claim that the choice of a distributed computing standard and its underlying middleware is an important starting point to obtain high fidelity, valid and scalable real-time simulations. This choice implies which operating system, which programming language and which hardware could be used for compliance with the middleware. The RTI is the distributed software for interconnecting various federates to a global federation execution. The RTI-NG (RTI Next Generation) (Bachinsky et al. 1999) was the first run-time infrastructure developed and used by the US Department of Defense; this RTI is no longer maintained. Since then, several approaches have been investigated to add real-time properties to HLA standard and underlying software RTI:

1. Multi-threaded synchronous process for RTI (Zhao and Georgeanas 2001) (McLean et al. 2004) (Boukerche and Kaiyuan 2005);
2. Global scheduling services in RTI (Zhao and Georgeanas 2001) (Boukerche and Kaiyuan 2005);
3. Real-time Optimized RTI services like time Management from Fujimoto and McLean (McLean et al. 2004) or Data Distribution Management for Boukerche works (Boukerche and Kaiyuan 2005);
4. Quality of service communication with, for example, RSVP (Ressource ReSerVation Protocol) (Zhao 2001) or specific protocols like VRTP (Virtual Reality Transfer Protocol) (Brutzman et al. 1997);
5. Use a real-time operating system to allow preemptive priority scheduling (Jansen et al. 2004).

These different techniques allow an improved use of system resources, better scalability and also a higher reactivity of services provided by the RTI. However, no work proposes a complete analysis from simulation requirements to implementation. Most of all, the run-time infrastructure is never clearly presented (except for (Zhao 2001) (Zhao 2001) which used RTI-NG).

**Bottom-Up approach (Actions Levels)**

The temporal properties of distributed real-time simulation are obtained from a complex combination of the application structure, the used HLA middleware and specific distributed algorithms, the software infrastructure (operating systems and communication protocols) and finally the physical infrastructure (type of computers, type of networks and distribution topology). The specific PRISME platform architecture is composed of:

**Hardware:** 4 real-time nodes with Opteron 6 core processors, 2 Graphical HP station computer with Intel Xeon processors and high performance GPUs (Graphics Processing Units), an ethernet Gigabit switch on a dedicated network and also two input organs (Yoke/Throttle/Pedal systems). This global system also proposes a particular advantage, a distributed clock technology allowing same clock reference to each node (Concurrent Computer Corporation 2001).

**Software:** Linux Red Hawk (Baietto et al. 2008) Operating system compliant with POSIX real-time standard (Gallmeister 1995). This RTOS (Real Time Operating System) has been already used in the simulation domain by
TNO laboratory which uses this OS to run their own RTI implemented in C++. Their experiments concluded that this operating system is suitable for real-time computing (Jansen et al. 2004).

**Middleware:** In our approach, we will rely on our Open Source RTI called CERTI because we have a complete knowledge of its implementation.

**CERTI Middleware**

For years, the French Aerospace Laboratory (ONERA) has been developing his own Open-Source middleware RTI compliant with HLA standard called CERTI (Siron et al. 2009). This RTI runs on several operating systems including Linux and Windows. It is recognizable through its original architecture of communicating processes (Cf. Figure 3). Each federate process interacts locally with an RTIA (RTI Ambassador) process through a Unix-domain socket (equivalent to Local Run-time Component or LRC). The RTIA processes exchange messages over the network, in particular with the RTIG (RTI Gateway) process (equivalent to Central Run-time Component or CRC), via TCP (and also UDP) sockets, in order to run the distributed algorithms associated with the RTI services.

![Figure 3: CERTI architecture](image)

The CERTI has, originally, no mechanism for taking into account quality of service and no tools to provide an end to end predictability. In this sense, it does not handle events differently according to a priority and it uses no predictability mechanism whatsoever at the network or the operating system level. In our case, a key benefit is to master the implementation of RTI used and thus be able to incorporate changes in the source code to ensure temporal predictability of CERTI.

**OUR APPROACH**

**Towards periodic federates**

The concept of periodic federates, named "repeatability within simulations" has been introduced by Fujimoto and McLean (Fujimoto 1997) (Fujimoto and McLean 2000) with their works on real-time and distributed simulations. Federates, involved in this kind of simulation, repeat the same pattern of execution periodically with a time step noted \( \Delta t \). During each step, federates carry four phases: a reception phase, a computation phase, a transmission phase and a slack time phase. ONERA and CNES studies (Noulard et al. 2008) show the necessity of explicitly adding a synchronization phase to ensure the global coherent run time of the whole simulation (Cf. Figure 4).

![Figure 4: Periodic federate scheme](image)

Historically, in DIS simulation standard (Cheung and Loper 1994), this synchronization phase is made (for each federate) by consulting global WCT (Wall Clock Time) available for each simulator. ONERA and CNES works present a new original synchronization mechanism by sending an interaction from the fastest federate, called pulse, which rhythms the whole simulation run-time. In Fujimoto and McLean works, synchronization and reception phases are made in the same time by time management mechanisms. To summarize, the synchronization phase can be done either by three different methods:

1) Consult the hardware clock on a mono-processor system or use a distributed hardware clock like RCIM (Real-Time Clock and Interrupt Module) system for distributed applications available on our Linux Red Hawk platforms;
2) The federate which has the highest speed cycle sends an interaction to all others in order to rhythm the execution of all others federates involved in the federation;
3) Use of Time Management HLA mechanisms to ensure messages delivery in all federation and synchronize every federates steps. Note that, these time steps could be different according to application requirements.

**Execution modes**

We distinguish two different run-time modes based on periodic federates. The first one is the Data Flow model. This kind of execution mode is only scheduled by the communication flow between each federate. Each federate waits for a data to run its local algorithm and computes its own new data for the rest of the federation. This approach could only be used on synchronous distributed systems like PRISE Red Hawk RCIM synchronized nodes. Federates communicate using HLA basic publish and subscribe principles through RTI services calls like \texttt{updateAttributeValue()} (Cf. Figure 5). We assume that the receiver federate is waiting for a \texttt{reflectAttributeValue()} callback in reception phase. Each federate then runs its own algorithm when it receives an available input data. The main interest of this run-time execution mode is the simplicity of modeling its behavior with a formal model compliant with real-time scheduling policies and techniques. However, the developers have to ensure by programming which cycle receives which data. This approach is not very suitable for adding new federate or to plug existing federates to another federation execution. Most of all, there is no safety guarantee during the run time. If the application was not
well scheduled, a federate could always be blocked (waiting for an expected data). So one needs to be accurate with the formal model and its implementation to ensure good execution of the whole federation.

![Figure 5: Data Flow execution mode](image)

Other execution modes use time-management mechanisms provided by HLA standard (Fujimoto 1998). During the runtime, each federate computations and communications are scheduled by time management principles and algorithms. A suitable deployment of these techniques ensures a consistent temporal behavior on a common time reference: the simulated time axis. This approach is the best way to maintain consistency between federates located on asynchronous computers (no common Wall Clock Time).

The main advantage of time management is the possibility to easily add some new federates. The temporal behavior and consistency of the whole simulation is based on simulated time coherence. The time advance could also be correlated to an hardware clock to ensure the respect of real time constraints. Accordingly with the HLA standard, all federates are both regulators and constrained. Two kinds of services allow the federate to express its requests for advancing its local logical time: nextEventRequest(t) and timeAdvanceRequest(t).

The nextEventRequest(t) service (noted NER(t)) allows to receive the next event available for asked simulated time NER(t) and then a timeAdvanceGrant(t)’ callback (noted TAG(t’)) given by the RTI with a time stamp equal to the time stamp t’ of the simulation message (t’ could be less than t). This kind of federate is called Event Driven federates (Cf. Figure 6).

![Figure 6: Event Driven execution mode](image)

The timeAdvanceRequest(t) service (noted TAR(t)) ensures the delivery of all available messages. The RTI grants this logical time advance (guaranteeing causality constraints) by invoking all the available reflectAttributeValue callbacks (noted RAV(t)) and finally by accepting the time advance through the invocation of the timeAdvanceGrant(t) callback. This kind of federate is called Time Stepped federates (Cf. Figure 7).

![Figure 7: Time Stepped execution mode](image)

Necessity of formal proof for real-time

To our knowledge, no related work from simulation community has linked any formal model from scheduling theory with concepts of distributed simulations (especially with HLA standard). Thus real-time simulations are usually validated by experiments rather than formal models and schedulability analysis. But, we claim that some formal models compliant with schedulability techniques are essential to validate real-time behavior of our simulations.

For example, researches on RT CORBA standard have investigated the validation of the global end to end behavior by combining scheduling techniques Deadline Monotonic algorithm, DPCP (Distributed Priority-Ceiling Protocol) (Dipippo et al. 2001) and an algorithm to map priorities founded by formal results to local priorities provided by local operating systems on each node.

Figure 2 shows that each federate (each computation made by a federate) is illustrated by a box. Each CERTI communication between federate is represented by an arrow. These data dependencies could be modeled by using different techniques. In previous paper (Chaudron et al. 2010), we showed the feasibility to formally validate basic Data Flow simulations on mono-processor system by combining Deadline monotonic techniques and simple precedence constraints. We extend this formalism to describe our distributed Data Flow applications by using and adapting Tindell and Clark holistic method (Tindell and Clark 1994). These techniques allows to take into account the dependency between the scheduling of tasks and messages in distributed real-time systems.

In order to add determinism to first generation time management mechanisms involved in CERTI software (based on Chandy-Misra-Bryant algorithm (Chandy and Misra 1979 ) ), we recently propose an analytical methodology to formally quantify the number of null messages exchanged between each time-driven real-time periodic federates (federates which use timeAdvanceRequest(t) service) involved in a real time simulation (Chaudron et al. 2011). We also add a new algorithm called NULL MESSAGE PRIME adapted to event-driven real-time periodic federates (federates which use nextEventRequest(t) service) which exhibits very
interesting properties, including a solution to the time creep problem. We currently investigate some model checking strategies using UPPAAL tool (Behmann et al. 2004) in order to exhibit formal proofs and better evaluate of time management services and their implementation. The formal validation part of our work is not described in present paper, we are here focusing on experimental aspects.

EXPERIMENTALS RESULTS

WCET and WCTT measurements

The execution time of a program usually depends on the input data. In the context of real-time systems, it is necessary to be able to estimate the WCET (Worst-Case Execution Time). For hard real-time systems, it is necessary to assess the execution time in the worst case to properly size the system and find the best allocation of tasks among the processors. In our case, we have made some measurements of execution time for a given temporal complexity of algorithm (Cf. Table 1). We assume that spatial complexity (memory) is properly dimensioned according to embedded systems requirements.

<table>
<thead>
<tr>
<th>O(n^m)</th>
<th>m=1</th>
<th>m=2</th>
<th>m=3</th>
<th>m=4</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=10</td>
<td>0.001</td>
<td>0.003</td>
<td>0.007</td>
<td>0.065</td>
</tr>
<tr>
<td>n=20</td>
<td>0.001</td>
<td>0.005</td>
<td>0.052</td>
<td>1.182</td>
</tr>
<tr>
<td>n=30</td>
<td>0.001</td>
<td>0.008</td>
<td>0.181</td>
<td>5.838</td>
</tr>
<tr>
<td>n=40</td>
<td>0.001</td>
<td>0.010</td>
<td>0.386</td>
<td>18.240</td>
</tr>
<tr>
<td>n=50</td>
<td>0.001</td>
<td>0.015</td>
<td>0.803</td>
<td>44.077</td>
</tr>
</tbody>
</table>

Table 1: Execution time of an algorithm with O(n^m) complexity (in milliseconds)

Calculation of WCTT (Worst Case Transit Time) values for any CERTI message must take into account three phases (Cf. Figure 8). Phase 1 is the copy on local host Unix domain socket and the local computation of the Sender Federation associated RTIA process. Phase 2 describes the time to read and write on different communication TCP (or UDP) sockets over the network or on the local host, and the time needed for RTIG local computation. Phase 3 is the copy on local host Unix domain socket and the local compute of Receiver Federation associated RTIA process.

![Figure 8: CERTI communication steps](image)

We have developed a benchmark called PING-PONG used to measure CERTI communication latency. Two federates PING and PONG exchange messages (with a given size specified by user) through CERTI. Table 2 gathers experimental measurements (given in milliseconds) of CERTI transit time with respect to three configurations:

- **Configuration 1**: Federate PING, federate PONG, both RTIAs and RTIG run on one single PRISE Red Hawk node;
- **Configuration 2**: Federate PING, its RTIA and RTIG run on a single PRISE Red Hawk node and Federate PONG and its RTIA run on another node;
- **Configuration 3**: Federate PING, its RTIA run on a single PRISE Red Hawk node, Federate PONG and its RTIA run on another node and finally RTIG run also on its own PRISE node;

<table>
<thead>
<tr>
<th>Message size</th>
<th>Config. 1</th>
<th>Config. 2</th>
<th>Config. 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 bits</td>
<td>0.293</td>
<td>0.252</td>
<td>0.236</td>
</tr>
<tr>
<td>500 bits</td>
<td>0.315</td>
<td>0.263</td>
<td>0.256</td>
</tr>
<tr>
<td>1000 bits</td>
<td>0.353</td>
<td>0.281</td>
<td>0.286</td>
</tr>
<tr>
<td>5000 bits</td>
<td>0.406</td>
<td>0.411</td>
<td>0.422</td>
</tr>
<tr>
<td>10000 bits</td>
<td>0.422</td>
<td>0.478</td>
<td>0.522</td>
</tr>
<tr>
<td>50000 bits</td>
<td>1.066</td>
<td>1.372</td>
<td>1.607</td>
</tr>
</tbody>
</table>

Table 2: CERTI WCTT measurements (in milliseconds)

Data Flow execution mode

For CNES federation, as illustrated in Figure 2, federate 1 and federate 4 run a loop of 50 ms (20 Hz) and federate 2 and federate 3 run a loop of 10 ms (100 Hz). The data flow execution model has a good behavior for real-time purpose on our specific platform. Federate 1 and 4 compute an algorithm in O(30^2) and Federate 2 and 3 compute an algorithm with complexity equal to O(10^2). Table 3 shows that all cycles respect corresponding periods (10 ms and 50 ms) and the global behavior is stable for all cycles.

<table>
<thead>
<tr>
<th>Fed. 1</th>
<th>Fed. 2</th>
<th>Fed. 3</th>
<th>Fed. 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>49,394</td>
<td>49,449</td>
<td>49,585</td>
</tr>
<tr>
<td>Mean</td>
<td>49,449</td>
<td>9,119</td>
<td>9,458</td>
</tr>
<tr>
<td>Max</td>
<td>49,585</td>
<td>9,131</td>
<td>9,501</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>2 ms</td>
<td>1,005</td>
<td>4,146</td>
</tr>
</tbody>
</table>

Table 3: Federate cycle duration in milliseconds (Data Flow periodic)

We also focus on the acceleration of the application rhythm to allow the federation to run as fast as possible. For these experiments, we keep the speed ratios between the different federates cycles. Thus federates 1 and 4 are five times slower than federates 2 and 3 (and also corresponding communications). We retain the complexity of the algorithms computed by each federate. Table 4 show that, with corresponding algorithms complexities, faster federates (2 and 3) could respect a computational period equal to 2 ms and slower federates could ensure the respect of a period less than 10 ms. These results show that CERTI could ensure high frequency communicating processes with Data Flow execution mode.
<table>
<thead>
<tr>
<th>Federate</th>
<th>Min</th>
<th>Mean</th>
<th>Max</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fed. 1</td>
<td>6,108</td>
<td>6,136</td>
<td>6,292</td>
<td>0,056</td>
</tr>
<tr>
<td>Fed. 2</td>
<td>1,041</td>
<td>1,176</td>
<td>2,119</td>
<td>0,267</td>
</tr>
<tr>
<td>Fed. 3</td>
<td>1,045</td>
<td>1,213</td>
<td>2,082</td>
<td>0,354</td>
</tr>
<tr>
<td>Fed. 4</td>
<td>6,048</td>
<td>6,158</td>
<td>6,355</td>
<td>0,092</td>
</tr>
</tbody>
</table>

Table 4: Federate cycle duration in milliseconds (Data Flow as fast as possible)

**Time Management execution mode**

For time management model, we choose the Time Stepped execution mode to ensure consistency between the real time and the simulated time. In this case, classical null message algorithm implemented in CERTI seems to have a good behavior to ensure real-time properties to our simulator (Cf. table 5). Indeed, all computed cycles are respected (10 ms and 50ms); the global behavior is also very regular even if some irregularities appears compared to Data Flow execution mode.

<table>
<thead>
<tr>
<th>Federate</th>
<th>Min</th>
<th>Mean</th>
<th>Max</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fed. 1</td>
<td>48,640</td>
<td>49,765</td>
<td>50,807</td>
<td>0,532</td>
</tr>
<tr>
<td>Fed. 2</td>
<td>9,514</td>
<td>9,592</td>
<td>10,618</td>
<td>0,172</td>
</tr>
<tr>
<td>Fed. 3</td>
<td>9,372</td>
<td>9,624</td>
<td>10,959</td>
<td>0,248</td>
</tr>
<tr>
<td>Fed. 4</td>
<td>48,029</td>
<td>49,474</td>
<td>50,787</td>
<td>0,841</td>
</tr>
</tbody>
</table>

Table 5: Federate cycle duration in milliseconds (Time Management periodic)

One more time, we accelerate the application rhythm to allow the federation to run as fast as possible by using classical CERTI time management implementation. The use of TAU (HLA services calls) for each federate steps seems to generate some overhead (compared with Data flow model). In this case, the number of NULL messages generated by original algorithm is acceptable for real-time specification (hard real time deadline). Table 6 shows that, with corresponding algorithms complexities, faster federates (2 and 3) could respect a computational period equal 7 ms and slower federates could ensure the respect of 15 ms period (for the worst case). These results show that CERTI could ensure high frequency communicating processes as well with Time management execution. As a conclusion, time management mechanisms provided by CERTI middleware enforce a good synchronization for our kind of real-time federates.

<table>
<thead>
<tr>
<th>Federate</th>
<th>Min</th>
<th>Mean</th>
<th>Max</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fed. 1</td>
<td>13,266</td>
<td>13,376</td>
<td>13,607</td>
<td>0,100</td>
</tr>
<tr>
<td>Fed. 2</td>
<td>1,582</td>
<td>2,676</td>
<td>6,487</td>
<td>1,883</td>
</tr>
<tr>
<td>Fed. 3</td>
<td>1,544</td>
<td>2,678</td>
<td>6,587</td>
<td>1,875</td>
</tr>
<tr>
<td>Fed. 4</td>
<td>13,293</td>
<td>13,427</td>
<td>13,766</td>
<td>0,139</td>
</tr>
</tbody>
</table>

Table 6: Federate cycle duration in milliseconds (Time Management as fast as possible)

**PERPECTIVES AND CONCLUSION**

We propose, in this paper, experimental results from our work on real-time simulations with our CERTI middleware. However, real-time analysis requires the modeling of several aspects of a distributed simulation. Different static scheduling and run time analysis have been studied under different hypothesis (single processor, distributed synchronous processors, distributed asynchronous processors, ...). Interested reader could refer to previous papers (Chaudron et al. 2010) (Chaudron et al. 2011) to get a more complete description of formal part of our work.

This paper shows that current CERTI performances are very good for real-time and/or high performance simulations. We have also developed and updated a lot of tools to manage the allocation of both federate and CERTI processes over PRISE processors and modify the priority of each one for compliance with scheduling technique used. These new implementations, that are not described in present paper, help to ensure better responsiveness of HLA services. Indeed, we pursue our efforts and we currently work on HP-CERTI (High-Performance CERTI) approach (Adelantado et al. 2004) to replace Unix and TCP communication sockets through shared memories (for exchange on the same node). In addition, we will evaluate the use of multi-threading for process RTIG and ensure real-time properties for all messages passing through it. As well, we plan to use real-time dynamic memory allocators from TLSF (Two-Level Segregate Fit) library (Masmano et al. 2004) and first experiments show promising results.

We have recently implemented and tested an HLA aircraft component-based federation composed by nine federates, each representing a specific part of the aircraft or environment (article under submission process). This simulation is human-in-the-loop and the operator could interact with the simulation by a federate which acquires the user orders transmitted by a real yoke/throttle/pedals system. Now, we think that our work on real-time simulations is mature (as well as our middleware CERTI). Indeed, hard real-time properties of our architecture (and both techniques to manage it) could allow the connection of simulators with real physical actuator and sensors and/or real embedded systems to run hardware-in-the-loop simulations with high-frequency requirements.

**REFERENCES**


Warp-Level Parallelism: Enabling Multiple Replications In Parallel on GPU

Jonathan PASSERAT-PALMBACH\textsuperscript{1,2,3} Jonathan CAUX\textsuperscript{1,2,3,4} Pridi SIREGAR\textsuperscript{4}
Claude MAZEL\textsuperscript{1,2,3} D.R.C. HILL\textsuperscript{1,2,3}
\{passerat, caux, mazel\}@isima.fr pridi.siregar@biocomputing.com david.hill@univ-bpclermont.fr

\textsuperscript{1}ISIMA - Institut Supérieur d’Informatique, de Modélisation et de leurs Applications, F-63173 AUBIÈRE
\textsuperscript{2}Clermont Université, Université Blaise Pascal, LIMOS, BP 10448, F-63000 CLERMONT-FERRAND
\textsuperscript{3}CNRS, UMR 6158, LIMOS, F-63173 AUBIÈRE
\textsuperscript{4}Integrative Biocomputing, Nouvelles Structures - Place du Granier, F-35135 CHANTEPIE

KEYWORDS
Stochastic Simulation, Multiple Replications In Parallel (MRIP), GP-GPU, CUDA, Warp-Level Parallelism (WLP)

ABSTRACT
Stochastic simulations need multiple replications in order to build confidence intervals for their results. Even if we do not need a large amount of replications, it is a good practice to speed-up the whole simulation time using the Multiple Replications In Parallel (MRIP) approach. This approach usually supposes to have access to a parallel computer such as a symmetric multiprocessing machine (with many cores), a computing cluster or a computing grid. In this paper, we propose Warp-Level Parallelism (WLP), a GP-GPU-enabled solution to compute MRIP on GP-GPUs (General-Purpose Graphics Processing Units). These devices display a great amount of parallel computational power at low cost, but are tuned to process efficiently the same operation on several data, through different threads. Indeed, this paradigm is called Single Instruction, Multiple Threads (SIMT). Our approach proposes to rely on small threads groups, called warps, to perform independent computations such as replications. We have benchmarked WLP with three different models: it allows MRIP to be computed up to six times faster than with the SIMT computing paradigm.

INTRODUCTION
Replications are a widespread method to obtain confidence intervals for stochastic simulation results. It consists in running the same stochastic simulation with different random sources and averaging the results. According to the Central Limit Theorem, the average result is approximated in an accurate enough way by a Gaussian Law, for a number of replications greater than 30. Thus, for a number of replications greater than 30, we can obtain a confidence interval with a satisfactory precision.

There are many cases where a single simulation can last for a while, so 30 of them run sequentially may represent a very long computation time. Because of this overhead, 30 replications are hardly run in most simulations. Instead, a good practice is often to run 3 replications when debugging, and 10 replications are commonly used to compute a confidence interval. To maintain an acceptable computation time while running 30 or more replications, many scientists proposed to run in parallel these independent simulations. This approach has been named Multiple Replication in Parallel (MRIP) in the nineties (Pawlikowski et al. 1994). As its name suggests, its main idea is to run each replication in parallel (Hill 1997; Pawlikowski 2003). In addition, when we explore an experimental plan we have to run different sets of replications, with different factor levels according to the experimental framework (Hill 1996; Amblard et al. 2003). In this paper, we will not consider any constraint that need to be satisfied when implementing MRIP. One of the main barriers that often prevents simulationists to achieve a decent amount of replications is, on the one hand, the lack of knowledge in the parallelization techniques and on the other hand the parallel computing facilities available. Our work tackles this problem by introducing a way to harness the computational power of GP-GPUs (General-Purpose Graphics Processing Units – GPUs hereafter), which are rather cheap compared to regular parallel computers, to process MRIP quicker than on a scalar CPU (Central Processing Unit).

GPUs deliver such an overwhelming power at a low cost that they now play an important role in the High Performance Computing world. However, this kind of devices display major constraints, tied to its intrinsic architecture. Basically, GPUs have been designed to deal with computation intensive applications such as image processing. One of their well-known limits is memory access. Indeed, since GPUs are designed to be efficient at computation, they badly cope with applications frequently accessing memory. Except by choosing the right applications, the only thing we can do to overcome this drawback is to wait for the hardware to evolve in such
a way. Last NVIDIA GPUs generations, codenamed Fermi, show a move in this way by improving cache memories available on the GPU. This leads to better performances for most applications at no development cost, only by replacing the old hardware by the state-of-the-art one.

Now, what we can actually think about is the way we program GPUs. Whatever the programming language or architecture one chooses to develop his application with, CUDA (Compute Unified Device Architecture) or OpenCL, the underlying paradigm is the same: SIMT (Single Instruction, Multiple Threads). Thus, applications are tuned to exploit the hardware configuration, which is a particular kind of SIMD architecture (Single Instruction, Multiple Data). To obtain speedups, we must propose parallel applications that will be SIMD compliant. This point reduces the scope of GPU-enabled applications.

In the SIMT paradigm, threads are automatically grouped into 32-wide bundles called warps. Warps are the base unit used to schedule both computation on Arithmetic and Logic Units (ALUs) and memory accesses. Threads within the same warp follow the SIMD pattern, i.e. they are supposed to execute the same operation at a given clock cycle. If they do not, a different execution branch is created and executed sequentially every time a thread needs to compute differently from its neighbours. The latter phenomenon is called branch divergence, and leads to significant performance drops. However, threads contained in different warps do not suffer the same constraint. They are executed independently, since they belong to different warps.

In this paper, we introduce Warp-Level Parallelism (WLP), a paradigm to evaluate the approach of using GPUs to compute MRIP, using an independent warp for each replication. Our study will:

- Describe a mechanism to run MRIP on GPU;
- Propose an implementation of our approach: WLP;
- Benchmark WLP with three different simulation models.

**GENERAL CONCEPTS OF GPU PROGRAMMING AND ARCHITECTURE**

This section does a brief recall of the major concepts introduced by GPU programming and especially by CUDA. It also basically describes how a GPU architecture is organized, since these aspects are directly tied to our approach.

**The Single Instruction Multiple Threads (SIMT) paradigm**

SIMT is the underlying paradigm of any CUDA application. It is based on the well-known SIMD paradigm. While using SIMD, the same instruction is executed in parallel on multiple computational units, but take different data flows in input. Instead of viewing SIMT as a simple SIMD variant, one needs to understand that it has been created to simplify applications development on GPU. The main idea is first to allow developers to deal with a unique function, named a kernel, which is going to be run in parallel on the GPU. Second, developers manipulate threads in SIMT, which are a much more common tool nowadays than traditional vectors enabling SIMD parallelization.

In order to handle SIMT more easily, CUDA introduces different bundles of threads. As a matter of fact, threads are grouped into blocks, which size and 3D-geometry are defined by the user. The whole blocks of a kernel form a 2D grid. Each thread will be uniquely identified in the kernel thanks to an identifier computed from a combination of its own coordinates and of its belonging block’s. More precisely, in addition to grid and blocks, CUDA devices automatically split threads into fixed-size bundles called warps. Currently, warps contain 32 threads. This group is extremely important in the low-level mechanisms running on a GPU.

As long as NVIDIA has defined both its GPU architecture and the SIMT paradigm, the latter is not only convenient, it also perfectly fits its host architecture. Its sole purpose is to be used on GPU architecture, which is quite different from other multi-core architectures, especially from CPU ones, as we will see in the next part.

**Basic architecture of a GPU**

While a CPU possesses few cores, each of them allowing the execution of one thread at a time, a GPU possesses a small number of Streaming Multiprocessors (SM) (for instance an NVIDIA Fermi C2050 has 14 SMs). Each SM embeds an important number of computational units (there are 32 floating point computational units - called Streaming Processor (SP) - on each SM of a Fermi C2050). In theory, the floating-point computation power of a GPU board is equal to the number of SMs multiplied by the number of SPs. Another figure that needs to be considered in the architecture is the number of warp schedulers. The latter are key elements of CUDA performance. In fact, memory accesses are done per warp. However, because of memory latency, the warps-schedulers select the warps that have their data ready to process. Consequently, the more warps can be scheduled, the more the memory latency can be hidden.
When the former generation of NVIDIA GPUs was issued with a single warp-scheduler per SM (Lindholm et al. 2008), Fermi now owns two warp schedulers per SM (Wittenbrink et al. 2011). They are first employed when threads need to be scheduled on the SM they have been assigned to. In fact, threads within a warp also achieve memory accesses in parallel, before processing the same instruction on these data. To sum up, when threads are bound to each other, and must execute the same instructions according to SIMD machinery, warps are the smallest unit that run in parallel on the different SMs of a GPU, and are the smallest GPU element that is able to process independent code sections. Indeed, given that different warps either run on different SMs, or on the same but at different clock ticks, they are fully independent to each other. Figure 1 shows a simplified representation of a SM of the Fermi architecture.

Moreover, GigaThread enables immediate replacement of blocks on an SM when one completes executing. Since context switching has been fastened with Fermi, blocks of threads can fully take advantage of the hardware device thanks to GigaThread dispatching capabilities. From an external point of view, and since we do not have the real specifications, we have noted that the dispatching of blocks does not seem to be deterministic. NVIDIA uses a specific way to place blocks on SMs: indeed, SMs will not be enabled in order. SMs bearing non-consecutive identifiers will in fact run consecutively ordered blocks.

A WARP MECHANISM TO SPEED UP REPLICATIONS

Two problems arise when trying to port replications to GPU threads, considering a replication per thread. First, we generally compute few replications, whereas we have seen that GPUs needed to achieve large amounts of computations to hide their memory latency. Second, replications of stochastic simulations are not renowned for their SIMD-friendly behaviour. Usually, replications fed with different random sources will draw different random numbers at the same point of the execution. If a condition result is based on this draw, divergent execution paths are likely to appear, forcing threads within a same warp to be executed sequentially because of the intrinsic properties of the device.

The idea that we propose in this paper is to take advantage of the previously introduced warp mechanism to enable fast replications of a simulation. Instead of having to deal with Thread-Level Parallelism (TLP) and its constraints mentioned above, we place ourselves at a slightly higher scope to manipulate warps only. Let this paradigm be called Warp-Level Parallelism (WLP), as opposed to TLP. Now running only one replication per warp, it is possible to have each replication to execute...
different instructions without being faced to the branch divergence problem.

But to successfully enable easy development of simulation replications on GPU using one thread per warp, two mechanisms are needed.

First, it is necessary to restrict each warp to use only one valid thread. By doing so, we ensure not to have divergent paths within a warp. Moreover, we artificially increase the device’s occupancy, and consequently, we take advantage of the quick context switching between warps to hide slow memory accesses. Theoretically, we should use the lowest block size maximizing occupancy. For instance, a C2050 board owns 14 SMs, and can schedule at most 8 blocks per SM. In this case, the optimal block size when running 50 replications would be 32 threads per block. This situation is represented in Figure 2, where we can see two warps running their respective first threads only. The 31 remaining threads are disabled, and will stall until the end of the kernel. Unfortunately, the GigaThread scheduler, introduced in the previous section, does not always enable a kernel to run on every available SM. In addition, SMs’ memory constraints might compromise this ideal case by reducing the number of available blocks per SM.

![Figure 2: Representation of thread disabling to place the application at a warp-level](image)

Second, there has to be an easy solution to get a unique index for each warp. TLP relies essentially on threads identifiers to retrieve or write data back. Thus, WLP needs to propose an equivalent mechanism so that warps can be distinguished to access and compute their own data.

Thanks to the two tools introduced in this section, it is possible to create a kernel where only one thread per warp will be valid, and where it will be easy to make each valid thread compute different instructions, or work on different data depending on the new index.

Although we could not figure out the real behavior of the GigaThread Engine dispatcher, the characteristics noticed in this part are sufficient to evaluate the performance of the dispatching policy. Furthermore, the new scheduling features introduced in Fermi significantly enhance the overall performance of our warp-based approach, given that it highly relies on warp scheduling and block dispatching.

**IMPLEMENTATION**

Now that we have defined our solution, we will propose an implementation in this section. To do so, we need to focus on two major constraints: first, we should keep a syntax close to C++ and CUDA, so that users are not confused when they use our approach. Second, we need to propose compile-time mechanisms as much as possible. Indeed, since WLP only exploits a restricted amount of the device’s processing units, we have tried to avoid any overhead implied by our paradigm.

This paper intends to prove that our approach is up and running. Thus, this section will only introduce a restricted number of keywords used by WLP. As we have seen previously, we first have to be able to identify the different warps, in the same way SIMT does with threads. One way to obtain the warp identifier is to compute it at runtime. Indeed, we know that warps are formed by 32 threads in current architectures [NVIDIA2011a]. Thus, knowing the running kernel configuration thanks to CUDA defined data-structures, we are able to figure out the warp identifier with simple operations only, similarly to what have done (Hong et al. 2011). The definition of a `warpIdx` variable containing the warp’s identifier can be written as in Figure 3:

```c
const unsigned int warpIdx = (threadIdx.x + blockDim.x * (threadIdx.y + blockDim.y * (threadIdx.z + blockDim.z * (blockIdx.x + gridDim.x * blockIdx.y))))) / warpSize;
```

![Figure 3: Const-definition of warpIdx](image)

Conceptually, this definition is ideal because `warpIdx` is declared as a ‘constant variable’, and the warp identifier does not change during a kernel execution. This formula fits with the CUDA way to number threads, which first considers threads’ x indices, then y and finally z, within a block. The same organization is applied to blocks numbering (Kirk and Hwu 2010). Please note that the `warpSize` variable is provided by CUDA. This makes our implementation portable since warp sizes may evolve in future CUDA architectures.

Although this method introduces superfluous computations to figure out the kernel’s configuration, we find it easier to understand for developers. Another way to compute the warp’s identifier would have been to write CUDA PTX assembly (NVIDIA 2011). The latter is the Instruction Set Architecture (ISA) currently used by CUDA-enabled GPUs. CUDA enables developers to insert inlined PTX assembly into CUDA high-level code,
as explained in [NVIDIA2011b]. However, this method is far less readable than ours, and would not be more efficient since we only compute warpIdx once: at initialization.

This warp identifier will serve as a base in WLP. When classical CUDA parallelism makes a heavy use of the runtime-computed global thread identifier, WLP proposes warpIdx as an equivalent.

Now that we are able to figure out threads’ parent warps, let us restrain the execution of the kernel to a warp scope. Given that we need to determine whether or not the current thread is the first within its belonging warp, we will be faced to problems similar to those encountered when trying to determine the warp identifier. In fact, a straightforward solution reckoning on our knowledge of the architecture quickly appears. It consists in determining the global thread identifier within the block to ensure it is a multiple of the current warp size. Once again, the kernel configuration is issued by CUDA intrinsic data structures, but we still need a reliable way to get the warp size to take into account any potential evolution. Luckily, we can figure out this size at runtime thanks to the aforementioned warpSize variable. Consequently, here is how we begin a warp-scope kernel in WLP:

```c
if ( ( threadIdx.x + blockDim.x * threadIdx.y + blockDim.y * threadIdx.z ) )
  warpSize == 0
```

Figure 4: Directive enabling warp-scope execution

We now own the bricks to perform WLP, but still lack a user-friendly API. Indeed, it would not be adapted to ask our users to directly use complex formulas without having wrapped them up before in higher-level calls. To do so, we chose to use macros, for the sake that they are compile-time mechanisms, thus not causing any runtime overhead, and that they are perfectly handled by nvcc, the CUDA compiler. Our previous investigations result in two distinct macros: WARP_BEGIN and WARP_INIT, which respectively mark the beginning of the warp-scope code portion, and correctly fill the warp identifier variable. When WARP_INIT presents no particularities, except the requirement to be called before any operations bringing into play warpIdx, WARP_BEGIN voluntary forgets the block-starting brace following the if statement. By doing so, we expect users to place both opening and closing braces of their WLP code if needed, just as they would do with any other block-initiating keyword.

To sum up, please note once again that this implementation mainly targets to validate our approach. Still, it lays the foundation of a more complete API dedicated to WLP. The efficient but not appealing intrinsic mechanisms are totally masked to users thanks to macros introduced in WLP.

RESULTS

In this part, we introduce three well-known stochastic simulation models in order to benchmark our solution. We have compared WLP’s performances on a Tesla C2050 board to those of a state-of-the-art scalar CPU: an Intel Westmere running at 2.527 GHz. For all of the three following models, each replication runs in a different warp when considering the GPU, whereas the CPU runs the replications sequentially. The following implementations use L’Ecuyer’s Tausworthe three-component PRNG, which is available on both CPU and GPU respectively through Boost.Random and Thrust.Random (Hoerberock and Bell 2010) libraries. Random streams issued from this PRNG are then split into several subsequences according to the Random Spacing distribution technique (Hill 2010).

Description of the models

First, we have a classical Monte Carlo simulation used to approximate the value of Pi. The application draws a succession of random points coordinates. The number of random points present in the quarter of a unit circle are counted and stored. At the end of the simulation, the Pi approximation corresponds to a ratio of the points in the quarter of a unit circle to the total number of drawn points. The output of the simulation is therefore an approximate of Pi value. This model takes two input parameters: the number of random points to draw and the number of replications to compute.

The second simulation is a M/M/1 queue. For each client, the time duration before its arrival and the service time is randomly drawn. All other statistics are computed from these values. The program outputs are the average idle time, the average time in queue of the clients and the average time spent by the clients in the system. Because it did not impact the performances, the parameters of the random distribution are static in our implementation. Only the number of clients in the system and the number of replications, which modify the execution time, can be specified when running the application.

The last simulation is an adaptation of the random walk tests for PRNGs exposed in (Vattulainen and Ala-Nissila 1995). The idea is to simulate a walker moving randomly on a chessboard-like map. The original application tests the independence of multiple flows of the same PRNG. To achieve this, multiple random walkers are run with different initializations of a generator on identically configured maps. Basically, each walker
computes a replication. In the end, we count the number of walkers in every area of the map. Depending on the PRNG quality, we should find an equivalent number of walkers in each area. When the original version splits the map in four quarters, our implementation uses 30 chunks to put the light on the opportunity of our approach when there are many divergent branches in an application.

Comparison CPU versus GPU warp

As we can see in Figure 5, the CPU computation time of the Monte Carlo application approximating the value of Pi grows linearly with the number of replications. The GPU computation time increases only by steps. This behaviour is due to the huge parallel capability of the device. Until the GPU card is fully used, adding another replication does not impact the computation time, because they are all done in parallel. So, when the board is full, any new iteration will increase the computation time. This only happens on the 65th replication because the GPU saved some resources in case a new kernel would have to be computed simultaneously. The same mechanism explains that after this first overhead, a new threshold appears and so on.

Due to this behaviour, GPUs are less efficient than CPUs when the board is nearly empty. When less than 30 replications are used, more than two-thirds of the board computational power is idle. Because sequential computation on CPU is widely faster than sequential computation on GPU, if only a little of the parallel capability of the card is used, the GPU runs slower. But when the application uses more of the card parallel computation power, the GPU becomes more efficient than the CPU.

![Figure 5: Computation time versus number of replications for the Monte Carlo Pi approximation with 10000000 draws](image)

The pattern is very similar for the second model: the M/M/1 queue (see Figure 6). When the board does not run enough warps in parallel, the CPU computation is faster than the GPU one. But with this model, the number of replications needed for the GPU approach to outperform the CPU is smaller than what we obtained with the previous simple model. The GPU computation is here faster as soon as 20 replications are performed, when it required 30 replications to show its efficiency with the first model. This can be explained by GPUs’ architecture, where memory accesses are far more costly than floating point operations in terms of processing time. If the application has a better computational operations per memory accesses ratio, it will run more efficiently on GPU. Thus, the GPU approach will catch with the CPU one faster.

This point is very important because it means that depending on the application characteristics, it can be adequate to use this approach from a certain number of replications, or not. A solution is to consider the warp approach only when the number of replications is big enough to guaranty that most of the applications will run faster.

![Figure 6: Computation time versus number of replications for a M/M/1 queue model with 10000 clients](image)

Comparison GPU warp versus GPU thread

If the advantages of WLP-enabled replications compared to CPU ones in terms of computation time have been demonstrated with the previous examples, it is necessary to determine if WLP outperforms the classic TLP.

This case study has been achieved using the last model introduced: our adaptation of the random walk. Figure 7 shows the computation time noticed for each approach: CPU, GPU with WLP and GPU with TLP (named thread in the caption). Obviously, CPU and WLP results confirm the previous pattern: the CPU computation time increases linearly when the WLP one increases by steps. TLP follows logically the same evolution shape as WLP. Although it is impossible to see it here because the number of replications is too small, it also evolves step by step, similarly to the warp approach. WLP consumes a whole warp for each replication. In the same time, TLP activates 32 threads per warp. Thus, the latter’s steps will be 32 times as long as
WLP’s. Having said that, we easily conclude that the first step in TLP will occur after the 2048th replication.

As we can see in Figure 7, the computation time needed by the thread approach is significantly more important than the computation time of the warp approach (about 6 times bigger for the first 64 replications). But WLP catches up with TLP when the number of replications increases. When more than 700 replications are performed, the benefit of using the warp approach is greatly reduced. The best use of the warp approach for this model is obtained when running between 20 and 700 replications. Please note that this perfectly matches our replications amount requirement. It even allows the user to run another set of replications according to an experimental plan, or to run another set of replications with a different high quality PRNG. The latter practice is a good way to ensure that the input pseudo-random streams do not bias the results.

![Figure 7: Computation time versus number of replications for a random walk model with 1000 steps (above: 100 replications, below: 1000 replications)](image)

These results are backed up by the output of the NVIDIA Compute Profiler for CUDA applications. The latter tool allows developers to visualize many data about their applications. In our case, we have studied the ratio between the time spent accessing global memory versus computing data. Such figures are displayed in Figure 8 for both TLP and WLP versions of the random walk simulation. Our approach obviously outperforms TLP, given that the ratio of overall Global Memory access time versus computation time is about 2.5 times bigger for TLP.

To explain this ratio, let us recall that computation time was lower for WLP. Since the same algorithm is com-

![Figure 8: Comparison of TLP and WLP ratio of the overall Global Memory access time versus computation time](image)

puted by the two different approaches, we should have noticed the same amount of Global Memory accesses in the two cases. In the same way, the profiler indicates significant differences between Global Memory reads and writes for TLP and WLP. These figures are summed up in Table 1:

<table>
<thead>
<tr>
<th></th>
<th>TLP</th>
<th>WLP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reads</td>
<td>225</td>
<td>18</td>
</tr>
<tr>
<td>Writes</td>
<td>302</td>
<td>104</td>
</tr>
</tbody>
</table>

Table 1: Number of read and write accesses to Global Memory for TLP and WLP versions of the Random Walk

**CONCLUSION**

This paper has shown that using GPUs to compute MRIP was both possible and relevant. Having depicted nowadays GPUs’ architecture, we have detailed how warp scheduling was achieved on such devices, and especially how we could take advantage of this feature to process codes with a high rate of branch divergent parts. Our approach, WLP (Warp-Level Parallelism), intends to allow users to easily distribute their experimental plans with replications on GPU.

WLP has been implemented thanks to simple arithmetic operations. Consequently, WLP displays a minimalist impact on the overall runtime performance. For the sake of user-friendliness, the internal mechanisms enabling WLP have been wrapped in high-level macros. At the time of writing, our version is functional and allows users to create blocks of code that will be executed independently on the GPU. Each warp will run an independent replication of the same simulation, determined by the warp identifier figured out at runtime. By doing so, we prevent performances to drop as they would do in a SIMT environment confronted to branch-divergent execution paths. WLP also tackles the GPU underutilization problem by artificially increasing the occupancy.

To demonstrate our approach performances, we have compared the execution times of a sequence of inde-
pended replications for three different stochastic simulations. Results show that WLP is at least twice as fast as cutting-edge CPUs when asked to compute a reasonable amount of replications, that is to say more than 30 replications. This will always be the case when a stochastic simulation is studied with a design of experiments, where for each combination of deterministic factors we have to run at least 30 replications, according to the previously mentioned Central Limit Theorem. WLP also overcomes the traditional CUDA SIMT performances by up to 6 to compute the same set of replications. Here, SIMT suffers of an underutilized GPU, whereas WLP takes advantage of a quick warp scheduling.

Insofar performances of WLP increase with the recent Fermi architecture compared to Tesla, we can expect this approach to be even more efficient with future CUDA architectures. We will validate this approach with bigger simulation models. As a matter of fact, two parameters need to be considered to determine how WLP will scale. On the one hand, a bigger model will often be more complicated, and will consequently contain much more divergent branches. When our approach should benefit of this aspect, on the other hand, bigger models will also consume more memory, which is the bottleneck of GPU devices.

The current version of WLP forces users to distribute their replications with our keywords. The target audience of our approach should, for the moment, be familiar with CUDA or GPU development. To lower the level of technical difficulty, we are currently thinking about an automatic tool, taking a simulation model and the number of replications to process in input, and producing the WLP equivalent in output, thus fully automating MRIP on GPU.

REFERENCES


SIMULATION BASED EVALUATION OF QOS-GUIDED SCHEDULING FOR DISTRIBUTED COMPUTING IN LARGE ENVIRONMENTS

Florin Pop

University POLITEHNICA of Bucharest, Faculty of Automatic Control and Computers, Department of Computer Science Spl. Independentei, 313, Bucharest 060042, Romania
E-mail: florin.pop@cs.pub.ro

KEYWORDS
Scheduling, Simulation, Performance Evaluation, QoS. Distributed Systems.

ABSTRACT
Scalability, reliability and fault-tolerance become important requirements for distributed systems in order to support distributed computation. A distributed system with such characteristics is called dependable. Large environments, like Cloud, offer unique advantages, such as low cost, dependability and satisfy QoS for all users. Resource management in large environments address performant scheduling algorithm guided by QoS constrains. This paper presents the performance evaluation of scheduling algorithm guided by QoS. The algorithms for QoS-guided scheduling are analyzed in order to satisfy users constrains considering in the same time independent capabilities of resources. This analysis acts like a profiling step for algorithm calibration. The performance evaluation is based on simulation. The simulator is MONARC, a powerful tool for large scale distributed systems simulation. The novelty of this paper consists in synthetic analysis results that offer guidelines for scheduler service configuration and sustain the empirical-based decision.

INTRODUCTION
The actual increasing interest in scheduling for heterogeneous distributed systems is due to the dimensions of some large scale applications, which makes inadequate a single parallel architecture to cover their needs for parallelism. When dealing with a computational Grid for parallel and distributed computing we have to efficiently exploit the computational power. In many practical cases, heterogeneous systems have proved to produce higher performance at lower cost than a single high performance computing machine (Xhafa and Abraham, 2010). Scheduling applications on wide-area distributed systems is useful for obtaining quick and reliable results in an efficient manner. Optimized scheduling algorithms for multi-criteria constraints are fundamentally important in order to achieve advanced resources utilization and considering scalability limits on hierarchical platforms (da Silva and Senger, 2011). The modern applications address many fields of activity like satellite data processing, medicine, and others. Understanding the timing behavior and users constrains of distributed applications gets more and more important because new real-time (like multimedia and health) applications require defined upper bounds for runtime performance, called deadlines, in order to provide application to application quality of service (QoS) (Chunlin and Layuan, 2007). In this context, the scheduling algorithms for distributed systems can be divided in two major categories, best effort based and QoS constraint based scheduling. The goal of the best-effort based scheduling algorithms is to minimize the execution time, ignoring other factors, such as the cost of accessing resources and QoS preferences (Wu et al., 2006). On the other hand the QoS constraint based algorithms attempt to maximize the performance, according to different QoS constraints.

Best effort algorithm can be chosen according to performance and the tasks that need to be scheduled from the following: hybrid heuristic, adaptive generalized scheduler, adaptive scheduling algorithm, heterogeneous earliest finish time, greedy randomized adaptive search procedure, simulated annealing algorithm, genetic algorithm, task duplication based scheduling algorithm for heterogeneous systems, dynamical critical path, fast critical path (Li and Ravindran, 2004). QoS algorithms consist of different classes: back-tracking algorithms, approximation algorithms, loss and gain algorithm, and Bio-inspired algorithms (genetic, immune, ant colony system algorithms) (Pop and Cristea, 2009).

The field of simulation was long-time seen as a viable alternative to develop new algorithms and technologies for distributed systems. Simulation represents a powerful support to enable the development of large-scale distributed systems, where analytical validations are prohibited by the nature of the encountered problems. The use of discrete-event simulators in the design and development of large scale distributed systems is appealing due to their efficiency and scalability. Their core abstractions of process and event map neatly to the components and interactions of modern-day distributed systems and allow the design of realistic scenarios. Compared with the alternative of implementing the new technology directly in real-world to demonstrate its viability, the simulation of distributed systems is a far better alternative because it achieves faster validation results, minimizing the costs involved by the deployment process (Dobre et al., 2009).
This paper presents a useful approach for analyzing the performance of scheduling algorithms for tasks with dependencies. Finding the optimal procedures for scheduling in Grid systems is important especially in large scale distributed computing systems and complex applications for different research areas.

The paper is organized as follows: next section presents the background provided by MONARC simulator for scheduling, and then the scheduling in distributed systems and QoS guided scheduling are presented. The papers end with test scenarios, experimental results, synthetic analysis and a brief overview of related work. Finally, we will conclude and will identify future works.

**SIMULATION MODEL FOR SCHEDULING**

MONARC is built based on a process oriented approach for discrete event simulation, which is well suited to describe concurrent running programs, network traffic as well as all the stochastic arrival patterns, specific for such type of simulation (Dobre et al., 2008). Threaded objects or “Active Objects” (having an execution thread, program counter, stack, etc.) allow a natural way to map the specific behavior of distributed data processing into the simulation program. With the MONARC simulation model, users can define various types of jobs to model common types of actions that can occur in any distributed systems.

The MONARC simulation model is not limited to specific activities, the user having the possibility to easily incorporate new advanced job behaviors, as specified in the simulation scenario being executed. A simulated regional center also uses the services of a job scheduler. In order to schedule a job for execution the scheduler executes an appropriate scheduling algorithm. The modeled job object contains a number of parameters that are used to estimate the time needed for execution. The time needed by a job to complete a CPU-intensive operation is estimated based on a number of attributes such as the CPU power, memory and the processing time needed to complete.

For the data processing jobs, these attributes depend on the type of data that the job works with (in the configuration file, the user can set this parameters for each data type used in the simulation). Once the CPU-intensive job starts processing the time needed to complete its operation is precomputed. If another job starts executing on the same processing unit before the first one completes, then an interrupt mechanism is used to handle the re-estimation of the time needed for both jobs. The time needed for an I/O intensive job (for example, a data transfer handling type of job) is based on the mechanism provided by the network model. In this case again an interrupt mechanism is used to simulate the competition for bandwidth usage of data transfer jobs.

Within the job model the user can define new jobs starting from the basic behavior provided. It can even combine several behaviors in one single composite job type. This aspect can be used to simulate a job that transfer some data, then processes it and in the end transfers back the obtained results. This behavior represents a composition between the processing and data transfer types of jobs and can be modeled using only five lines of codes. But, in the same type, the user can include a job which does all the data processing according to some advanced algorithm, extending in this case one method provided by the processing data type of job.

One interesting aspect is the job decompositions being offered by the job model. The user can specify a situation where a job requests some data, and then split itself in several parallel tasks, each one processing a particular chunk of data, and in the end the obtained results are reassembled and sent back. This fork-join programming paradigm can be modeled with the dependence mechanisms being offered.

Any job can instantiate new jobs. This means that, for example, one job receives the data, splits it into chunks and instantiate processing jobs, each one supplied with one specific chunk of data. It then specify the dependence, meaning it specifies what jobs must be executed after it finishes its own execution.

The dependency between jobs can be specified in the form of a DAG structure (Pop et al., 2009). The simulation model also allows the evaluation of advanced scheduling algorithms such as the ones we were particularly interested in, evaluating DAG Grid strategies. However, in order to accommodate the modeling of the DAG scheduling algorithms we had to extend this default behavior of the job model. The simulation model (task states, queues and event) for scheduling is presented in Table 1.

**SCHEDULING FOR DISTRIBUTED COMPUTING**

For task scheduling problem in distributed computing, considering dependencies for tasks, the model is bag-of-task with dependencies called DAG (Directed Acyclic Graph). In a DAG, a vertex (node) is the task and an arc is the communication constrain between two tasks. In a distributed system, the communication cost will be ignored if both tasks are run on the same processor. A schedule is an assignment of tasks (in the required order) on each processor. The goal is minimizing the makespan (or other function that is mention as a measure for QoS) of the schedule. Makespan represents the time elapsed between the start of the first task and the end of the last task and it is a good measure for QoS for scheduling problem.

The following algorithms give a suboptimal solution to the task scheduling problem. The trade-offs considered are minimizing makespan, running time of algorithm, number of processors and task communication costs (Amato and An, 2000). These algorithms were chosen because they are much closed to DAG scheduling.

*Wave Front Method (WFM):* The wave fronts of the graph are determined according to the level of the vertices in a
breadth-first-search traversal of the DAG. The vertices in each wave front are independent from each other, and are all assigned to different processors (Hamed and Nooraliee, 2009).

<table>
<thead>
<tr>
<th>FOR-each simulation task</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF the task is in the &quot;created&quot; state</td>
</tr>
<tr>
<td>Assign it to a worker thread from the pool and Change the state to &quot;ready&quot;</td>
</tr>
<tr>
<td>ENDIF</td>
</tr>
<tr>
<td>IF the task is in the &quot;ready&quot; state</td>
</tr>
<tr>
<td>Restart its execution</td>
</tr>
<tr>
<td>ENDIF</td>
</tr>
<tr>
<td>IF the task is in the &quot;finished&quot; state</td>
</tr>
<tr>
<td>Remove it</td>
</tr>
<tr>
<td>ENDIF</td>
</tr>
<tr>
<td>ENDFOR</td>
</tr>
</tbody>
</table>

Wait until all tasks that were running block again or finish their execution

Take, from the future queue, the event(s) with the minimum timestamp. The simulation time advances, becoming equal to that timestamp

<table>
<thead>
<tr>
<th>FOR-each event taken from the queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Look for the destination task</td>
</tr>
<tr>
<td>IF is waiting for an event (i.e. it is in the &quot;waiting&quot; state)</td>
</tr>
<tr>
<td>Deliver the event to the task</td>
</tr>
<tr>
<td>ELSE</td>
</tr>
<tr>
<td>Put the event into the deferred queue</td>
</tr>
<tr>
<td>ENDIF</td>
</tr>
<tr>
<td>ENDFOR</td>
</tr>
</tbody>
</table>

IF there are no more alive tasks and no more events in the queues |
Exit the simulation

| ENDIF |

**Critical Path Merge (CPM):** A critical path in a DAG is a maximum weight root to leaf path (the path weight is the summation of all vertex and edge weights on the path). CPM computes the critical path, clusters all tasks in it, assigns them to the same processor, and removes them from the graph. This process is iterated until all tasks are scheduled. This logic behind this algorithm is that getting all the nodes on the current critical path on the same processor removes the edge costs and the duration of the critical path itself. Also the duration of an infinitely parallelizable dependency graph will still be equal to the duration of the critical path so reducing it is a necessary condition for optimality (Wieczorek et al., 2009).

**Heavy Edge Merge (HEM):** Heavy Edge Merge works by iteratively clustering vertices (tasks) along edges with non-increasing weights. During an initialization stage, the edges are sorted in non-increasing order by edge weight, one task is assigned to each (virtual) processor, and the makespan of this assignment is computed. Then, all edges are processed in sorted order. For each edge, the makespan resulting from merging the tasks associated with the endpoints (perhaps clusters themselves) is computed. If the makespan increases, then the merge is not performed.

**Min-min heuristic:** Min-min heuristic uses minimum completion time (MCT) as a metric, means that the task which can be completed earliest is given priority. This heuristic considers a graph of tasks (G) and begins with the set U of all unmapped tasks. Then the set of minimum completion times tasks:

\[ M = \{ \text{min \_compl \_time}(M_i, T_1) \mid i, j \text{ in } G \} \]

is found. M consists of one entry for each unmapped task. Next, the task with the overall minimum completion time from M is selected and assigned to the corresponding machine. Then the workload of the selected machine will be updated and finally the newly mapped task is removed from U. This process repeats until all tasks are mapped.

**QOS-GUIDED SCHEDULING**

The QoS guided scheduling must to satisfy some requirements. These requirements could be asked by users or by resources. Regarding the importance of these requirements a QoS guided policy could be application centered, resource centered or balance.

The QoS request is showed by vector of QoS parameters described for each task \( T_i \), \( i = 1, 2, \ldots, N \) as

\[ q(T_i) = [q_{i1}, q_{i2}, \ldots, q_{ik}] \]

and the weights for the parameters as

\[ w(T_i) = [w_{j1}, w_{j2}, \ldots, w_{jk}], 0 \leq w_j \leq 1, j = 1, 2, \ldots, k \] .

All weights respect the normalization rule:
\[
\sum_{j=1}^{k} w_{ij} = 1
\]

In this model each weight is used to show the importance of each parameter. For example, if only CPU usage is important for one task, the user will set 1 for the CPU weight and zero for the others.

The large distributed environments are in a continuous and dynamic change. When the resource capabilities are changed all the scheduling procedure must consider this new state of the system. Let us assume that a distributed infrastructure consists of \( M \) resources. The capabilities of a distributed resource \( R_i, i = 1, 2, \ldots, M \) is expressed with the resource parameter vector as:

\[
r(R_i) = \begin{bmatrix} r_{i1} & r_{i2} & \ldots & r_{ik} \end{bmatrix}
\]

The \( r_{ij} \), \( j = 1, 2, \ldots, k \) elements indicate independent capabilities of the \( j \)-th resource that affect its performance.

The number of QoS constrains that could be addressed to a resource is equal with number of resource capability, so the \( q \) and \( r \) vector have the same size, \( k \).

In this model, \( q_j \) and \( r_j \) have the same unit, and the ECA (Event Condition Action) resource manager rules (that exists in a distributed system) compare \( q_j \) with \( r_j \) for each \( j \) from 1 to \( k \). If the resource provides the requirements needed for the task, it can be chosen as the best matched resource. We introduce satisfy operator:

\[
\leftrightarrow: R \times T \rightarrow \{0, 1\}.
\]

\( R_i \leftrightarrow T_j \) means that the resource \( R_i \) can satisfy the task \( T_j \) and guarantee QoS parameters. We introduce a Threshold in order to define the satisfy operator as:

\[
R_i \leftrightarrow T_j = \left( \sum_{j=1}^{k} \left( \frac{r_{ij}}{q_{ij}} \times w_{ij} \right) \geq \text{Threshold} \right)
\]

As we mention, \( k \) is the number of QoS parameters. The possible values for this operator could be 0 or 1. For a valid schedule that respects all QoS constrains the satisfy operator must be 1 for all tasks assignments to resources.

According with this QoS model, it is quite difficult to make a comparison among different scheduling systems, since each of them is suitable for different situations. For different scheduling systems, the class of targeted applications (that are composed by multiple tasks with dependencies) and resource configurations may differ significantly. In this subsection, a number of evaluation criteria for Grid scheduling systems are proposed: application performance promotion, system performance promotion, scheduling efficiency, reliability, scalability, and applicability to applications and environments.

When designing the scheduling infrastructure of a Grid system, these criteria are expected to receive careful consideration. Emphasis may be laid on different concerns among these evaluation criteria according to practical needs in real situations. There are some objective functions that could be used in order to optimize the process of scheduling. These functions could be a key order to satisfy the QoS constrains. There are bottleneck functions (for instance the makespan and the maximum lateness) and sum/mean functions. The latter ones may also appear in: mean/sum of completion time, mean/sum of weighted completion time, mean/sum of flow time, mean/sum of weighted flow time, mean/sum of tardiness, number of late tasks and total weight of late tasks.

**TEST SCENARIOS, EXPERIMENTAL RESULTS AND SYNTHETIC ANALYSIS**

Using MONARC’s extensions we proceeded to evaluate the scheduling algorithms in order to satisfy the QoS constrains discussed in this paper. We were particularly interested in analyzing their performance considering the use of two realistic scenarios. For this reason the modeling experiment considered the use of two and eight connected processors and a set of tasks with dependencies. The communication costs were considered between 0 and 20 and the tasks execution time were considered between 0 and 40.

The evaluated scheduling strategies were: Wave Front Method (WFM), Critical Path Merge (CPM), Heavy Edge Merge (HEM) and Min-min heuristic (MIN), discussed in this paper.

In our experiments, in order to satisfy the QoS constrains and to have the value 1 for satisfy operator for all assignments, we exclude all un-matching possibility. The makespan (as maximum execution time provided by scheduler) and logarithmic runtime (measured after tasks execution) were considered in order to compare the performances op evaluated scheduling strategies.

The results for presented scenarios are represented in Figures 1-4. The quantitative analysis of these graphics shows that CPM, MIN and WFM are simple and efficient. WFM has good results because the input graph was artificially built. HEM gives good schedules but takes a lot of time. Depending of the particularities of the input graph, each algorithm behaves strongly or weakly. As a direct observation, the site manager of a cluster must adapt algorithms (set-up) to the problem and this paper gives you the necessary insight.

**RELATED WORK**

Regarding simulation as a tool for scheduling evaluation there are multiple research projects and papers in the last ten years. Alea simulation is based on the GridSim simulation toolkit which was extended to provide a simulation environment that supports simulation of varying Grid scheduling problems. Alea demonstrates the features of the GridSim environment implementing an experimental centralized Grid scheduler which uses advanced scheduling techniques for schedule generation (Klusáček et al., 2007). Li and Buyya (2009) present statistical models that are able
to reproduce various autocorrelation structures, including pseudo-periodicity and long range dependence. By conducting model-based simulation they quantitatively evaluate the performance impacts of workload autocorrelations in Grid scheduling. The results obtained indicate that autocorrelations result in system performance degradation, both at the local and the Grid level. Few years ago, Phatanapherom et al. (2003) sustain that to develop grid scheduling algorithms, a high performance simulator is necessary since grid is an uncontrollable and unrepeatable environment. They propose a discrete event simulation library called HyperSim is used as extensible building blocks for grid scheduling simulator. The use of event graph model for the grid simulation is proposed. This model is well supported by HyperSim which yields a very high performance simulation. Fu and Fan (2010) focus in their paper on how to schedule a system with distributed resources for multiple task execution. They explore the dynamic scheduling method for the parallel tasks with dependencies in distributed environments.

CONCLUSIONS

Simulation is a very powerful tool, and now maybe the only one, considering the complexity (and cost !) of Grid systems. In Grid environments, it is hard and even impossible to perform scheduler performance evaluation in a repeatable and controllable manner as resources and users are distributed across multiple organizations with their own policies. In addition, Grid test-beds are limited and creating an adequately-sized test-bed is expensive and time consuming.

The aim of the experiments was to evaluate a few scheduling algorithms (for task without and with dependencies) in order to measure a QoS constrains (like makespan). It is very hard to compare these algorithms because there are many different assumptions and conditions from which some of the scheduling algorithms start. Tasks with DAG dependencies are frequent in case of Grid applications and they require advanced scheduling procedures that must consider QoS requirements. In this chapter was proposed a simulation-based solution to evaluate the performances of Grid scheduling algorithms. The results could be used in decisions regarding optimizations to existing Grid DAG Scheduling and for
selecting the proper algorithm for DAG scheduling in various actual situations. The main contribution of the presented research consists in the development of the simulation layer in MONARC that is appropriate for DAG scheduling algorithms evaluation. It was introduced a set of recent algorithms and presented the solution to evaluate DAG scheduling algorithms using a generic simulator for large scale distributed systems guided by QoS constrains.

In this field, future work will include, among other things: the analysis of a wider set of scheduling algorithms currently used in Grid systems; the establishment of relevant performance measures and an improved simulation model; the evaluation of the current scheduling algorithms, using the chosen model. We will consider new scheduling algorithms for real-time scenarios, solutions for backup and recovery from error (re-scheduling) and solving the problem of co-scheduling and multi-criteria constraints scheduling.

ACKNOWLEDGMENTS

The research presented in this paper is supported by national project SERAFIMO, ID: 12113 and co-supported by national project "SORMSYS - Resource Management Optimization in Self-Organizing Large Scale Distributes Systems", Contract No. 5/28.07.2010, Project CNCSIS-PN-II-RU-PD ID: 201. The work has been co-funded by the Sectorial Operational Program Human Resources Development 2007-2013 of the Romanian Ministry of Labor, Family and Social Protection through the Financial Agreement POSDRU/89/1.5/S/62557.

REFERENCES


BIOGRAPHY

FLORIN POP (Bachelor’2005–Master of Science’2007–PhD’2008) is a lecturer in the Computer Science Department of the University Politehnica of Bucharest. His research interests are oriented to: scheduling in Grid environments (his PhD research), distributed system, parallel computation, communication protocols and numerical methods. He received his PhD in Computer Science in 2008 with “Magna cum laude” distinction. He is member of RoGrid consortium and participates in several research projects, in collaboration with other universities and research centers from Romania and from abroad developer. Florin Pop has important contributions in satellite image processing in Grid systems. Main
results in this domain are: design and implement an algorithm for segmentation, classification and change detection in satellite images like MODIS. Florin Pop receives a few awards for his scientific and research activities: "Best Paper Award", on "3PGCIC 2010, International Conference on, P2P, Paralel, Grid, Cloud and Internet Computing" for paper A fault-tolerant approach to storing objects in distributed systems; IBM PhD Assistantship 2007 (Top ranked 1st in CEMA out from 17 awarded students). Project Title: Optimization of Decentralized Scheduling Strategies in Grid Environments; Oracle Award 2007-2008 for his significant results in his PhD Studies.

E-mail: florin.pop@cs.pub.ro, Web-page: http://florinpop.ro
A GENERIC METHODOLOGY TO DEVS PARALLEL AND DISTRIBUTED SIMULATION

Adedoyin Adegoke
African University of Science and Technology, Abuja
Nigeria
E-mail: aadegoke@aust.edu.ng

Mamadou K. Traoré
Université Blaise Pascal, LIMOS,
CNRS UMR 6158,
Clermont-Ferrand 2, France
E-mail: traore@isima.fr

KEYWORDS
DEVS, Parallel and Distributed Simulation, Tree, Process, Processor

ABSTRACT

DEVS (Discrete Event System Specification) is a Modeling and Simulation methodology that can be used to perform both sequential and distributed simulations. In so doing, it uses the Parallel and Distributed Simulation (PADS) scheme. This paper proposes a methodology which gives a description of performing simulation with DEVS PADS. This is achieved by identifying the components and operations common to works in the area of DEVS PADS. Also by proposing a generic approach for applying DEVS PADS for Modeling and Simulation. This generic approach is necessary as it provides a basic foundation for works in DEVS PADS to be built on.

1. INTRODUCTION

The development of computers and latest technology trends (especially in the area of parallel and distributed computing) have presented alternative methods for effective distribution of simulation (High Performance Clusters, Grid, Cloud computing) by offering major economic benefits and scalability. Therefore, there is a need to exploit the computing power of nowadays technologies by distributing simulation on multiple processors. This is to reduce execution time, improve performance, perform real time execution, and integrate simulators.

A general Modeling and Simulation methodology for describing discrete-event systems (also continuous systems after quantization) is the Discrete Event System Specification (DEVS), a platform for the modeling and simulation of sophisticated systems in a variety of domains. As the system under study becomes more and more complex, the simulation tends to be increasingly time-consuming and resource-demanding. In the quest for better performance, PADS (Fujimoto 1990) is widely accepted as one of the best choices to speed up discrete-event simulation systems. There are some advantages in using PADS. First, the use of parallel processors promises an increase in execution speed and a reduction in execution time. Second, the potentially larger amount of available memory on parallel processors will enable the execution of larger simulation models. Third, with the use of multiple processors comes an increased tolerance to a possible processor failure. Thus, simulation architecture

can be called parallel if its main design goal is to reduce execution time while the term distributed simulation could be referred to as inter-operating geographically dispersed simulator (Zeigler et. al. 2000).

PADS is a matured field of study but its adaptability to DEVS (Zeigler et. al. 2000) simulation formalism is difficult to implement. Also, there is no guideline, help or open library for performing simulation with DEVS PADS. However, a methodology can be very helpful to take into account the variety of possible DEVS PADS implementations. To achieve this, in this paper we propose this methodology and an underlying meta-model.

The rest of the paper is organized as follows: Section 2 presents a background of the DEVS simulation system and operations in DEVS PADS. Section 3 then presents the methodology and its meta model. Section 4 presents a discussion about this methodology. Finally, we conclude in Section 5.

2. KEY CONCEPTS

In this section we give brief definitions of some of the concepts in use in this paper.

• Root Coordinator is a special simulating element that drives the global aspects of the simulation on a tree; it initializes and ends the simulation (when a termination condition is detected).

• Nodes: They are simulation entities used for executing DEVS models. These nodes are Coordinators, Simulators and Root Coordinators. The Root Coordinator has an event loop which sends event messages and controls the simulation cycles while the Coordinator and Simulator are capable of receiving, treating and sending event messages.

• Simulation Tree: A tree is made up of nodes. The Root Coordinator is always at the top of the Tree’s hierarchy and has a Coordinator as its descendant. Also, the Coordinator has either a Coordinator or Simulator as its descendant but the Simulator has none.

• Process: We define it as a stream of execution. It contains two types of entities during execution they are active and passive entities. An active entity is an entity that is currently active in an execution stream e.g. Java threads, ADA Tasks. While a passive entity is part of an execution stream but not actively involved until it is triggered e.g. function calling in Object Oriented Paradigm. We consider that a process would have at most one active entity. If a process has more than one active entity, those
entities are then regarded as being autonomous sub-processes. Also, there can be more than one passive entity in a process.

- Activity: Set of actions that are performed at the receipt of an event
- Processor is a computing resource that allows the execution of a program (a process, an entire tree, any other executable code) on itself.
- Simulation Graph: A representation of the relationship between the identified aspects in DEVS simulation.

3. DEVS PADS

The focus of this section is on the operational features we were able to identify in DEVS PADS. With this, we describe what can be done when utilizing DEVS PADS for modeling and simulation and not how to do it.

3.1. DEVS Simulation Protocol

DEVS formalism (Zeigler et. al. 2000) provides a comprehensive modeling and simulation framework for modeling and analysis of Discrete Event Systems. It specifies system behavior as well as system structure. System behavior in DEVS is described as input and output events as well as states while system structure is built from the composition of atomic or coupled models. The DEVS Simulation Protocol uses a Tree-like representation as its basis. The distinctiveness of the DEVS framework is in its hierarchical compositional structures which help in complexity reduction. A DEVS coupled model is composed of several DEVS (atomic or coupled) models and a DEVS atomic model is a basic component that cannot be decomposed any further.

There are algorithms (abstract simulator) which consists of different simulating entities or Nodes and are used in the execution of models. They are organized in a hierarchy that imitates the hierarchical structure of a model. In these algorithms, a DEVS atomic model is executed by a Simulator and a DEVS coupled model is executed by a Coordinator. See Figure 1. Also included in the abstract simulator is the Root Coordinator.

3.2. Tree Transformation

It has been observed that altering a DEVS simulation tree structure can improve simulation performance and also enhance distribution. This transformation is usually achieved either by reducing the number of Nodes of a tree or increasing their number.

As presented in (Kim et al. 1996) the hierarchical structure of the simulator (which has a one-to-one correspondence with the DEVS model architecture) can increase the communication overhead between Nodes. The process of reducing the number of these entities on a tree is also known as flattening. In CD++ (Wainer 2009), the expansion was achieved by introducing new simulation entities into the simulation tree structure. This is to enable the distribution of simulating entities on different processors.

3.3. Tree Splitting

Tree splitting can be referred to as the decomposition of a simulator tree to form sub-trees based on the analysis of the model’s structure. We identified two types namely (based on multiplicities), single tree structure and multiple tree structure. Figures 4(a) and 4(c) are examples of a single tree structure and a multiple tree structure respectively.

Executing a model with a single tree structure can be expressed as having an entire model tree simulated with the use of a central scheduler called the Root Coordinator. Root Coordinator is a special Coordinator that drives the global aspects of the simulation tree; it initializes and ends the simulation (when a termination condition is detected).

Single tree structures are mostly implemented using Classic DEVS (CDEVS) and Parallel DEVS (PDEVS) algorithms. In CDEVS (Zeigler et. al. 2000) events are processed in a sequence. This approach is the simplest form of simulation but it is rigid and does not properly reflect the simultaneous occurrence of events in the system being modeled. Also, serialization reduces possible utilization of parallelism during the occurrence of events. On the other hand, Chow and Zeigler (Chow and Zeigler 1994) introduced PDEVs as a possible solution to the problem of serialization. According to Chow, one desirable property provided by PDEVs is the degree of parallelism which can be exploited in parallel and distributed simulation. It beats the restrictions in CDEVs in both execution time and memory usage. PythonDEVS (Jean-Sébastien and Vangheluwe 2002) uses this mapping strategy as an implementation of the CDEVS formalism and as a consequence it performs sequential simulation.

We look at the multiple tree structure as when a tree can be split into different sub-trees with each having its own Root Coordinator and different simulation clocks. This is the preferable solution in distributed simulation. It uses asynchronous algorithms which rely on the distributed synchronization protocols for synchronization instead of the Root Coordinator.

The two basic asynchronous algorithms in use are the Optimistic and Conservative (Pessimistic) DEVS algorithms. In distributed simulation, there is an inherent constraint in the time-stamp order (not in their real-time arrival order) with which events occur and are processed, this is called locality.
constraint. Some algorithms were proposed to resolve this constraint, the Conservative (Byrant 1977; Chandy and Misra 1978; Misra 1986) algorithm which always prevent this constraint through the use of look-ahead mechanism and Optimistic (Jefferson 1985) algorithm which detect and resolves it through the use of the roll-back mechanism. Optimistic algorithms, in contrast to Conservative algorithms, enable increased degrees of parallelism and do not depend on application-specific data to decide on events that are safe for processing. 

DEVS-Ada/TW (Christensen and Zeigler 1990) is the first attempt to combine DEVS and Time Warp mechanism for Optimistic Distributed simulation. In this work, the hierarchical DEVS model is partitioned at the highest level of the hierarchy and as a consequence, the flexibility of partitioning models is restricted.

The DOHS (Distributed Optimistic Hierarchical Simulation) scheme proposed by (Kim et al. 1996) is a method of distributed simulation for hierarchical and modular DEVS models. It uses the Time Warp mechanism for global synchronization. Each node of the simulation tree structure is revised to adapt to a simulation parallel/distributed environment.

Contrary to optimistic approaches, few parallel DEVS simulators belong to the conservative class. In (Zeigler et al. 2000), a distributed simulation framework (Conservative Parallel DEVS Simulator) is described for non-hierarchical DEVS models using conservative synchronization. In addition, the performance of a conservative approach depends strictly on a good look-ahead.

3.4. Process Clustering

DEVS trees make use of logical processes in driving the execution of DEVS models. A one-process execution denotes having events processed in a serially and orderly manner i.e. one after another. This restricts concurrent execution streams. (Himmelspach 2006) denotes this form of execution as “sequentialization”. A desired speed up may not be achieved when using a one-process execution stream but on the other hand it is easier, safer and faster to implement.

Although using many processes could speed-up execution as each could execute events without interrupting other processes but not in all cases. For example, let’s take a situation whereby at each simulation run more processes are created and are allowed to exist throughout the entire simulation process. They would consume more memory thereby slowing down simulation time and increase the cost of communication between them. This type of communication is called inter-process communication. During implementation it is essential to manage how processes access resources that are common to all of them e.g. shared data type. Locks, Semaphores, Monitors and other synchronizing mechanisms can be used to coordinate these processes. The CCD++ (Jafer and Wainer 2009) implementation of DEVS formalism utilizes many processes for model execution.

3.5. Processor Mapping

We considered that the number of processors play a major role in speed, performance and efficiency that can be achieved during simulation. We therefore categorize this into 2 distinct classes; “one-processor” or “many-processors”.

On a uniprocessor system, the entire program runs on one processor so there is gain in time due to the reduced overhead cost, it is however limited to the size of the memory in use. The type of communication that takes place in this case is called an intra-processor communication.

In order to coordinate simulation on many networked processors, some form of inter-processor communication is required to convey information and data between processors and synchronize each processor’s activities. The memory architecture in many processors is either shared memory (processors have direct access to common physical memory), or distributed memory. Meanwhile, in shared memory only one processor can access the shared memory location at a time and synchronization is achieved by controlling access to it. Distributed memory refers to the fact that the memory is physically distributed as well. Memory access in shared memory is faster but it is limited to the size of the memory. Therefore, increasing the number of processors without increasing memory size can cause severe bottlenecks. Inter-processor communications is usually achieved through distributed communication mechanisms (MPI, HLA, and so on).

As a consequence of using more than one processor, the Nodes can be split into a set of partition blocks based on certain decision criteria and mapped unto the available number of processors, this is called partitioning. In the case of no partition, simulation is performed on a single processor machine. Different partitioning algorithms have been proposed an example is Generic Model Partitioning (GMP) algorithm (Park et al. 2006).

3.6. Simulation Graph

We interpret the construction of a DEVS Parallel and Distributed Simulation (PADS) for simulation as a move from the original DEVS Simulation Tree (ST) to a Simulation Graph (SG). A SG is obtained through the depictions of the relationship between Trees, Processes and Processors. The SG structure gives the complete form of a typical DEVS PADS scheme.

The basic idea behind a SG strategy for DEVS PADS is the elements it depicts and their multiplicities. As shown in Figure 2 which is an example of a SG strategy; it uses many Trees, many Processes and many Processors.

4. METHODOLOGY AND UNDERLYING META MODEL

In this section we propose a methodology which attempts to provide a basic standard for performing modeling and simulation with DEVS PADS. This methodology thus describes a structural and behavioral view in exploiting DEVS PADS.
which shows the major elements used in DEVS PADS simulation (i.e. Tree, Process, Processor).

(Figures 2: Simulation Graph)

4.1. Methodology

By using a state chart, we present the trajectories which describe the set of all possible paths that can be taken during the construction of the Simulation Graph (SG). It is worth noting that the methodology iterates on each state until some user-defined satisfaction criteria are reached (optimal splitting, optimal clustering, optimal mapping and optimal transformation).

Each state of this methodology consists of a Simulation Structure. A Simulation Structure is realized as a result of an operation and a refinement from the previous Structure.

(Figure 3: Generic Methodology)

The Simulation Skeleton is the structure of the simulation protocol that can fit the PADS scheme and it is obtained after the Simulation Tree has been split. The Simulation Bundle is a collection of cluster of Nodes. These clusters are bundled into the available number of Processes.

This methodology uses a layered approach in its representation, see Figure 5. As we move up the layers, each Simulation Structure inherits the DEVS PADS element of the Simulation Structure beneath it. The Simulation Tree consists of just the Tree and the Simulation Graph combines all the elements used in DEVS PADS i.e. Tree, Process and Processor. The benefit of this approach is that it gives a
straight and clear guideline for realizing DEVS PADS by avoiding the possible pitfalls. For example, to modify the Simulation Graph for adaptation to a new computing architecture, starting from the SG layer there would be a need to move to the layer below it or back to the Simulation Tree before making the necessary adjustments. Using this layered approach proposed in this methodology, this way, errors can be easily detected and corrected.

Figure 5: Layered Approach of the Methodology

4.2. Meta Model

In this section we describe a meta-model (Figure 6) using Unified Modeling Language (UML). This language is used to explicitly define each of the objects required in the construction of a DEVS PADS simulator.

PADS with DEVS have been identified in this paper to majorly consist of 3 components i.e. Trees, Processes and Processors. At the top of the Simulation Tree hierarchy is the Root (i.e. Root Coordinator) which coordinates the entire simulation and has an Abstract Simulator (Coordinator or Simulator) as a descendant. A Coordinator is composed of subcomponents which are AbstractSimulators as well. The Root and Abstract Simulator are simulating entities or Nodes which belong to atleast a Process. These Nodes interact through Synchronization mechanisms i.e. at the receipt of messageType they execute a set of actions contained in the method called Activity. One or more of these Processes are mapped onto available number of Processors.

DEVS models, Coupled (which contain Models as well) and Atomic, are executed by an AbstractSimulator. However, the constraints and relationship restricts them to being used for execution during simulation.

5. DISCUSSION

Having taken a look at what can be done when utilizing DEVS PADS for simulation, the Simulation Structures and the operations that can be performed on them; we discuss some of the potential benefits of this methodology.

Figures 6: Meta Model for DEVS PADS
A simple and efficient guideline
The whole essence of this paper is to provide a guideline, help or open library for performing simulation with DEVS PADS. This has been achieved through the proposed methodology which also uses a layered approach in its representation. This approach provides a common frame of reference for performing simulation with DEVS PADS. Also it promotes modularity and Simulation Structure reuse.
- A basis for introducing the evaluation of DEVS PADS strategies.

As it has been observed that there are different practices behind the concept of exploiting DEVS PADS which usually results to obtaining a Simulation Graph. This whole idea has been found to be based on varying the number of elements (i.e. Trees, Processes, and Processors). For example, in Parallel CD++ (Troccoli and Wainer 2003), the Simulation Graph Strategy used consists of a Tree, many Processes and many Processors while the Abstract Threaded Simulator of the James II (Himmelspach and Uhrmacher 2006) package a Tree, many Processes and a Processor and Conservative CD++ (Jafer and Wainer 2010) uses many Trees, many Processes and many Processors.

With the various practices that have been observed, they introduce the need for evaluation. This would be used to determine which of the obtained Simulation Graph more efficient or which would give an improved performance and so on. This methodology therefore gives a foundation for achieving this through its generic nature.

A meta model that can be supported by a software tool.

Based on the DEVS formalism, the SimStudio aims at providing a complete Modeling and Simulation framework (Traoré 2008; Touraille et al. 2011). The meta model can be integrated into this framework as it (SimStudio) provides a Simulation Layer which supports the DEVS PADS scheme and an Automation Layer which provides automatic code synthesis. In the SimStudio framework, the meta model can be supported as the framework uses a modular software architecture which relies on plugins which create a means for us to develop each operation found in the methodology as a plugin to the framework.

6. CONCLUSION

This paper is part of a more general research direction that we investigate DEVS PADS to help practitioners from various simulation domains comprehend the concept. In this paper, we presented a methodology which shows the basic elements and the operational features that are used for modeling and simulation with DEVS PADS. This is essential as it helps to provide a basic guideline for utilizing DEVS PADS.

By identifying the common Simulation Structures and the operations which can be performed on them, we were able to introduce the layered approach used by the methodology. Also, through the conception of a meta-model which serves as the backbone for the methodology we were able to show a framework that supports DEVS PADS scheme.

Further works include providing formal definitions for each Simulation Structure. Also, we intend automating this methodology and its meta model in SimStudio Framework Architecture (Traoré 2008; Touraille et al. 2011).

REFERENCES


BUSINESS SIMULATION
KEYWORDS

ABSTRACT
Effective business process description is one of the most important tasks to achieve the effectiveness of the company. The typical problem that arises when describing the processes to stakeholders is the lack of understanding to the formal business process notations and definitions. That is why we need a business modeling method that is easy to understand for a stakeholder and that has a formal background for an analyst. In this paper we are extending the concept of storyboards. Storyboard is a sequence of pictures and multimedia that represent one possible flow of the process. Each picture represent process activity performed by actor and is described by multimedia and/or text description. This paper deals with creating a structured image that will automatically capture the essential links between artifacts. These links can then be used in the subsequent description of the transformation process into formal languages such as UML, BPMN.

INTRODUCTION
Managing a company is not easy. Managing the company effectively is even harder. Description and effective enactment of company business processes is one way how to do management effectively. Business processes are formally described, enacted and then controlled by managers. Everything works effectively and everybody is satisfied. The problem arises when the company processes must be presented to the people that are not familiar with the standard business modeling methods and notations. Communication between business process analyst and stakeholders of the processes is important part of the process description. Process analyst has to be introduced to the company environment. Company representatives and employees must understand the description of the processes. Common language is the notation or tool that is used for the process description. Business analysts are using many different formal or informal methods and notations that are usually not easy to understand. Thus, the stakeholders of the processes have to learn new language/notation to understand the company processes to be able to perform them.

In our opinion, more effective way is to develop and use an intuitive method that is very easy to understand, is formal and can be automatically transformed to standard modeling notations. This transformation request is important for the usage of methods that are already used by process analyst to analyze and simulate business processes. This paper describes the extension of storyboard method (Jůchová 2010) that is being developed at our university. In following section we shortly describe our understanding of the business modeling. In next sections we introduce our extension to storyboard method. Last section describe meta-model for our Storyboard method extension.

BUSINESS MODELING
What is the business modeling? Business modeling is the activity that is performed by the business analysts. Analysts try to model the current state of the enterprise business and also its predicated future state. Business process modeling (BPM) tools are used to do that. BPM tools capture the business flow of activities by the business processes and describe the many additional pieces of information by other techniques (Laguna, 2004; Penker, 2000; Vondrak, 1999). The purpose of the business modeling is to model the reality of the enterprise business. Models could be used then in many ways. For example for the Business Process Re-engineering (BPR) that are the methods that support activities by which an enterprise reexamines its goals and how it achieves them, followed by a disciplined approach of the business process redesign. Business modeling is also the first step in the process of Enterprise Resource Planning (ERP) information systems implementation.

STORYBOARD METHOD
The aim of storyboard method is to facilitate creating a process description in a form understandable to stakeholders and then allow the automatic acquisition and transform of certain information in the formal language which can be the basis for creating a formal description of process or simulation model of process.

The basic concept of the storyboard method was presented in (Jůchová 2010). Storyboard is the multimedia picture frame that contains sequence of successive shots. Each shot represents one business process fragment – one action that is performed in business process. Actions are sequenced the same way as they are performed in the business process. Audio or video files can be added to each action shot. Each storyboard (a sequence of shots - Figures 1) captures the possible execution of process. Every shot (action process) is described by an illustrative picture of activity, name, description, list of input and output artifacts, and other attributes. These data can be used to semi-automatic
assembly of process using Higher Dimensional Automata (HDA).

Figures 1: Sequence of successive shots

**Extension of Storyboards by Structured Shot**

The description of process by storyboard contains lot of information, but this information is visible only for humans not for computer programs. If we use process description as input for specification of information system or any simulation tools we must enter this information to computer in suitable form again.

Improvement of this method is obtaining data from the images themselves in computer understandable form. Such a data can be transformed and imported to any other application. It is not about any automatic image recognition, but the creation of structured scenes from predefined parts. Graphic form of presentation and capture information can be a very useful in many aspects (Roam 2008), but creating or searching for a suitable image in the normal way is very challenging. Furthermore, information contained in an image usually cannot be automatically processed.

However, we can observe that the various images of activities contain similar elements that are placed in the context of the background image along with other elements. Take for example the activity "Prepare goods for shipment" for which storekeeper is responsible, which search for goods from "Order" in the warehouse, put it in boxes and confirm "Order" when the goods are ready for shipment (See Figures 2).

<table>
<thead>
<tr>
<th>Activity name</th>
<th>Prepare goods for shipment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process name</td>
<td>Order fulfillment</td>
</tr>
</tbody>
</table>

**Scenario**

Storekeeper is responsible for the preparation of goods that are specified by the filled-out order. As soon as the goods are collected and prepared, package can be sent. Whole procedure of preparing goods lasts approximately 30 minutes. Stores foreman can keep an eye on storekeeper during his work.

**Roles**

Storekeeper

**Inputs**

Order [filled-out]

**Outputs**

Order [prepared], Package [prepared for shipment]

**Executing time**

30 minutes

**Access privileges**

Storekeeper, Stores foreman

Figures 2: Activity Prepare goods for shipment shot description

Picture for this activity should include a storekeeper, goods, order, goods packed in a package, a confirmed order. All these elements should be placed in warehouse environment (See Figures 3 – Figures 8).

Figures 3: Order Figures 4: Goods Figures 5: Goods packed in a package

Figures 6: Confirmed order Figures 7: Storekeeper

Figures 8: Warehouse environment

If these elements are put together into a single image, arises clear and illustrative picture of activity "Prepare goods for shipment". Now we can from the structure and method of creating the image, identify artifacts that are used in the activity and who and where performs this activity.

Figures 9: Picture for activity “Prepare goods for shipment”

If we use another extension of individual elements from which the image is composed, we can get more information about relationship between artifacts. It can be noted that,
based on mutual positions of individual elements, we can infer relationships between elements. If we determine areas for all elements that are important to infer the relationship between elements, we can derive these relationships automatically. In our case, it makes sense to determine these areas for two elements: Warehouse environment and Storekeeper. Storekeeper as well as any other person will have three areas (Figures 10). Two of them are at hands and suggest that the Storekeeper uses elements in these areas and the third is over head and its suggest that the Storekeeper thinking or solves about that element in this area. Warehouse environment determine only two areas (Figures 11), one area for responsible worker and one for the output elements. This allows determine who is responsible for the activity in the warehouse and what the output artifacts are.

![Figures 10: Defined areas for Storekeeper](image)

![Figures 11: Defined areas for Warehouse environment](image)

**STORYBOARD METHOD META – MODEL**

Based on these defined properties for the generated images we define a meta-model for method storyboard, which will allow capture description of processes through storyboards (shot sequence) as described in (Jůchová 2010), and also to capture the structure of images for individual shots. These meta-model allow us to capture more information in formal form.

Structure of meta-model is shown in Figures 12. Base of meta-model are the images represented by the class Shot. The sequence of images defines one of the possible process executions (class ProcessPath). The set of possible executions describes the entire process, which is represented by the class Process. This model allows you to capture branching processes, which is realized through a two execution of the process, which starts with the same sequence of shots, and start differ at the branching point. In order to capture the concurrence of two actions is established class ParallelExecution which specialized class Shot and consists of several objects of class Shots, which can run in parallel.

Shot himself, whose creation is the main subject of this paper, consists of individual parts (class ShotPart). In order to create a hierarchical structure of parts there exists self-binding in the class ShotPart. This link ensures that each part can be specialized by other versions. For example, part "Form" may be a specialized by part "Order", "Invoice" or "Payment Order". This binding, also ensures that each part can have more than one generalized part. For example, "Payment Order" is a specialization of part "Form" but also of "Internal Document".

![Figures 12: Meta-model structure](image)

Each part can contain several areas (class RelationshipArea) which represent the connectivity to the other parts. Each region has association back on the part (ShotPart). Each area can also define several semantic meanings that determine the meaning of relationship. Class RelationshipArea includes association "restricted to" that ends in the class ShotPart, which limits what parts or it specialization can be used in this relationship.

**FUTURE WORK AND CONCLUSIONS**

This paper presents an extension of storyboard method that allows preserve information depicted by image of storyboard shot for future use in subsequent steps. Currently we work on a prototype application that allows us to create a description of the processes using the storyboard method and will enable us to create a picture from various pre-defined parts. Subsequently, we developed transformation procedures, which allow us to transform the captured relations into a form that is usable for example in the simulation process by using ontologies.

**REFERENCES**


TRUST MODEL FOR SOCIAL NETWORK

Arnostka Netralova and Jiri Safarik
Department of Computer Science and Engineering
University of West Bohemia
Univerzitni 8, Plzen
Czech Republic
E-mail: netralova@kiv.zcu.cz

KEYWORDS
Trust, trust modeling, personal trust, social network.

ABSTRACT

The paper deals with the personal trust modeling in social networks. Terms trust, personal trust, trust affecting factors, and trust representation are introduced. The proposed trust model integrates more factors which affect trust determination in social network. The model covers basic factors as reciprocal trust, subject reputation, subject trust recommendations, and is extended by trusting disposition. The role of these factors participating in trust forming is discussed. Modifications of parameter values describing mentioned factors and their effects on personal trust evolution are investigated. Examples of behavior of trust model for social networks are examined by parameter studies. The studies demonstrated that the model enables to study the dynamics of trust evolution in social networks.

INTRODUCTION

The meaning and characteristics of trust were described in many works focused on social issues (Fukuyama 1995; Seligman 1997; Sztompka 1999; Gambetta 2000). In next steps trust models were created (Chang et al. 2005; Lifen 2008). Further, trust plays an important role in e-service (Liu et al. 2008), e-commerce (Zhang et al. 2008), e-banking, peer-to-peer networks (Chen and Yeager 2008). Wide-spreading of communication through social networks, where humans have to collaborate, asked adding trust to these electronic systems (Mui 2002; Carrington et al. 2005; Dwyer et al. 2007; Richters et al. 2011).

Various approaches were designed for computational models of trust. Let us give just a few examples based on agent technology (Rettinger et al. 2007; Sankaranarayanan et al. 2010a), fuzzy systems (Chang et al. 2005; Sankaranarayanan et al. 2010b), Markov models (Hussain et al. 2005), or game theory (Sankaranarayanan 2007). The aim of our work is building trust model and simulation of the trust evolution in social network. The model closely reflects members of social networks, especially introduces differentiating members in their disposition to trust somebody. The organization of the paper is following. Firstly, the term trust and its representation are introduced. Next, we introduce the trust model and we use it for trust affection modeling in social network. Finally, behavior of the trust model is investigated and the results are presented.

TRUST AND TRUST REPRESENTATION

Trust is a fact of everyday life and plays an explicit role in societies. We all make trust decisions, most of us every day in our lives, and many times per day (Luhmann 1979). The decision to trust is based on evidence to believe, or be confident in, someone something’s good intentions towards us (Yamamoto Y., A Morality Based on Trust, 1990). Based on Gambetta (Gambetta 2000), we interpret trust as a confidence in the ability or intention of a person to be of benefit to trustworthy something or someone at sometime in future. Generally, trust can be quantified by a value from an interval (a, b), where a, b (a<b) are integer or real numbers. Verbal trust levels are possible to represent by values from this interval (see in Figure 1). Trust in our model is represented by a value from continuous interval (0, 1). Value 0 represents complete distrust and value 1 means blind trust.

Figure 1: Trust Representation

PERSONAL TRUST REPRESENTATION

Further, we specify an interpersonal trust representation, i.e. trust between two subjects (Netralova and Safarik 2009). Consider a group of n subjects represented as the set \( S = \{ s_1, s_2, ..., s_n \} \). The measure of personal trust between the subject \( s_i \) and \( s_j \) is introduced as follows

\[
t_{ij} = \ell(s_i, s_j), \quad t_{ij} \in (0,1], \quad \text{where } i, j = 1, ..., n, \quad i \neq j,
\]

and

\[
\sum_{j=1, j \neq i}^{n} t_{ij} = 1 \quad (1)
\]

Further we suppose that both values \( t_{ij} \) and \( t_{ji} \) exist, thus providing reciprocal trust. The directed weighted graph is used for personal trust representation in the social network.
Vertices represent the subjects, oriented edges represent trust relations between subjects and the weights are trust values. The direction of the edge reflects possible trust asymmetry, i.e. $t_{ij} \neq t_{ji}$ (trust of $i$-th subject in $j$-th one may differ, and usually differs, from trust of $j$-th subject in $i$-th one).

Example of personal trust representation in a small group is shown in Figure 2. This group consists of three individuals A, B and C. The individual A trusts to B by value 0.9, trust value of individual B to A is 0.4, individual B trusts to C by 0.6, C to B trusts by value 0.8, trust value of individual C to A is 0.2, and trust A to C is 0.1. Note, that the graph does not contain self-looped edges.

![Figure 2: Trust in Group](image)

Interpersonal trust is formed by many factors. Beside usually considered factors, which are reputation and recommendation, we introduce a subject’s trusting disposition. The reputation of the subject comes after individual experience and by some information dissemination about subject in its neighbourhood and influences trust formation considerably. Information about another subject that other subjects have passed on is called recommendation. Trusting disposition represents a non rational behaviour of a subject and is modeled by random factor.

**TRUST IN SOCIAL NETWORK**

Social network can be described as a social structure created by the individuals that are bonded together on the basis of some particularity. This particularity can cover e.g. family relationship, friendship, financial transactions, common interest, and so on.

Applications of social networks are used in biology for disease spread simulation, in economy and marketing to secure higher profit, e.g. advertisements aimed at population groups. Internet and information technologies have the most significant role in social network applications. Social network is not only global network Facebook or Twitter, but also auction portal e-Bay, and its Czech modification Aukro, or specialized servers, e.g. Czech server Heureka focused on discussions about experience and practice in e-commerce.

Social network connects individuals in the groups. Individuals in social network are called actors. Relationships among actors form attitude of one actor to another one. The simplest attitude is binary one – believe or do not believe; in reality the attitude comprises more values. Primary factor in the process of forming attitude is trust.

Generally, social networks are modelled by the oriented weighted graphs similarly to the personal trust model.

Beside the trivial case of two actors, the basic form of social network formed from three actors, called triad, is in Figure 3. Shaded line represents orientation of both trust, and reputation. This convention will be used in Figures in the rest of the paper as well.

![Figure 3: Trust Relations in Triad](image)

In our model, the relationships among the actors $s_i$ ($i = A, B, C$) in triad are represented by reciprocal trust $t_{ij}$ and $t_{ji}$ ($i, j = A, B, C, i \neq j$), reputation $r_i$ and $r_j$, and trusting disposition $g_i$. Sending the recommendation to actor modifies its trust is described in detail in next section. A simple example of recommendation $d^A_{CO}$ which is sent from actor C to actor B about its trust to actor A is shown in Figure 4.

![Figure 4: Trust Relations with Recommendation](image)

Based on the recommendation (dotted line in Figure 4) trust $t_{BA}$ and reputation $r_{BA}$ (dashed line in Figure 4) will be modified.

**SOCIAL NETWORK TRUST MODEL DESCRIPTION**

Trust formation $T_{ij}$ between two actors (trustor and trustee) is generally expressed as function
\[ T_{ij} = f(t_{ij}, t_{ji}, d_{ij}, r_j, g_i), \]  

(2)

where \( t_{ij}, t_{ji} \) is reciprocal trust of actors, \( d_{ij} \) are the recommendations of \( i \)-th actor to \( j \)-th actor about trust to \( k \)-th actor, \( r_j \) is reputation of \( j \)-th actor, and \( g_i \) is trusting disposition of \( i \)-th actor.

Reputation of an actor is computed from its neighbors, which provide their rating of the actor. The rating scale has six degrees - very negative experience VNE, neutral - negative NN, neutral positive NP, positive experience PE, and very positive experience VPE. For each degree, a real value is assigned (see Table 1). The reputation of an actor is arithmetic mean of the actor’s neighbors rating values.

<table>
<thead>
<tr>
<th>Rating</th>
<th>Degree</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VNE</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>NE</td>
<td>1</td>
<td>0.2</td>
</tr>
<tr>
<td>NN</td>
<td>2</td>
<td>0.4</td>
</tr>
<tr>
<td>NP</td>
<td>3</td>
<td>0.6</td>
</tr>
<tr>
<td>PE</td>
<td>4</td>
<td>0.8</td>
</tr>
<tr>
<td>VPE</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 1: Rating, Degree and Value Description of Reputation

Actual trust value \( T_0 \) is produced by two components - previous trust \( t_0 \) and change of trust (gain or loss)

\[ T_0 = t_0 + \Delta T_0. \]  

(3)

Thus, for trust variation of \( i \)-th actor (trustor) to \( j \)-th actor (trustee) the formula (4) is proposed

\[ \Delta T_0 = \sqrt{t_{0ij} d_{ij}^* w_d w_r w_g}, \]  

(4)

where \( t_{0ij} \) is previous trust of \( i \)-th actor in \( j \)-th one, \( t_{ij} \) is previous trust of \( j \)-th actor in \( i \)-th one, and tendency of reciprocal trust influence is reflected by geometric mean; \( d_{ij}^* \) is average of recommendations about \( j \)-th actor to \( i \)-th computed by formula

\[ d_{ij}^* = \frac{\sum r_{ik} t_{ki}}{p}, \]  

(5)

where \( t_{ki}, t_{ki} \) exists, and \( p \) is number of those actors; \( r_j \) is reputation of \( j \)-th subject described by formula

\[ r_j = \frac{\sum r_{ij} t_{ji}}{q}, \]  

(6)

where \( r_{ij} \) is actor’s rating values given by \( q \) actors in its neighborhood; \( g_i \) is trusting disposition expressed by the probability distribution function; influence of recommendation, reputation, and trusting disposition is determined by the weight coefficients \( w_d, w_r, w_g \) from the interval \((0, 1)\).

Then, final trust \( T_0 \) formula is following

\[ T_0 = t_0 + \sqrt{\frac{\sum r_{ij} t_{jk} d_{ij}}{p} w_d (\sum r_{ji} t_{ik} q) w_r g_i w_g}. \]  

(7)

CASE STUDY

To illustrate trust evolution under parameter changes, we took an example of small social network on which the essential cases of trust changes are shown (Havel 2011). Social network consists of fourteen actors with their trust relations. Actors and their known contacts in social network are given by the matrix of actors \( S_0, S_1 \) when the actors \( i, j \) know each other, \( S_0 = 1 \) otherwise.

Trust matrix \( T \) describes reciprocal trust of actors. Existing trust is given by value in the interval \((0, 1)\). Value -1 represents the situation when the actors do not know each other or the fact that reciprocal trust is not known.

\[
\begin{pmatrix}
-1 & 0.2 & 0.99 & -1 & -1 & 0.99 & -1 & 0.99 & -1 & -1 & 0.6 & 0.8 & -1 & -1 \\
0.2 & -1 & 0.8 & -1 & 0.6 & -1 & -1 & -1 & -1 & -1 & -1 & -0.3 & -1 \\
0.99 & 0.99 & -1 & -1 & 0.4 & 0.9 & 0.7 & -1 & -1 & -0.6 & 0.3 & 0.3 & -1 & -1 \\
-1 & -1 & -1 & 0.01 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 0.2 & 0.001 & -1 & -1 & -1 & -0.5 & -1 & -1 & -1 & -1 & -1 & -1 \\
0.99 & 0.99 & 0.9 & -1 & -1 & 0.6 & 0.2 & -1 & -1 & 0.6 & 0.2 & -1 & -1 & 0.7 \\
-1 & -1 & -1 & 0.4 & -1 & -1 & 0.8 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & 0.3 & -1 & -1 & -1 & -1 & 0.3 & -1 & -1 & -1 & -1 \\
-1 & -1 & 0.7 & -1 & -1 & -1 & -1 & -1 & -1 & 0.4 & 0.01 & -1 & -1 \\
-1 & -1 & -1 & -1 & -0.6 & -1 & -0.7 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
0.5 & -1 & -0.3 & -1 & -1 & -1 & -1 & -1 & -1 & 0.8 & -1 & -1 & 0.2 & -1 \\
0.4 & -1 & -1 & -1 & -1 & -1 & 0.1 & 0.01 & -1 & -0.7 & -1 & -1 & -1 & -1 \\
-1 & -0.8 & -1 & -1 & -1 & 0.8 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & 0.4 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
\end{pmatrix}
\]

Actor rating degrees are placed in the rating matrix \( R \).

\[
\begin{pmatrix}
-1 & 5 & 5 & -1 & -1 & 5 & -1 & -1 & 5 & 1 & 5 & 5 & -1 & -1 \\
5 & -1 & 5 & 5 & -1 & -1 & 5 & -1 & -1 & 5 & 1 & 5 & 5 & -1 \\
4 & 3 & 1 & -1 & -1 & 5 & 3 & 1 & -1 & 5 & 2 & 4 & -1 & -1 \\
-1 & -1 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
5 & 5 & 5 & -1 & -1 & 1 & 4 & 4 & -1 & -1 & 1 & 3 & -1 & -1 \\
-1 & -1 & 3 & -1 & -1 & 2 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -4 \\
4 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & 0 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 5 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
4 & -1 & -1 & -1 & -1 & -1 & 3 & 5 & -1 & 0 & -1 & -1 & -1 & -1 \\
-1 & 3 & -1 & -1 & -1 & 5 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 1 & -1 & -1 & -1 & 3 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
\end{pmatrix}
\]
Trusting dispositions of actors are stored in trusting disposition vector $G$.

$G = [0.99, 0.99, 0.8, 0.01, 0.01, 0.7, 0.4, 0.6, 0.3, 0.5, 0.8, 0.99, 0.01, 0.4]$

The greater is value of trusting disposition parameter, the higher is tendency to trust anticipation.

The weights - recommendation weight $w_{rk}$, reputation weight $w_{r}$, and trusting disposition weight $w_{d}$ were set to the value 0.6 in this case.

**Reciprocal trust growth**

Let us start with a pair of actors with rather low reciprocal trust. On the other hand they have high reputation and high trusting disposition. One of them has high recommendations, while the other one’s recommendation is average. Situation is shown in Figure 5. Four actors, i.e. 1, 2, 3 and 6, create the core of chosen part of social network. Pair of actors under study is 1, 2. We expect subsequent growth of both trust $T_{12}$, $T_{21}$, while the first should grow more rapidly. The results from our model depicted in Figure 6 are in good concordance with the expectation.

Figure 5: Trust Relations for Computation of Reciprocal Trust $T_{12}$, $T_{21}$

Figure 6: Study of Reciprocal Trust Growth $T_{12}$, $T_{21}$

**Recommendation influence**

We explored how the trust under reputation growth will increase. The trust into an actor is computed for all six degrees of actor ratings by its neighbors. The study covers actors 1, 3, 9, and 11 (see Figure 7).

Figure 7: Reputation Influence – Needful Trust Relations

Reputation values of actor 11 gradually increase from 0 to 5 (rating matrix $R$, row 11). The other parameters do not change. Trust growth between actor 3 and actor 11 is depicted in Figure 8 and is in good in concordance with expected behavior.

Figure 8: Study of Reputation Influence on Trust $T_{3,11}$

In this study the influence of recommendations on trust evolution is explored. Recommendation can be sent to direct neighbor actor only about direct neighbor of sender actor. To demonstrate the influence we selected the subgraph with actors 1, 6, and 8 (see Figure 9).

Figure 9: Recommendation Influence – Needful Trust Relations
Studied trust is between actor 6 and actor 8. The growth of this trust is brokered by common friend - actor 1 that will send the recommendation about actor 8 to actor 6. Influence of recommendation depends on trusts of actor 6 into actor 1, and actor 1 into actor 8. The value of element of trust matrix $T [6,1]$ is gradually changed to 0.1, 0.3, 0.5, 0.8, and 0.99, in association with element of trust matrix $T [1,8]$ which value is changed to 0.2, 0.3, 0.6, 0.8, and 0.99 pair wise. The results of this study are shown in Figure 10. According to expectation trust increases more rapidly with higher recommendations, cause by increase of trust between broker-actor and recommended actor as well between recommending actor and broker.

![Figure 10: Study of Recommendation Influence on Trust $T_{68}$](image)

**CONCLUSION**

We developed the personal trust representation, where trust values are from continuous interval compared e.g. to Markov models using trust levels. Based on this representation, trust model for social networks was developed and implemented. Model enables to study the dynamics of trust evolution in a social network under changing trust forming factors. Upcoming model modification will allow covering the effect of intentional trust affection of social network members.

**REFERENCES**


**AUTHOR BIOGRAPHIES**

**ARNOSTKA NETRVALOVA** was born in Plzen, Czech Republic. She is senior lecturer in Department of Computer Science and Engineering at Faculty of Applied Sciences of University of West Bohemia. She holds M.Sc. in Computer Science from University of West Bohemia in 1977. Her research interest in modeling and simulation in medicine
covered models of temperature homeostasis. She received Ph.D. from the same university in 2010. Her present research is focused on trust modeling and simulation.

JIRI SAFARIK was born in Kromeriz, Czech Republic. He received his Ph.D. degree from Slovak University of Technology in 1984. Currently, he is professor in Department of Computer Science and Engineering at Faculty of Applied Sciences of University of West Bohemia. His research covers distributed systems, distributed and parallel simulation.
COMPETENCY-BASED HUMAN RESOURCES ALLOCATION IN BUSINESS PROCESS SIMULATIONS

Štěpán Kuchař, Jan Martinovič and Tomáš Kocyan
Department of Computer Science
VŠB – Technical University of Ostrava
17. listopadu 15/2172
70833 Ostrava-Poruba, Czech Republic
E-mail: stepan.kuchar@vsb.cz, jan.martinovic@vsb.cz, tomas.kocyan@vsb.cz

KEYWORDS
Human Resource Allocation, Business Process Simulation, Competency Management, Vector Space Model

ABSTRACT
Many business process simulation models treat all human resources of one role as interchangeable but human resources in real processes are unique and their skills and competencies affect their productivity. It is therefore better to consider the varying levels of competencies in allocation of the human resources in automatic process simulations. This paper describes how to encode the resources' competencies into the vector space model and use it to find the fitting resources for performing process activities.

INTRODUCTION
A large number of business processes are based on the activities performed by human resources (HR) and skilled and properly trained human resources are one of the main sources of the company's competitive advantage (Chang and Huang 2005; Hatch and Dyer 2004). But they have to be managed well so that their potential can be fully utilized. This is done by managers. Good manager knows her workers, she can lead, motivate and inspire them, she knows which trainings could enhance their productivity and she tries to optimally plan the work and allocate right workers to the right activities. But in automatic simulations, managers are not available to plan these allocations, because the simulation is fully performed by the machine. This article introduces a solution to the problem of evaluating available human resources to the current activity in the ongoing process based on the resource's competencies. This evaluation will enhance the simulation by the means of finding the appropriately skilled resource that can be allocated to the activity based on the actual availability of evaluated resources. It can be also used to support the decisions for training or hiring new resources based on the requirements of the process activities and predicted workload of the process.

RELATED WORK
Existing business process simulation models are not very concerned with accurate human resources modeling and description (Rozinat et al. 2009; Aalst et al. 2008). Many simulation models suppose that all resources in one role have the same skills and that it does not matter which one of these resources will be allocated to the activity (Pesic and Aalst 2005; Russell et al. 2005). But clearly each human resource in the process is unique with his own set of skills and experiences, each one has specific working habits and performance (André et al. 2010). In our paper we use the competency-based approach to differentiate individual resources in the process and to correctly allocate resources to the process activities during business process simulations.

Our evaluation approach is inspired by the recommender systems used for human resources selection in e-recruitment. The goal of these systems is to find the right applicant for given job offer and are used in personal agencies and HR departments to support the recruitment decisions of HR consultants. Research concerning these systems is however focused mainly on mining required data about applicants and job offers from their textual representation using ontologies (Crow and DeSanto 2004; Radevski and Trichet 2006), learning models (Aiolfi 2009) or n-grams (Kessler et al. 2009) and only a few are concerned with the search and evaluation methods (Malinowski et al. 2006; Kessler et al. 2008). We localize this problem to the simulation of internal processes and search the appropriate human resource for each subsequent activity in the process. Our evaluation method presented in this paper also considers required levels of skills used by the process activities and determines how much better or worse the resource is. This will be used in our future research to dynamically change the performance of human resources throughout the whole process simulation.

COMPETENCY-BASED DESCRIPTION OF HUMAN RESOURCES AND ACTIVITIES
The first problem that has to be solved is the description of the human resources' skills in the process. For this the competency models (Dreyfus and Dreyfus 1980; Sinnott et al. 2002; Ennis 2008) and skills frameworks (e.g. Skills Framework for the Information Age (SFIA Foundation 2010), NHS Knowledge and Skills Framework (UK Department of Health 2004)) are widely used. Competency models describe various competencies which are important for the company and its processes. Competencies are defined as sets of knowledge, abilities, skills and behavior that contribute to successful job performance and the achievement of organizational results (Sinnott et al. 2002).
Skills frameworks have the same purpose, but they describe skills particular for one domain rather then general competencies. But in fact skills are just a special type of competencies.

Competency models and skills frameworks also describe how to measure and evaluate individual competencies. In most cases competencies are measured by a number of advancing stages where higher levels of competency include everything from their lower levels. The first such evaluated competency model was envisioned by Stuart and Hubert Dreyfus in 1980 in their paper about directed skill acquisition of students (see Dreyfus and Dreyfus 1980). Their model had five levels: Novice, Competence, Proficiency, Expertise and Mastery. Later models used the same system, but they did not keep the number of levels. There is no standard for how many levels should a competency model have and every model defines his own set of levels.

Let's have a small example of one HR consultant working in a personal agency. His competencies in a 10-level model could look like this:

- Personnel knowledge in IT – 5. level,
- Personnel knowledge in Management – 3. level,
- Internal IS user skill – 6. level,
- Communication – 5. level,
- Customer knowledge of VSB-TUO – 4. level,
- Customer knowledge of EUROSI – 0. level.

Domain specific skills (personnel knowledge, internal IS user skill), general competencies (communication, psychology) and knowledge of the environment (customer knowledge) are contained in this example. It is clear that competencies in the model have to be based on the company's requirements and professional domain.

All activities in the process also have competency-based requirements that describe what competencies should the human resources performing this activity know. Each activity will therefore be defined by the set of competency levels for each resource type performing the activity specifying that only resources with given level or higher will do the activity as planned. Resources with lower competencies are able to finish the activity, but they have to spend additional time with learning how to perform the activity and their work is prone to contain more errors.

A simple example of requirements for the activity of leading an interview with an applicant in the personal agency follows:

- Personnel knowledge – 5. level,
- Internal IS user skill – 0. level,
- Communication – 6. level,
- Customer knowledge – 4. level.

If we compare this example with the resource example from previous chapter, one can notice the generalization of some requirements (personnel knowledge and customer knowledge). When assessing the resource's competencies it is better to define the competency levels in specific parts of the domain so that the resources are assessed as precisely as possible. On the other hand the activity requirements should only define a level for the whole competency category and relevant part of the domain will be specified by actual process case. In other words, if the personal agency tackles with a case where they have to find a programmer for the company VSB-TUO, then the requirements in this case will be refined as personnel knowledge in IT and customer knowledge of VSB-TUO.

### COMPETENCY VECTOR REPRESENTATION

To evaluate human resources based on the process activity requirements we utilized the vector space model that is very often used in document searches (see Berry 2003). One of this model’s advantages is the option of ranking vectors according to their decreasing similarities to the query vector (see Salton 1988). A common similarity measure used in the vector space model is the cosine measure (see Lee 1997).

To use this model we just needed to find a way to describe the resource competencies and activity requirements as vectors. This could be solved directly by creating a simple vector containing the competency levels for given resource (e.g. a vector \((5,3,6,5,4,0)\) defining the resource competencies from the example in the previous chapter) but this representation behaved very poorly in our experiments. Resources with low competencies were highlighted in this representation thanks to the inherent properties of the cosine measure.

Much better results were achieved by fragmenting the competencies to a number of vector elements equal to the highest possible level of the competency. Each level of any competency therefore corresponds to one vector element that contains number 1 if the resource has mastered this level or number 0 if it has not. The bordering element is able to hold a real value to enable more accurate evaluation of partially mastered competency levels. This enables

Every competency \(c_1, ..., c_n\) is therefore split into fragmented competencies \(f_{c_1,1}, f_{c_1,2}, ..., f_{c_1,maxlv}(j)\) where \(maxlv(j)\) is the highest possible level of the competency \(c_j\).

Vector representation for the fragmented competencies \(f_{c_j,k}\) and resource \(r_j\) is:

\[
\mathbf{f}_{j,k} = (f_{i_1,k_1,1}, f_{i_1,k_1,2}, ..., f_{i_1,k_1,maxlv(j)})
\]

where the values \(f_{i_1,k_1}\) are determined as:

\[
\begin{align*}
    f_{i_1,k_1} &= 1 & k \leq i_1,
    f_{i_1,k_1} &= i_1 - k + 1 & i_1 < k < i_1 + 1
  \end{align*}
\]

where \(i_1\) is the actual competency level of the competency \(c_j\) mastered by the resource \(r_j\).

All mastered competency levels of the resource \(r_j\) can then be described by the vector:

\[
\mathbf{r}_j = (f_{i_1,1,1}, ..., f_{i_1,1,maxlv(1)}, ..., f_{i_n,1,1}, ..., f_{i_n,1,maxlv(n)})
\]

To create such vectors for all resources in the process, it is crucial that all resources have levels assigned to all competencies \(c_1, ..., c_n\). If this isn't the case, then the competencies have to be unified. Missing competencies are set to level 0 and some competencies can be merged with their level defined by their weighted average.

Vector representation of the activities follows the same pattern but with its own specifics. In contrary to the
resource competencies the activity requirements describe the lowest required level of the competency for given activity. But to reduce the actual value of over skilled resources, it is also useful to limit the highest required level. This leaves the more skilled resources for more difficult tasks. The fragmented competency levels can then be set to 1 in between the lower and upper limits and 0 otherwise. The bordering values are also able to hold real values to describe partial requirements.

The vector representing the competency requirements of activity \( a \) for competencies \( r_{c_1}, r_{c_2}, \ldots, r_{c_n} \) is:
\[
a = (f_{rl_{a,1}}, \ldots, f_{rl_{a,1}, maxvl(j)}), \ldots, f_{rl_{a,n}}, \ldots, f_{rl_{a,m}, maxvl(s)})
\]
where \( f_{rl_{a,j}} \) is defined as:
\[
f_{rl_{a,j},k} = \begin{cases} 
0 & , k \leq lrl_{a,j}-1 \\
k+1-lrl_{a,j} & , lrl_{a,j}-1 < k < lrl_{a,j} \\
1 & , lrl_{a,j} \leq k \leq hrl_{a,j} \\
lrl_{a,j}+1-k & , hrl_{a,j} < k < hrl_{a,j}+1 \\
0 & , k \geq hrl_{a,j}+1
\end{cases}
\]

where \( lrl_{a,j} \) is the lower limit level of the required competency \( r_{c_j} \) for activity \( a \), and \( hrl_{a,j} \) is the upper limit. In order to lessen the influence of resource competencies that don’t appear as requirements in activity \( a \) we define:
\[
f_{rl_{a,j},k}=0, lrl_{a,j} = 0 \land hrl_{a,j} = \text{maxvl}(j)
\]
for every \( k = 1, \ldots, s \).

To successfully use this representation in the vector space model all required competencies have to be defined for all activities in the process and they have to be compatible with the resource competencies. The following conditions have to be met to ensure this compatibility:

1. Resource and activity competencies are the same and are in the same order:
\[
(\forall a \in A) [n = m \land (\forall j \in \{1,2,\ldots,n\}) [c_j = r_{c_j}(a)]
\]
where \( A \) is a set of all observed activities in the process, \( n \) is the number of resource competencies and \( m \) is the number of activity competencies.

2. Resource and activity competencies share the same scale for their levels:
\[
(\forall a \in A) [(\forall j \in \{1,2,\ldots,p\}) (\forall j \in \{1,2,\ldots,n\}) \left( l_{c_j}(j) = \text{minvl}(j), \text{maxvl}(j) = \text{maxvl}(j) \right) \lor \left( r_{rl_{a,j}}(j) = \text{minvl}(j), \text{maxvl}(j) = \text{maxvl}(j) \right)]
\]
where \( A \) is a set of all observed activities in the process, \( p \) is the number of resources in the process, \( n \) is the number of resource competencies, \( \text{minvl}(j) \) is the lowest possible level of the competency \( c_j \) and \( \text{maxvl}(j) \) is the highest possible level of \( c_j \).

3. All levels of corresponding competencies \( c_j \) and \( r_{c_j}(a) \) have to share the same meaning for all activities in the process and all \( j = 1, \ldots, n \).

**EXAMPLE OF USE**

Let’s look at the simple process example in Figure 1 to show how the evaluation of resources could enhance the simulation of the process. Let there be two sequential activities in the process called Find Applicant (FA) and Contact Applicant (CA), the former being performed by an Administrative Worker and the latter by a Personal Consultant. Let’s assume that there are 3 Administrative Workers (AW) and 2 Personal Consultants (PC) in the process and their mastered competency levels are summarized in the Table 1 along with the required competencies defined for the two activities (competencies are defined on a 5-level scale).

**Figure 1: Activity Diagram of the Process Example**

**Table 1: Resource and Activity Competency Levels**

<table>
<thead>
<tr>
<th>Competency</th>
<th>AW1</th>
<th>AW2</th>
<th>AW3</th>
<th>PC1</th>
<th>PC2</th>
<th>FA</th>
<th>CA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personnel knowledge - IT</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>2-3</td>
<td>3</td>
</tr>
<tr>
<td>Internal IS skill</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Communication</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Customer knowledge- VSB</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

The simulation of such process first starts the activity Find Applicant but needs one of the Administrative Workers to perform this activity. But which one of the three should the simulation choose in this case? This problem is solved by converting all three resources and the FA activity to the competency vector representation and using the vector space model to evaluate them. Based on the rules defined in previous chapters the resource AW1 can be described as:
\[
AW1=(1,1,0,0,0, 1,0,0,0,0, 1,1,1,1,0, 1,1,1,0,0)
\]
and the description of remaining workers is very similar. On the other hand activity FA can be represented as:
\[
FA=(0,1,1,0,0, 0,0,1,1,1, 0,0,0,0,0, 0,0,0,0,0)
\]

Now it is just a matter of using the formula for cosine measure to compute the similarity of individual resource’s vectors with vector FA. The similarities are 13.5% for WA1, 44.7% for WA2 and 33.8% for WA3. WA2 is apparently the best resource for this activity, but it is also interesting to discover how much better is the resource when compared with the basic activity requirements. This can be easily done by creating a referential resource for the activity that will have exactly the same competencies as the activity requirements (i.e. \( l_{c_j} = lrl_{a,j} \)). The referential resource for activity FA is therefore described as:
\[
r_{FA}=(1,1,0,0,0, 1,1,1,0,0, 0,0,0,0,0, 0,0,0,0,0)
\]
and its similarity with vector FA is 40%. This result shows that WA2 is only slightly better that was required by the activity and also WA3 is only slightly worse. Both these resources can be used to perform this activity and their allocation can be decided by the priority of the current process case so that better resources are allocated to higher priority process cases. WA1 on the other hand is significantly weaker and should perform the activity only when there is no other choice.
After the activity FA is completed the simulation continues with starting the activity CA. Here the similarities are 37% for PC1, 40% for PC2 and 33.3% for the referential resource. Both consultants are able to perform this activity without problems and there is only a slight difference between them.

CONCLUSION AND FUTURE WORK

Evaluation method proposed in this paper can effectively solve the problem of allocating human resources during the automatic process simulations based on their competencies.

This method serves only as a basic solution that opens new possibilities for our future research. Only one process without parallel activities was presented in the example so we did not have to solve the worker utilization. Real processes can contain parallel activities and most importantly many different process cases can run at the same time so the availability of the resources has to be taken into consideration. Thisopens a question if the process should wait for the best resource when it is unavailable or if less suitable available resource should be allocated. We also plan to implement this method to the simulation tool of the BPM Method (Kuchaf et Kožuszník 2010) and extend it with performance evaluation based on the comparison with the referential worker because more experienced resources (with higher competency levels) have better performance (Hatch and Dyer 2004). Another problem of this method is the fact that high competency levels have the same weight as the lower levels. But the growth of the resource’s capabilities is not linear and grows slowly at higher levels (Hanne and Neu 2004). This can be solved by introducing lower or even negative weights to the high activity requirement levels.

ACKNOWLEDGEMENTS

This research has been supported by the internal grant agency of VSB-TU of Ostrava - SP2011/56 Knowledge approach to the modeling, simulation and visualization of software processes and elaborated in the framework of the IT4Innovations Centre of Excellence project, reg. no. CZ.1.05/1.1.00/02.0070 supported by Operational Programme ‘Research and Development for Innovations’ funded by Structural Funds of the European Union and state budget of the Czech Republic.

REFERENCES


Los Alamitos, CA, USA: IEEE Computer Society.


Market Segmentation through Conjoint Analysis using Latent Class Models

Liberato Camilleri, Lara Marie Azzopardi
Department of Statistics and Operations Research
University of Malta
Msida (MSD 06) Malta
E-mail: liberato.camilleri@um.edu.mt

KEYWORDS

ABSTRACT
Conjoint Analysis is accepted by market researchers as a reliable and suitable instrument for measuring consumer preferences. The popularity of conjoint analysis hinges on the belief that it produces valid measurements of consumer preferences for the features of a product or service. It is the marketers’ methodology for assessing the impact of proposed actions on the market and finding out how buyers trade-off among competing products and suppliers. A popular application of conjoint analysis is market segmentation which addresses the heterogeneity in consumer preferences. Market segmentation assumes that a heterogeneous population is represented as a collection of homogeneous subgroups where customers in each cluster have similar needs and similar views of how to value a product. Other applications of conjoint analysis include pricing strategies, product positioning, competitive analysis, promotional policies, new product identification and distribution decisions. This paper describes the issues in implementing conjoint analysis and then illustrates the methodology to perform market segmentation using latent class analysis. The application focuses on customer preferences when evaluating the worth of mobile phones given demographic and product-related predictors.

1. IMPLEMENTING CONJOINT ANALYSIS
Conjoint analysis involves a framework of distinct steps, which include the selection of the utility (preference) function; selection of a method, design and procedure for data collection; the selection of a measurement scale for the response variable and the selection of an estimation method.

The utility function relates the benefit of a product profile to defined attributes (predictors). These attributes could either be discrete or continuous. There are basically three types of utility functions. The vector model assumes a linear relationship between the utility of a product and an attribute having a metric scale. The ideal point model, very often a quadratic function, assumes the existence of an ideal manifestation. The utility reaches a maximum value at one attribute value. The part-worth model relates the utility of a product to a categorical attribute. A parameter is estimated for each attribute category.

The four types of data collection methods for conjoint analysis include the self-explicated technique (Srinivasan and Wyner 1989); the full profile approach (Green and Rao 1971); the two-factor method (Johnson 1974) and the hybrid technique. Self-explicated techniques assess the subjects’ utilities directly. The respondents are first asked to worth the levels of each attribute separately by rating them on a discrete preference scale and then asked to rate the importance of each attribute, perhaps using a different preference scale. Part-worths are computed by multiplying the importance weights with the attribute-level desirability ratings. In a full profile approach a respondent has to worth a complete set of profiles (stimulus cards) describing a product where each profile incorporates one level from each of the attributes of interest. The flexibility in scaling makes the full profile approach more attractive than other approaches. The main argument that favours the full profile approach is that it comes closer to a real buying situation in which the respondents react to a set of total profile descriptions, which are realistic representations of real items. In a two-factor approach, items are assessed through two-way attribute tables (trade-off matrices) in which the rows and the columns represent the levels of the two selected attributes. Each respondent has to rank all combinations of the levels of the two attributes in the associated matrix elements. The number of two-way tables, that has to be ranked, increases with the number of attributes used. An advantage of this approach is that it reduces information overload on respondents because all the attributes are evaluated two-at-time. The limitations of this approach are the exhaust removal of two-way tables that need to be filled out and the lack of realism in decomposing the set of attributes to two-at-time combinations. A hybrid approach combines the self-explicated task with aspects of the full profile conjoint analysis. The first part of the interview uses a self-explicated approach in which a respondent is asked to give a direct judgment of each attribute and its levels prior to the presentation of the profiles. The self-explicated context puts emphasis on evaluating products feature by feature rather judging the product as a whole. HCA (Hybrid conjoint analysis) uses full profiles in the second stage; whereas ACA (Adaptive conjoint analysis) uses partial profiles, composed of only a subset (usually two or three) of attributes, in paired-comparisons and which is viewed as a modern form of the two-factor method. The primary advantage of these hybrid techniques over other methods is that they allow a greater number of attributes to be managed by evaluating smaller numbers of profiles. The reduction in the number of profiles judged is compensated by the information collected from the self-explicated interview.
The three types of data collection designs include the complete factorial, the bridging and the fractional factorial designs. If a conjoint application is confined to a limited number of attributes with a limited number of levels then the full profile approach can be implemented by a complete factorial design. Such a design includes all possible combinations of the levels of the attributes in the study. This approach offers no problem with orthogonality (independence of the attributes) and all main effects and their interactions are estimable. The major limitation of this approach is that most applications include several attributes with varying number of levels. Implementing a complete factorial design would create a large number of incentives (item profiles) and will result in information overload on the respondents. The bridging and fractional factorial designs resolve this problem by reducing the number of incentives. In a bridging design the whole set of attributes is split into subsets and each card deck is composed of attribute level combinations from any subset of the attributes. To link part-worth functions across the various subsets of attributes one or two attributes will be common across all card decks. In a fractional factorial design the design is reduced systematically in such a way that the attributes are orthogonal as much as possible. In some commercial applications, the attributes are correlated and so an orthogonal design can produce stimuli that are not realistic. Other orthogonal displays can be tried by permuting sets of attribute levels if some of the stimulus profiles turn out to be non-representative.

Verbal, paragraph and pictorial descriptions are basically the three ways of presenting the incentives. In a verbal presentation the incentives are presented on information sheets using either key words or descriptive sentences or a combination of both. The paragraph description approach provides a more realistic and complete description of the stimuli and is used when comparing and testing different advertising claims. A drawback of this procedure is the information overload on respondents by having to read large quantities of information. Reducing the total number of descriptions may produce very inaccurate parameter estimates at the individual level. Another limitation is that verbal and paragraph descriptions are subject to response biases resulting from the order in which attributes are presented. The importance of an attribute is to some extent affected by the position of the attribute in the stimulus card. Visual presentations can either be graphic, where drawings or photographs are used or physical, where real products and prototypes are used. The use of profile cards and other pictorial material causes less fatigue to the respondents by providing an easier way to get information and hence allow a greater number of attributes to be included in the study. Another advantage is that the visual stimuli are more realistic because in the marketplace consumers choose their products by inspecting them. Such an inspection is more closely approximated by pictorial presentations. The use of film clips and full-scale prototypes is essential to give respondents maximum exposure to the stimulus especially when the task involves a radical new product idea. The primary disadvantage is that visual displays may exhibit additional information, such as style and colour of the item, that the researcher has no intention to analyze.

Three data collection procedures include person-to-person interviews or use mail and online questionnaires. Using person to person interviews is a rather slow process and very time consuming. The use of mail questionnaires ensures geographic representativeness but may suffer from lack of response. Phone-mail-phone procedures are used by several researchers to ensure a high completion rate with negligible missing data problems and simultaneously reduce the selection bias of respondents. A relatively new method for data collection is the online questionnaire in which the respondents receive the questionnaire and send their reply via e-mails.

The response modes used to evaluate incentives or stimuli can be divided into metric, non-metric and choice based. Ranking and paired profile comparisons are non-metric procedures. In rank data the outcome is just an order of preferences. It may express the preference-worthiness of a profile but does not result in metric ordinal preference data. In a paired-profile comparison the respondent has to declare his preference between two incentives. One of the shortcomings of rank-based data is the distortion caused by the interference between less and highly important variables. Rating, constant sum comparisons and dollar metrics are metric procedures. In rating data respondents grade the profiles subjectively on an interval scale, assuming that they perceive scale spacing. The outcome expresses the intensity of the preferences. In a constant sum comparison respondents are asked to allocate a fixed number of points across a number of profiles. This method provides importance weights that depend on the perceived importance of each profile. Another way of obtaining interval-scale judgments is the dollar metric approach. In this graded paired comparison a respondent has to compare two items and has to state the price that must be added to the least preferred item to make it equally worth to the other. The results are then aggregated to obtain an interval, scaled dollar metric of comparisons. Limitations to this approach are that respondents may have biased perceptions with regards to the use of price differences as a response measure and is a slow procedure compared to the rating method. Choice-based conjoint analysis relies on data from a discrete choice experiment in which each product is a hypothetical combination of attributes chosen by an experimental design procedure. The respondents are presented with profile descriptions of two or more competing items that vary on one or more attributes and their task is to choose the most preferred item. The major advantage of a choice-based task is that it has greater external validity because it mimics what consumers actually do in the marketplace. Moreover, it is a simpler task for respondents to choose incentives rather than rate or rank these alternatives. The major limitation of choice-based analysis is that it contains minimal information about consumer preferences. A choice simply indicates which profile is most preferred but it does not provide an estimate of the utility of the product profiles.

Modern statistical analysis is based on the likelihood principle that all the information in the observed data is contained in the likelihood. The likelihood can be defined as the probability of the observed responses expressed as a function of the unknown parameters. Hence a likelihood
method can use the data optimally. Maximum likelihood and Bayesian analysis are the two main areas that use likelihood methods. Hierarchical Bayes methods derive part worths by combining information on the distribution across respondents. The posterior distribution of individual parameters is estimated using a computationally intensive method called Gibbs sampling that produces estimates of each respondent’s part worths and standard errors. Hierarchical Bayesian analysis provides very flexible output and the researcher may choose among many possible population distributions; however the method requires considerable expertise to execute properly. During the last four decades researchers have used these estimation techniques to estimate parameters of models for different types of conjoint data.

2. MARKET SEGMENTATION

Traditionally, market segmentation in conjoint analysis was carried out using either a-priori or post hoc procedure. In a-priori segmentation analysis the number of segments is determined in advance by the researcher and individual-level preference judgments are combined at the segment level. Actually, this is not appropriate since demographic and psychographic predictors rarely describe adequately the heterogeneous utility functions. In post-hoc or tandem segmentation, estimation and clustering are carried out consecutively. Individual-level parameter estimates are first obtained from normal regression models and then individuals are clustered on the basis of similarity of the estimated parameters by using Ward’s hierarchical or K-means non-hierarchical clustering procedures. This two-stage approach also has problems since different clustering methods will produce different outcomes. Moreover, the initial utility estimation method using regression analysis and the subsequent cluster analysis optimize different and unrelated objective functions.

To address the limitation of a-priori and post-hoc methods, several integrated conjoint segmentation methods were proposed in which the parameters within the segments are estimated at the same time that the segments are identified. Thus a single criterion of interest is optimized under a set of constraints. (Hagerty 1985) proposed a method using a weighting scheme representing a factor-type partitioning of the sample. The scheme optimizes the expected mean squared error of prediction in validation samples. (Ogawa 1987) proposed a non-overlapping hierarchical clusterwise regression procedure that allows for concurrent estimation and segmentation using logit estimation. (Kamakura 1988) proposed a similar methodology for conjoint models using least squares estimation. (Wedel and Kistemaker 1989) proposed a generalization of the clusterwise regression to handle more than one observation per individual and which yields nonoverlapping, nonhierarchical segments. (DeSarbo, Oliver and Rangaswamy 1989) proposed an overlapping nonhierarchical clusterwise regression method that uses a simulated annealing algorithm for optimization. (Wedel and Steenkamp 1989, 1991) proposed a fuzzy nonhierarchical clusterwise regression algorithm that permits subjects to have partial membership in at least one segment.

Probably, the advent of latent class and finite mixture models stands out to be the most far-reaching development in market segmentation. The merit of these models is that they allow for simultaneous segmentation, estimation and enable statistical inference. Work on latent class models was initiated by (Quandt 1972) who introduced the concept of switching regression models. (Goldfield and Quandt 1973, 1976) proposed a hidden Markov switching regression approach in which membership of observations within a cluster is modelled by a Markov process. (Engel and Hamilton 1990) extended the switching regression method to time series. The models describe discrete shifts in autoregressive parameters, where the shifts themselves are modelled by a hidden discrete-time Markov process. (DeSarbo, Wedel, Vriens, Rangaswamy 1992) and (Wedel and DeSarbo 1995) proposed a multivariate normal latent class model using the EM algorithm which calculates the posterior probabilities in the E-step. In an excellent review, (Vriens, Wedel and Wilms 1996) conducted a Monte Carlo comparison of several traditional and integrated conjoint segmentation methods. The authors found that Latent Class segmentation models performed best in terms of parameter recovery, segment membership recovery and predictive accuracy.

3. A LATENT CLASS MODEL

The latent class model described below was proposed by (DeSarbo, Wedel, Vriens, Rangaswamy 1992) in which market segments and part-worth utilities are estimated simultaneously using mixtures of multivariate conditional normal distributions. Parameters of these mixtures are estimated using the EM algorithm. Given that the \( n \)th respondent belongs to the \( k \)th segment, the conditional multivariate density of the dependent vectors \( y_s = (y_{s1}, \ldots, y_{sk}) \) for \( j = 1, \ldots, J \) replications is:

\[
 f_s(y_s; \beta_k) = (2\pi)^{S/2} |\Sigma_k|^{1/2} \exp \left[ -\frac{1}{2} (y_s - \Theta_k)^T \Sigma_k^{-1} (y_s - \Theta_k) \right]
\]

where \( \Sigma_k \) is the variance-covariance matrix of \( y_s \) given segment \( k \). The unconditional density function is:

\[
 f_s(y_s; \pi, \beta) = \sum_k \pi_k f_s(y_s; \beta_k)
\]

Using a likelihood approach, the log likelihood function can be formulated as follows:

\[
 \ln L(\pi, \beta) = \ln \prod_{n=1}^{N} f_s(y_s; \pi, \beta) = \sum_{n=1}^{N} \ln \sum_{k=1}^{K} \pi_k f_s(y_s; \beta_k)
\]

Maximizing the expected log-likelihood function is not an easy task. An effective procedure that fits a latent class model with \( K \) segments is to maximize the expected complete log-likelihood function using the EM algorithm. The idea behind the EM algorithm is to augment the observed data by introducing unobserved 0-1 indicators \( \lambda_{nk} \), indicating whether the \( n \)th respondent belongs to the
Given the matrix \( \mathbf{A} = (\lambda_{ab}) \) of unobserved data, the complete log-likelihood function is:

\[
\ln L(\mathbf{r}, \mathbf{p} | \mathbf{A}) = \sum_{a=1}^{K} \sum_{b=1}^{B} \lambda_{ab} \ln f(y_{ab} | \mathbf{p}_{a}) + \sum_{a=1}^{K} \sum_{b=1}^{B} \lambda_{ab} \ln (\pi_{a})
\]

In \( L(\mathbf{r}, \mathbf{p} | \mathbf{A}) \) has a simpler form than \( \ln L(\mathbf{r}, \mathbf{p}) \) and is easy to maximize. Once the parameter \( \mathbf{p}_{a} \) and \( \pi_{a} \) are estimated, the posterior probability \( \hat{p}_{ab} = E(\lambda_{ab}) \) can be calculated using Bayes’ theorem.

\[
\hat{p}_{ab} = E(\lambda_{ab}) = \frac{\hat{\pi}_{a} \cdot f(y_{ab} | \hat{\mathbf{p}}_{a})}{\sum_{b=1}^{B} \hat{\pi}_{a} \cdot f(y_{ab} | \hat{\mathbf{p}}_{a})} \quad \text{where} \quad \sum_{a=1}^{K} \hat{p}_{ab} = 1
\]

The iterative procedure is initiated by first setting pseudo random real values to \( \hat{p}_{ab} \) in the range [0-1]. The EM algorithm updates alternately the parameters \( \hat{\mathbf{p}}_{a}, \hat{\pi}_{a} \) and the posterior probabilities \( \hat{p}_{ab} \) until it converges. Subjects are then assigned to the segment with highest posterior probability \( \hat{p}_{ab} \).

The Bayesian information criterion (BIC) will be used in this latent class model to identify the number of segments.

\[
BIC = -2 \log L + d \log N
\]

\( d \) is the number of estimated parameters and \( N \) is the number of respondents.

4. APPLICATION

The main objective of this study is to establish which factors influence consumers’ choices when buying mobile phones; which characteristics of the mobile phones are identified as most important by consumers in the marketplace. What feature of the product effectively improves market sales? Do consumers give more priority to price or to brand? These are some of the questions that will be addressed in this paper. The four selected mobile phone attributes included brand (A and B), price (€150, €175 and €200), whether the mobile phone has internet access and touch screen facility. By choosing a complete factorial design, twenty-four profiles of different mobile phones were generated using a full profile approach. The stimuli were described using a verbal approach by providing details about each attribute. The questionnaire was sent to a 778 university students using an online survey. The respondents had to rate each profile using a 7-point Likert scale, where 1 corresponds to an unworthy mobile phone and 7 corresponds to a very worthy one. The participants were also asked to specify their gender, age and number of mobile phones they owned.

The latent class model included all four item-attributes and three individual covariates. Since some of the predictors are categorical and others are continuous, a mixed model was assumed since it allows some attributes to follow the part-worth model while others follow the vector model. To identify the optimal number of segments, the latent class model was fitted several times each time changing the number of segments from 1 to 3. For each solution the BIC criterion was computed. Table 1 displays that the two-segment solution is the one which minimizes the criterion.

<table>
<thead>
<tr>
<th>Number of segments ( K )</th>
<th>Deviance (-2 log ( L ))</th>
<th>Number of parameters ( d )</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>43192</td>
<td>9</td>
<td>43252</td>
</tr>
<tr>
<td>2</td>
<td>41503</td>
<td>18</td>
<td>41623</td>
</tr>
<tr>
<td>3</td>
<td>41464</td>
<td>27</td>
<td>41644</td>
</tr>
</tbody>
</table>

Table 1: BIC value for each segment solution

For each respondent, two posterior probabilities were computed which provided the probabilities that the respondent belonged to segment 1 and 2. The algorithm then allocated each respondent to the segment with highest posterior probability. In the two-segment model, 467 respondents were allocated to segment 1 and 311 subjects were allocated to segment 2.

5. RESULTS OF LATENT CLASS ANALYSIS

Table 2 displays the parameter estimates and standard errors for each segment solution.

<table>
<thead>
<tr>
<th>Term</th>
<th>Segment 1 Par. Est.</th>
<th>Segment 1 St. Error</th>
<th>Segment 2 Par. Est.</th>
<th>Segment 2 St. Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>3.752</td>
<td>0.091</td>
<td>3.153</td>
<td>0.081</td>
</tr>
<tr>
<td>Brand (A)</td>
<td>1.307</td>
<td>0.023</td>
<td>0.069</td>
<td>0.078</td>
</tr>
<tr>
<td>Brand (B)</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
</tr>
<tr>
<td>Price (€150)</td>
<td>0.118</td>
<td>0.034</td>
<td>1.942</td>
<td>0.118</td>
</tr>
<tr>
<td>Price (€175)</td>
<td>0.085</td>
<td>0.027</td>
<td>1.627</td>
<td>0.121</td>
</tr>
<tr>
<td>Price (€200)</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
</tr>
<tr>
<td>Int. access (Yes)</td>
<td>1.452</td>
<td>0.026</td>
<td>0.140</td>
<td>0.103</td>
</tr>
<tr>
<td>Int. access (No)</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
</tr>
<tr>
<td>Touch screen (Yes)</td>
<td>1.358</td>
<td>0.033</td>
<td>-0.123</td>
<td>0.104</td>
</tr>
<tr>
<td>Touch screen (No)</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
</tr>
<tr>
<td>Gender (Male)</td>
<td>-0.014</td>
<td>0.042</td>
<td>0.013</td>
<td>0.079</td>
</tr>
<tr>
<td>Gender (Female)</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
<td>alias</td>
</tr>
<tr>
<td>Age</td>
<td>-0.007</td>
<td>0.052</td>
<td>-0.014</td>
<td>0.045</td>
</tr>
<tr>
<td>No. of mobiles</td>
<td>-0.115</td>
<td>0.079</td>
<td>-0.037</td>
<td>0.083</td>
</tr>
</tbody>
</table>

Table 2: Parameter estimates and standard errors

Respondents in segment 1 have strong brand preferences but do not consider the price as a monetary constrain. These respondents give more worth to brand A mobile phones having touch screen facility and internet access. On the other hand, respondents in segment 2 are price sensitive but hardly discriminate between the brands. These respondents do not value much any of the mobile phone facilities and see no bargain in buying expensive phones. In both segments, the worth of mobile phones tends to decrease with an increase of user’s age and an increase in the number of mobile phones owned by user; however, both predictors are not significant at the 0.05 level of significance. The mean rating scores provided by males and females varied marginally across the levels of brand, price, internet access and touch screen facility.
REFERENCES


AUTHOR BIOGRAPHY

LIBERATO CAMILLERI studied Mathematics and Statistics at the University of Malta. He received his PhD degree in Applied Statistics in 2005 from Lancaster University. His research specialization areas are related to statistical models, which include Generalized Linear models, Latent Class models, Multilevel models and Mixture models. He is presently a senior lecturer in the Statistics department at the University of Malta.
USING DATA MINING FOR BANK DIRECT MARKETING: AN APPLICATION OF THE CRISP-DM METHODOLOGY

Sérgio Moro and Raul M. S. Laureano
Instituto Universitário de Lisboa (ISCTE – IUL)
Av.ª das Forças Armadas
1649-026 Lisboa, Portugal
E-mail: scmoro@gmail.com, raul.laureano@iscte.pt

Paulo Cortez
Centro Algoritmi, Dep. Sistemas de Informação
Universidade do Minho
4800-058 Guimarães, Portugal
E-mail: pcortez@dsi.uminho.pt

KEYWORDS
Directed Marketing, Data Mining, Contact Management, Targeting, CRISP-DM.

ABSTRACT
The increasingly vast number of marketing campaigns over time has reduced its effect on the general public. Furthermore, economical pressures and competition has led marketing managers to invest on directed campaigns with a strict and rigorous selection of contacts. Such direct campaigns can be enhanced through the use of Business Intelligence (BI) and Data Mining (DM) techniques.

This paper describes an implementation of a DM project based on the CRISP-DM methodology. Real-world data were collected from a Portuguese marketing campaign related with bank deposit subscription. The business goal is to find a model that can explain success of a contact, i.e. if the client subscribes the deposit. Such model can increase campaign efficiency by identifying the main characteristics that affect success, helping in a better management of the available resources (e.g. human effort, phone calls, time) and selection of a high quality and affordable set of potential buying customers.

BACKGROUND
Bank direct marketing

There are two main approaches for enterprises to promote products and/or services: through mass campaigns, targeting general indiscriminate public or directed marketing, targeting a specific set of contacts (Ling and Li 1998). Nowadays, in a global competitive world, positive responses to mass campaigns are typically very low, less than 1%, according to the same study. Alternatively, directed marketing focus on targets that assumable will be keener to that specific product/service, making this kind of campaigns more attractive due to its efficiency (Ou et al. 2003). Nevertheless, directed marketing has some drawbacks, for instance it may trigger a negative attitude towards banks due to the intrusion of privacy (Page and Luding 2003).

It should be stressed that due to internal competition and current financial crisis, there are huge pressures for European banks to increase a financial asset. To solve this issue, one adopted strategy is offer attractive long-term deposit applications with good interest rates, in particular by using directed marketing campaigns. Also, the same drivers are pressing for a reduction in costs and time. Thus, there is a need for an improvement in efficiency: lesser contacts should be done, but an approximately number of successes (clients subscribing the deposit) should be kept.

Business Intelligence and Data Mining

According to Turban et al. (2010), BI is an umbrella term that includes architectures, tools, databases, applications and methodologies with the goal of using data to support decisions of business managers. DM is a BI technology that uses data-driven models to extract useful knowledge (e.g. patterns) from complex and vast data (Witten and Frank, 2005).

The CRoss-Industry Standard Process for Data Mining (CRISP-DM) is a popular methodology for increasing the success of DM projects (Chapman et al., 2000). The methodology defines a non-rigid sequence of six phases, which allow the building and implementation of a DM model to be used in a real environment, helping to support business decisions (Figure 1).

![CRISP-DM process model](image)

Figure 1 The CRISP-DM process model (adapted from Chapman et al., 2000)
CRISP-DM defines a project as a cyclic process, where several iterations can be used to allow final result more tuned towards the business goals. After identifying the goal to achieve (Business Understanding phase), the data needs to be analyzed (Data Understanding) and processed (Data Preparation). According to Witten and Frank (2005), data has concepts (what needs to be learned), instances (independent records related to an occurrence) and attributes (which characterize a specific aspect of a given instance). The Modeling phase builds the model that represents the learned knowledge (e.g. given an instance, the model can be used to predict the target value that represents the goal defined). Next, the model is analyzed in the Evaluation phase, in terms of its performance and utility. For instance, in classification tasks (to predict a discrete target), common metrics are the confusion matrix (Kohavi and Provost 1998) and the Receiver Operating Characteristic (ROC) curve (Fawcett 2005). If the obtained model is not good enough for use to support business, then a new iteration for the CRISP-DM is defined. Else, the model is implemented in a real time environment (Deployment phase).

**Data Mining on Marketing Campaigns**

Given the interest in this domain, there are several works that use DM to improve bank marketing campaigns (Ling and Li, 1998)(Hu, 2005)(Li et al, 2010). In particular, often these works use a classification DM approach, where the goal is to build a predictive model that can label a data item into one of several predefined classes (e.g. "yes", "no"). Several DM algorithms can be used for classifying marketing contacts, each one with its own purposes and capabilities. Examples of popular DM techniques are: Naïve Bayes (NB) (Zhang, 2004), Decision Trees (DT) (Aptéa and Weiss, 1997) and Support Vector Machines (SVM) (Cortes and Vapnik, 1995).

To access the classifier performance, classification metrics, such as accuracy rate or ROC curve, can be used. Yet, for marketing campaigns, the Lift is the most commonly used metric to evaluate prediction models (Coppock 2002). In particular, the cumulative Lift curve is a percentage graph that divides the population into deciles, in which population members are placed based on their predicted probability of response. The responder deciles are sorted, with the highest responders are put on the first decile. Lift can be effectively used as a tool for marketing managers to decide how many contacts to do (from the original set) and also to check if, for some goal of target responses, there is an alternate better model.

**MATERIALS AND METHODS**

**Bank direct marketing data**

We collected data from a Portuguese bank that used its own contact-center to do directed marketing campaigns. The telephone, with a human agent as the interlocutor, was the dominant marketing channel, although sometimes with an auxiliary use of the Internet online banking channel (e.g. by showing information to specific targeted client). Furthermore, each campaign was managed in an integrated fashion and the results for all channels were outputted together.

The dataset collected is related to 17 campaigns that occurred between May 2008 and November 2010, corresponding to a total of 79354 contacts. During these phone campaigns, an attractive long-term deposit application, with good interest rates, was offered. For each contact, a large number of attributes was stored (e.g. see Table 2) and if there was a success (the target variable). For the whole database considered, there were 6499 successes (8% success rate).

**Computational Environment**

All experiments reported in this work were conducted using the rminer library (Cortez 2010), which is an open source package for the R tool that facilitates the use of DM techniques (and in particular, classification tasks), and rattle, a graphical user interface for DM in R (Williams, 2009). The rminer main advantage is that only a few set of coherent functions are required for a complex DM analysis. As an example, the following R/rminer code was used to build a DT and obtain a ROC curve (the rminer functions are underlined):

```r
library(rminer) # load the library
# read the data:
D=read.table("data.csv",sep=";",header=TRUE)
AT=c(6,7,9,12,17,20,22,25,27,30,31,36,43,46,48,51
,57,58,61,5)
D=D[,AT] # select some attributes
DF=na.omit(D) # remove missing data
M=mling(y~.,DF,method="holdout",2/3,model="dt ",Runs=20) # model several decision trees
# plot the ROC curve
mgraph(M,graph="ROC",Grid=10,baseline=TRUE)
```

In this paper, we used three DM rminer models: NB, DT and SVM (with Gaussian kernel), while the rattle tool was used for graphical data exploration, in Data Understanding phase.

**EXPERIMENTS AND RESULTS**

In this section, we explain the experiments performed and analyze the results obtained over a set of three CRISP-DM iterations.

**First iteration – project viability and goal definition**

Starting on the Business Understanding phase (of the CRISP-DM), it was clear that the goal was to increase efficiency of directed campaigns for long-term deposit subscriptions by reducing the number of contacts to do.

During the Data Understanding phase, we analyzed the data main characteristics. The output presented in the reports of previous campaigns was composed of two values: the result (nominal attribute with the possible values enumerated in Table 1) and the amount of money invested (numeric value in euro). For this research, only the nominal result was accounted for, thus the goal is to predict if a client will subscribe the deposit, not regarding
which amount is retained, turning it a classification task. Results are grouped together taking into account the type of contact, as shown in Table 1.

<table>
<thead>
<tr>
<th>Contact result</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Successful</td>
<td>Concluded contact</td>
</tr>
<tr>
<td>Unsuccessful</td>
<td></td>
</tr>
<tr>
<td>Not the owner of the phone</td>
<td>Cancelled contact</td>
</tr>
<tr>
<td>Did not answer</td>
<td></td>
</tr>
<tr>
<td>Fax instead of phone</td>
<td></td>
</tr>
<tr>
<td>Abandoned call</td>
<td></td>
</tr>
<tr>
<td>Aborted by the agent</td>
<td></td>
</tr>
<tr>
<td>Scheduled by other than the client</td>
<td></td>
</tr>
<tr>
<td>Scheduled by the client himself</td>
<td></td>
</tr>
<tr>
<td>Scheduled – deposit presented to the client</td>
<td>Scheduled contact</td>
</tr>
<tr>
<td>Scheduled – deposit not presented</td>
<td></td>
</tr>
<tr>
<td>Scheduled due to machine answer</td>
<td></td>
</tr>
</tbody>
</table>

The data attributes obtained from the campaign reports were most of them related directly to contact information (e.g. time and duration of contact, type of phone used, agent who made the call). Although this specific domain information may be of great value, clearly there was a need for collecting client information. Therefore, using internal databases of the bank institution, we collected extra attributes that better describe the clients’ characteristics (Table 2), resulting in a total number of 59 attributes (including the output target).

<table>
<thead>
<tr>
<th>Name</th>
<th>Description and Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Personal Client Information</strong></td>
<td></td>
</tr>
<tr>
<td>Age</td>
<td>Age at the contact date (Numeric ≥ 18)</td>
</tr>
<tr>
<td>Marital status</td>
<td>Married, single, divorced, widowed, separated (Nominal)</td>
</tr>
<tr>
<td>Sex</td>
<td>Male or Female (Nominal)</td>
</tr>
<tr>
<td><strong>Bank Client Information</strong></td>
<td></td>
</tr>
<tr>
<td>Annual balance</td>
<td>in euro currency (Numeric)</td>
</tr>
<tr>
<td>Debt card?</td>
<td>Yes or No (Nominal)</td>
</tr>
<tr>
<td>Loans in delay?</td>
<td>Yes or No (Nominal)</td>
</tr>
<tr>
<td><strong>Last Contact Information</strong></td>
<td></td>
</tr>
<tr>
<td>Agent</td>
<td>Human that answered the call</td>
</tr>
<tr>
<td>Date and time</td>
<td>Referring to when the contact was made</td>
</tr>
<tr>
<td>Duration</td>
<td>Of the contact (in seconds)</td>
</tr>
<tr>
<td><strong>First Contact Information</strong></td>
<td></td>
</tr>
<tr>
<td>Agent</td>
<td>Human that answered the call</td>
</tr>
<tr>
<td>Date and time</td>
<td>Referring to when the contact was made</td>
</tr>
<tr>
<td>Duration</td>
<td>Of the contact (in seconds)</td>
</tr>
<tr>
<td><strong>Visualization’s Information</strong></td>
<td></td>
</tr>
<tr>
<td>Number of times the client has seen the product in the home banking site</td>
<td></td>
</tr>
<tr>
<td><strong>History Information</strong></td>
<td></td>
</tr>
<tr>
<td>Result of the last campaign if another contact was made</td>
<td></td>
</tr>
<tr>
<td>Days since last contact in other campaign</td>
<td></td>
</tr>
</tbody>
</table>

After this analysis, some simple data pre-processing (Data Preparation phase) was performed and then the first execution of miner algorithms (Modeling phase) was conducted, to get some preliminary rough models. At this stage, only one model was obtained, through the use of the NB algorithm. For DT and SVM, the miner got out of memory or the processing did not end despite waiting several hours. Simplification (e.g. better data selection) was needed if knowledge was to be extracted successfully.

### Second iteration – goal redefinition

One of the hypotheses for the difficulty in obtaining models was the high number of possible output values, i.e. class labels. With this in mind, in the Business Understanding we transformed the output into a binary task, by using only the conclusive results of Table 1: successful and unsuccessful. It should be noted that for all the other results, there is always an uncertainty about client’s real intentions regarding the contact offer. Hence, the non-conclusive instances were discarded, leading to a total of 55817 contacts (the same 6499 successes).

After this goal redefinition, we were capable of testing the NB and DT algorithms in the R miner tool in the Modeling phase. However, there was still a large number of inputs to be considered (58), missing data (not handled yet), thus the predictive performances could be improved in another CRISP-DM round.

### Third iteration – variable and instance selection

We first assumed that there were several irrelevant input attributes that difficult the DM algorithm learning process (e.g. by increase of noise). To test this hypothesis, we went back to the Data Understanding phase and analyzed which attributes could influence the target. For this purpose, the rattle tool was used, in particular its graphical capabilities. For example, Figure 2 shows that the Sex attribute can be discarded, since the rate of successes for Male and Female is almost the same. With a similar analysis, we deleted half (29) of the considered inputs, leading to 29 input variables and 1 target output. We should note that while this graphical analysis may have flaws (e.g. there could be interactions of a particular Sex value with other attributes), the results of Table 3 do back this approach.

![Figure 2 Influence of Sex in the contact target](image-url)
Preliminary experiments, using NB and removing the attributes suggested by the Rattle analysis, also backed our manual feature selection procedure.

In addition to input attribute reduction, there were also several instances with missing values that were dealt with during the Data Preparation phase. While some DM models (e.g., DT) work well with missing data, there are others (e.g., SVM) that require missing data substitution or deletion. Since we had a large dataset, we opted to discard the examples that contained missing values, leading to a dataset with 45211 instances (5289 of which were successful – 11.7% success rate).

In this third iteration of CRISP-DM, during the Modeling phase, we managed to successfully test the three DM algorithms (i.e., NB, DT and SVM). Also, it is worth to notice that in all CRISP-DM iterations, the models were validated by using a holdout split, where the whole dataset was randomly divided into training (2/3) and test (1/3) sets. Given the large number of instances, 2/3 of them were considered good enough to build the models. To get more robust estimates of the performances, we applied 20 runs for each DM model in the second and third CRISP-DM iterations.

**Results**

Table 3 shows the predictive results for the test data during the three CRISP-DM iterations. The results are shown in terms of the mean value of the runs considered.

The AUC plots the False Positive Rate (FPR) versus the True Positive Rate (TPR) and allows identifying how good is the class discrimination: the higher the better, with the ideal model having a value of 1.0.

![Figure 3 ROC curves for the best predicting models](image)

![Figure 4 Lift analysis for the best predicting models](image)

Table 3 Predictive metrics for all the DM algorithms and CRISP-DM iterations

<table>
<thead>
<tr>
<th>CRISP-DM Iteration</th>
<th>1&lt;sup&gt;st&lt;/sup&gt;</th>
<th>2&lt;sup&gt;nd&lt;/sup&gt;</th>
<th>3&lt;sup&gt;rd&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instances × Attributes (Nr. Possible Results)</td>
<td>79354×59 (12)</td>
<td>55817 × 53 (2)</td>
<td>45211 × 29 (2)</td>
</tr>
<tr>
<td>Algorithm</td>
<td>NB</td>
<td>NB</td>
<td>DT</td>
</tr>
<tr>
<td>Number of executions (runs)</td>
<td>1</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>AUC (Area Under the ROC Curve)</td>
<td>0.776</td>
<td>0.823</td>
<td>0.764</td>
</tr>
<tr>
<td>ALIFT (Area Under the Lift Curve)</td>
<td>0.687</td>
<td>0.790</td>
<td>0.591</td>
</tr>
</tbody>
</table>

Overall, the results show that there was a clear evolution in prediction capabilities for models obtained, with each iteration resulting in better models than the previous one. In particular, the best predictive model is SVM, which provides a high quality AUC value, higher than 0.9.

To complement the ROC analysis, we also used the cumulative Lift curve area. Similar to the AUC, the random baseline classifier produces a 0.5 Area under the Lift curve (ALIFT), and the perfect ALIFT value is 1.0.

Both the ROC and Lift curves are shown in Figure 3 and Figure 4, respectively, for the three models obtained in the third iteration. Lift also favors the SVM model, which provides the higher cumulative lift area and whose curve is always above the remaining methods, whatever the selection of contacts.

to the contact selection to be loaded into the campaign, since some inputs are related to runtime contact execution, after the campaign has began. Nevertheless, we still believe it is useful to improve marketing campaigns. For instance, using a sensitivity analysis method (Cortez and Embrechts, 2011), we can characterize the SVM inputs influence in success. Figure 5 shows the importance of the five most relevant inputs in the SVM model.

Call duration is the most relevant feature, meaning that longer calls tend increase successes. In second place comes the month of contact. Further analysis can show (Figure 6) that success is most likely to occur in the last month of each trimester (March, June, September and
December). Such knowledge can be used to shift campaigns to occur in those months.

![Figure 5 Most relevant inputs of the SVM model](chart)

**CONCLUSIONS**

In this paper, we apply a Data Mining (DM) approach to bank direct marketing campaigns. In particular, we used real-world and recent data from a Portuguese bank and performed three iterations of the CRISP-DM methodology, in order to tune the DM model results. In effect, each CRISP-DM iteration has proven to be of great value, since obtained predictive performances increased. The best model, materialized by a Support Vector Machine (SVM), achieved high predictive performances. Using a sensitivity analysis, we measured the input importance in the SVM model and such knowledge can be used by managers to enhance campaigns (e.g. by asking agents to increase the length of their phone calls or scheduling campaigns to specific months). Another important outcome is the confirmation of open-source technology in the DM field that is able to provide high quality models for real applications (such as the rminer and rattle packages), which allows a cost reduction of DM projects.

In future work, we intend to collect more client based data, in order to check if high quality predictive models can be achieved without contact-based information. We also plan to apply the best DM models in a real setting, with a tighter interaction with marketing managers, in order to gain a valuable feedback.

**REFERENCES**


Li, W., Wu, X., Sun, Y. and Zhang, Q. 2010. “Credit Card Customer Segmentation and Target Marketing Based on Data Mining”. In Proceedings of International Conference on Computational Intelligence and Security, 73-76.


**BIOGRAPHIES**

**SÉRGIO MORO** (Computer Engineer from Instituto Superior Técnico) is a software engineer at a Portuguese bank and a MSc student at the Instituto Universitário de Lisboa (ISCTE-IUL).

**RAUL M. S. LAUREANO** is Assistant Professor at the Business School of the Instituto Universitário de Lisboa (ISCTE–IUL), Lisbon, Portugal and researcher at UNIDE (ISCTE-IUL), with research interests including Computer Skills Evaluation and Assessment, and Applied Statistics and Data Mining in Accounting and Management.

**PAULO CORTEZ** is Associate Professor at the Dep. of Information Systems of the University of Minho and researcher at centro Algoritmi, with interests in the fields of business intelligence and neural networks. His research has appeared in Journal of Decision Support Systems, Neurocomputing, and others (see http://www3.dsi.uminho.pt/pcortez).
INDUSTRIAL DECISION MAKING
A SIMULATION STUDY OF A
MULTI-SITE PRODUCTION PLANT USING ARENA

António Carvalho Brito (*)
Carlos Bragança de Oliveira (*)
Paulo Sá Marques (**) 
UGEI (*)
Faculdade de Engenharia
Universidade do Porto
INESC (**)
Rua Dr. Roberto Frias; P-4200-465
Porto, PORTUGAL
E-mail: acbrito@fe.up.pt
E-mail: braganca@fe.up.pt
E-mail: pmarques@inescporto.pt

KEYWORDS
Simulation, Modelling, Manufacturing, Arena.

ABSTRACT
The evolution in demand is leading industry market to seek for product orders in smaller lots and shorter deadlines. A need therefore arises to adapt industry production to these market changes. A portuguese plumbing and bath accessories company decided to study a new organization layout for its multi-site production plant. The objective was to evaluate different organization scenarios in order to rationalize the floor layout and reduce costs.

This Paper presents a study to evaluate different scenarios for a multi-site production plant using the Arena simulation software tool. The simulation model uses VBA (Visual Basic for Applications) to dynamically generate the simulation model. The product data is read from Excel spreadsheets or text files. Autocad is used for defining the plant layout and generating the working center coordinates. The Autocad drawing is used as the simulation background allowing the display of a realistic scaled animation.

INTRODUCTION
Computer simulation has become widely used in the analysis of complex problems and is now an important tool on evaluating production systems.

In 1976 the concept of Visual Interactive Simulation (VIS) was introduced by Hurrion (Hurrion 1976). With VIS an image of the model running is displayed on the screen and the user can interrupt the model running, at any time, and interact with it in a way that he can influence the future behaviour of the model.

Visual Interactive Simulation aggregates a set of graphics, visual and interactive techniques which can be incorporated with different levels of extension in a simulation system. VIS helps the analyst in visually identify the problems and in finding solutions. In the past years, there have been a large number of situations where VIS was successfully applied to different areas, including manufacturing and warehousing (Brito and Basto 1992), (Brito and Marques 2007), (Dangelmaier and Bachers 1986), (Marin et al. 1998), (Feliz-Teixeira and Brito 2003), (Burnett and LeBaron 2001) and (Macro and Salmi 2002). Most of the current simulation software incorporate interactive and graphical facilities and work has been done in interfacing simulation with CAD layouts (Vik et al. 2009). Simulation software has modules such as process, stations, transporters, conveyors, etc., to help design and analyse manufacturing systems. A survey can be found in (Smith 2003).

Arena (by Rockwell Software) is one of the most used simulation tools and has specific functionalities for modelling manufacturing systems. Some of the reasons why Arena was chosen for this study can be found in (Dias et al. 2011). A simulation model was developed in order to evaluate different scenarios for a multi-site production plant for a portuguese plumbing and bath accessories company.

THE FACTORY PLANTS
The company has four plants, two located at one site and the other two at another site, within a few miles distance. Each plant do some specialized work, such as foundry or coating but for the majority of the products the production path includes all the plants. This forces parts to be moved between the two sites as needed. Transportation is done by a truck twice a day. So, parts must be frequently moved between sites, which causes production delays and often damages in the parts.

The company wants to study different scenarios in order to reduce lead times and increase productivity. The main objectives are to reduce movement of the parts between sites and to reduce the number of workers that can be achieved by merging similar work centers located in different plants. The evaluation of the different scenarios was quite complex so the company took the decision to develop a simulation model.
THE CONCEPTUAL MODEL
The company produces almost thirty thousand products. It was not realistic to include in this study all the products, so it was decided to use families instead of single products. A family is defined as a set of products with similar characteristics. These characteristics include identical functionality, common operations sequence and similar product structure tree. Forty five families were identified but analyzing data it was concluded that only twenty five families had significant demand in the last few years.

Some concepts were used in defining the model. The Work Center (WC), described as the smallest production unit. A WC is a set of similar machines belonging to the same section. A section do similar work in the plant. The WC capacity is defined as the time available for production. The capacity of each WC is provided in hours per day. Only one 8 hours shift is considered. The number of resources in each WC is obtained dividing the number of available hours per 8. For example, if one WC works in two shifts with a resource, equivalent to 16 hours a day, it is transformed into a shift with two resources, equivalent to the same number of hours (16h). The work center costs are represented by a cost/unit of time which includes all the associated costs.

For each family the WC operations sequence were defined and the corresponding times of operations (minutes). The times of operations can be unitary or equal to the unitary time multiplied by the number of products (lot) to be produced. The family operations sequence is equivalent to the products operations sequence. This sequence was called natural sequence because it represents the common sequence of operations in the WC. Times of operations in the WC are computed as the weighted average of the family products times of operations where the weights are equal to the number of units sold of each product. This approach guarantees that the operation time in each WC is equivalent taking the individual products or the families.

The production is scheduled based on the monthly sales. A spreadsheet model transforms the monthly sales in daily sales. The model inputs are, for each family, the number of lots sold and the lot size. Another input required by the model is the WC coordinates. These coordinates are used to represent WC graphically and also to compute the travel distances between WC. The plant drawings are created in Autocad which is also used to generate the WC coordinates.

THE SIMULATION MODEL
For the development of the simulation model it was decided to use the Arena software, from Rockwell Automation Inc., because of its graphical interface and the availability of several manufacturing modules (parts, stations, routes, etc.). Another important reason to choose Arena was the availability of a programming language, VBA (Visual Basic for Applications). The amount of data needed to characterize the plants doesn’t allow data to be inputted manually. The VBA is used to read each scenario data and dynamically generate the model. The data is read from Excel spreadsheets and from text files. VBA modules are also used to define some of the model internal logic due to its complexity.

Arena uses the Process approach for simulation modeling. In this approach it is necessary to define the sequence of operations for each entity (family part). Each operation is modeled as a process which represents a WC and has properties that can be defined or accessed by an Arena form as shown in Figure 1.

Figure 1 - Process representing from WC: FF_ACAB001
As each scenario can have about 250 WC, it would be very time consuming to create the WCs manually. The process and its properties are dynamically generated based on a series of text files (see Table 1), most of them created by spreadsheet models.

Table 1 – Model input files

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacities</td>
<td>Available resources over time in WCs</td>
</tr>
<tr>
<td>CXY</td>
<td>WC coordinates, initial resources and costs</td>
</tr>
<tr>
<td>Production</td>
<td>Demand for each family over time</td>
</tr>
<tr>
<td>Routes</td>
<td>Possible links between WCs</td>
</tr>
<tr>
<td>RoutesTime</td>
<td>Family routes and WC’s operation times</td>
</tr>
<tr>
<td>FactorXy</td>
<td>Scale factor for importing Autocad file (.dx)</td>
</tr>
</tbody>
</table>

The sequence of Arena modules that are generated dynamically for each WC is similar to the one shown in Figure 2 for WC: FF_ACAB0001.

Figure 2 - Modules generated for WC: FF_ACAB0001
The enter and leave modules allow parts to be forwarded accordingly to the family operations sequence. This sequence is defined dynamically based on the data in the file “RouteTimes”. So, a modification in the family sequence implies only an update in the file and not a manual change in the Arena model, which will be much more difficult to accomplish. Each process module has a queue associated, represented by a line (see Figure 1),
where the parts wait until they can be processed. The queue waiting rule is by default FIFO (First In First Out) but a decide module was added (see Figure 2) to allow other rules to be used. This module uses an entity attribute to sort the entities in the queue. Two rules were used: MOR (Most Operations Remaining) and LPT (Longest Processing Time).

THE SIMULATION RUN
The model data files can be edited directly or generated by Excel spreadsheet models. Before running the simulation there is an initial step to generate an intermediate model, where the plants layouts only if the plants layouts are defined by importing selected Autocad drawings. At this stage the WCs positions are also defined based on the xy coordinates imported from Autocad. This step must be carried out again only if the plants layout or WC position are modified. Figure 3 shows the simulation screen.

Figure 3 - Simulation screen
The following step is to generate the final model. This step must be repeated if the family operations sequence is modified. Several models can be generated, repeating this procedure, in order to analyze different scenarios. The Arena model setup option should be used to define the replication parameters. A common value used in this study was running the model for one year (108960 minutes = 277 days x 8 hours x 60 minutes). The simulation can be run with animation - more detail is available, or without animation - the simulation runs faster. The Autocad drawings are used as a background for the simulation. This allows a scaled and realistic animation of what is happening in the plants. Parts (moving on the screen following the family path sequence from one WC to the next) can be seen. If an operation is needed in a WC located at another plant the part must wait for transportation. The truck moving from one plant to another can also be seen during the simulation. The simulation screen also includes the plants scaled layout where the WC are displayed, the trucks for moving parts between sites as well as the simulation clock. During simulation it is possible to visualize any part of the layout using the Arena zoom command. Figure 4 shows a zoom detail of one of the plants. In the display each individual WC can be identified by its name (see Figure 4).

Figure 4 – Simulation detail
For each WC there is a colour sign which is red if the WC is busy or green if it is idle. Near the WC name there is also a text box showing the number of parts waiting to be processed. This can be quite useful to identify bottlenecks during simulation. The movement of the parts are displayed by a circle painted with the family colour and the family name attached to it. In order to be able to follow this detail the simulation speed must be reduced using the arena control buttons. Table 2 shows the time taken to run the simulation using a Pentium IV at 3.2Ghz with 1Gbyte of RAM and Windows XP with different levels of simulation speed.

Table 2 – Duration at different simulation speeds

<table>
<thead>
<tr>
<th>Animation</th>
<th>Simulation speed (N° of times faster)</th>
<th>N° of minutes taken to simulate one year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inactive</td>
<td>52000</td>
<td>2</td>
</tr>
<tr>
<td>Fastest</td>
<td>5750</td>
<td>19</td>
</tr>
<tr>
<td>Slowest</td>
<td>160</td>
<td>681</td>
</tr>
</tbody>
</table>

MODEL VERIFICATION AND VALIDATION
The conceptual model and the assumptions made were deeply discussed with company professionals during several meetings in order to obtain an agreement. The verification allowed to assure that the simulation model is a good representation of the conceptual model. The visual inspection of the simulation run was very useful for the model validation. The professionals could participate and give their inputs and criticism contributing for improving the model. Because of the large volume of data an Excel spreadsheet model was developed to compute the expected
results for each scenario. These results were then compared to the simulation results allowing to correct the errors found. The data included total times for WC and product family, available capacity and costs.

RESULTS

Although other scenarios were studied we mentioned here only two: the initial scenario (A) and the one site scenario (D) with two plants. Several configurations were used. Unitary and lot production with or without setup times. To evaluate the behaviour of the model in an ideal situation a configuration with infinite capacity (n° of resources x 10) was used. The same data and configurations were used in both scenarios to be able to compare the results. The family lot size used in both scenarios are showed in Figure 5.

![Figure 5 - Families lot size](image)

Figure 6 shows the number of lots for each family for one year production.

![Figure 6 - Nº of family lots for one year production](image)

The results obtained for scenario D didn’t show significant improvement relative to scenario A. Figure 7 compares the number of lots ordered with the number of lots produced in scenario A and D. The difference between the number of lots produced in scenario D is only marginally greater (0.5%) than the number of lots produced in scenario A. This can be explained by the small influence of the delay introduced by the transportation between sites.

![Figure 7 – Nº of lots ordered and produced](image)

Nevertheless, the scenario D has other advantages because it eliminates duplicate WC and eliminates the transportation costs between sites. These costs were estimated to be around seventy thousands euros per year. Another study carried out was to balance the work load from the WCs. So, it was decided to increase the resources of the WCs that presents an utilization rate greater than 80%. The resources were increased until the utilization rate drop down to near 80%. On the other side, the resources were reduced of the WCs where the utilization rate was less than 50%. The resources were reduced until the utilization rate reached 50%. As the number of resources are integers there are certain situations that the reduction is not possible. The relative increase in the number of lots produced is showed in Figure 8.
Figure 8 – Variation (%) of the n° of lots produced In this scenario the overall capacity was reduced by 20% and the estimated cost reduction was equal to 10,5%.

5. CONCLUSION
The use of simulation packages, such as Arena, can be very helpful in analyzing and improving the performance of production plants. In Arena the availability of modules such as process, stations, transporters, etc. makes the development of the simulation model easier. However, there are systems, as the one presented here, where the use of a programming language, like VBA, is necessary to model the complex logic and dynamically generate the simulation model. The use of colours, graphics and animation also helped to validate the model and see why problems arise during the simulation run and also to interpret the simulation results.

The simulation project objectives were achieved. The existence of two sites doesn’t seem to be a relevant drawback if careful plant management is done. Nevertheless, the transport between the sites introduces a delay and if eliminated result in annual cost savings if just one site were considered. The study shows that there are significant improvements in balancing the load of the WCs. This can be done by increasing the resources in the WC identified as bottlenecks, and reducing the resources in the WC with low utilization rates. This solution also reduces the overall production costs.

As future work, it is expected to further develop the VBA programming to automatically generate simulation models for production plants. This will allow a faster development of the models and more flexibility to evaluate different alternatives.

REFERENCES

Acknowledgements:
The authors would like to thank to CIFIAL for all the support given to this work and to the project team from INEGI and INESC. Special thanks to the persons involved directly with the project: Ludgero Marques, Luís Marques, Luis Rodrigues, António Oliveira, Vítor Mota, Jorge Cruz and Elisa Cruz from CIFIAL, João Paulo Pereira, Jorge Teixeira da Silva e Silvia Esteves from INEGI and Manuel Feliz Teixeira.

AUTHOR BIOGRAPHIES
ANTÓNIO CARVALHO BRITO was born in Porto in 1958, and went to Universidade do Porto, where he studied mechanical engineering and obtained his degree in 1981, staying in the same institution as Lecturer where he did his M.S. thesis in 1985. He obtained his Ph.D. in Simulation from the School of Management at the University of Cranfield (U.K.) in 1993. He is Assistant Professor at Universidade do Porto since 1993, where he leads research in the field of Visual Interactive Simulation. His e-mail address is: acbrito@fe.up.pt

CARLOS BRAGANÇA DE OLIVEIRA was born in Vila de Cucujães (Portugal) in 1958, and went to Universidade do Porto, where he studied mechanical engineering and obtained his degree in 1981, and his M.S. thesis in 1987 staying in the same institution as researcher until 1992. He was teacher in several Universities and is Invited Assistant Professor at Universidade do Porto since 2007, where he do research in the field of Agent Simulation. His e-mail address is: braganca@fe.up.pt

PAULO SÁ MARQUES was born in Porto (Portugal) in 1963, and went to Universidade do Porto, where he studied electrical engineering, and obtained his degree in 1988. He is in INESC Porto since 1998, where he is now project manager. His e-mail address is: pmarques@inescporto.pt
MDE IN ACTION: FIRST STEPS TOWARDS A CROP MODEL FACTORY

Guillaume Barbier\textsuperscript{(1,2,3,4)}, François Pinet\textsuperscript{(5)}, David R.C. Hill\textsuperscript{(2,3,4)}

\begin{itemize}
  \item (1) ITK SAS, 5 rue de la cavalerie F-34000 Montpellier, guillaume.barbier@itkweb.com
  \item (2) Clermont Université, BP 10448, F-63000 Clermont-Ferrand
  \item (3) CNRS, UMR 6158, Université Blaise Pascal, LIMOS, F-63173 Clermont-Ferrand
  \item (4) ISIMA, Computer Science & Modelling Institute, BP 10125, F-63177 Clermont-Ferrand
  \item (5) Cemagref, UR TSCF, BP 50085, F-63172 Aubière
\end{itemize}

KEYWORDS

Crop Model, Process Based Models, Model Driven Engineering, Visual Modelling Framework

ABSTRACT

In an industrial context of web-based decision support system for croppers, agronomists in our firm design crop models using a scientific programming language. Then in a second step this model is transposed in Java by computer engineers. Due to keeping both codes in evolution, this double implementation causes losses in cost production. To avoid such costs, this paper proposes to apply model driven engineering techniques to identify a metamodel for the domain of mechanistic modelling in agronomy. We started by a reverse engineering of three existing model implementations. The resulting metamodel is the basis for the definition of a mechanistic model factory dedicated to agronomists. This visual tool will offer automatic model linking and data conversion features: they rely on the definition of metadata associated with the inputs/outputs of the different models and on the use of adapters. These automation possibilities should grant easy design capabilities and optimal model reuse in the future. The designed models are linked to the metamodel proposed thus permitting the generation of a ready-to-use Java implementation of the mechanistic model.

INTRODUCTION

ITK SAS main activity is to develop web-based decision support systems (DSS) for crop or wine grape producers. These DSS rely on mechanistic models, which aim at modeling the physiological processes occurring during the plant growth. It is indeed believed that a good representation of the different biological processes occurring in the field will help identify the best suited management option and its appropriate timing.

Difficulties arise when trying to have two different worlds meet. On one side scientists, mainly agronomists, use scientific programming languages (Matlab, R) to develop the scientific model. On the other side computer engineers have to transpose it on a Java platform for industrial use. This development process is time consuming and costly. To optimize it we propose to design a mechanistic model factory for agronomists. This paper introduces the first step towards this long-term goal.

There is a wide diversity of mechanistic models in agronomy, being influenced by the precision with which the plant topology is described:

- Big leaf models: considering the plant to be one single leaf like the STICS crop model which has been adapted to several plant species (Brisson et al. 2009).
- Leaf layered models (Lawless et al. 2005).
- Functional structural plant models (FSPM) being the most detailed approach with explicit
consideration of the plant architecture (Vos et al. 2007).

Aside from the plant topology, the way the different biophysical processes are modeled and organized, their scale (with regards to space and/or time) is responsible for most of the mechanistic models diversity. This diversity comes from: the aim of the model, the expected accuracy, the lack of knowledge on the plant biology or even the specificities of some plant species like floral initiation or vernalization (a physiological phenomenon linked to the exposure of seedlings to low temperature having an effect on the plant development and flowering).

In the scope of our work, we consider three models (wheat, cotton and vine) which have been developed based on different published models (Jallas 1998; Jamieson et al. 1998; Louarn 2009). These three models encompass the diversity of crop models with different data structures, processes and ways to take them into account. The wheat model is derived from the leaf layered model Sirius (Jamieson and Semenov 2000; Lawless et al. 2005) including specificities on vernalization and floral initiation (Gate 1995) and a soil tank model.

The vine model and the cotton model are topological ones. The former includes sun interception (Riou et al. 1989), evapotranspiration (Feddes et al. 1978; Brisson and Perrier 1991) and different water soil models from the classic tank model to a two-dimensional model obtained from Kamel (Braudou et al. 2009). The latter is much more precise on the plant topology: leaves, flowers and branches have each their specific position (Jallas 1998).

A short presentation of MDE (Section 1) shows the importance of identifying the specific perimeter of the domain of work. Study of the three models shows strong similarities between mechanistic models in agronomy and process based models (PBMs) linked with a specific data structures (Section 2). In our context the plant data structure is easily defined by a class diagram (Section 3) and linked to a metamodel that will be used for the definition of the PBM part of the crop model. The importance of this design is emphasized for reuse purposes, model design assistance, with a focus on the Adapter class on which an important part of the functionalities relies (Section 4). A graphical user interface (GUI) for crop modelers is proposed. It will permit design and reuse of crop models with built-in functionalities for model linking, conversions, time scaling. This tool will generate a ready-to-use Java implementation (Section 5).

MODEL DRIVEN ENGINEERING

MDE is an integrative approach of software production. It aims at covering the whole software lifecycle. According to (Greenfield and Short 2003) the object-oriented paradigm has reached its limits, which is echoed by (Bézivin 2005), both references state that the “everything is a model” paradigm should lead to the industrialization of software production.

As was the case for OOP in its beginnings with the “what is an object?” debate, the beginning of MDE raised the “what is a model?” question. Based on (Minsky 1965) definition, (Bézivin and Gerbé 2001) state that “a model is a simplification of a system built with an intended goal in mind. The model should be able to answer questions in place of the actual system.” We would have added “in the restricted scope of the chosen simplification”. This definition introduces the first basic relation of MDE: RepresentedBy. A system is represented by a model which gathers only the elements of the real system needed to answer the modeler’s questions.

As for the simplification of the system, the simpler is the representation the easier will be the specification of any kind of automation. The modeler is to focus on a very specific aspect of the software lifecycle or application functionalities. She/He is restricting that way his modeling effort to a specific domain. This will permit the definition of a Domain Specific Language (DSL) (Greenfield and Short 2003; Booch et al. May 2004) whose grammar is ruled by a metamodel. Any model in the specific domain scope ConformsTo the metamodel. This conformity relationship is the 2nd main relation and is the essence of MDE. The metamodel is required to be concise in order to carry out transformation operations which represent the third main MDE concept. Transformations may be executed at the same level of abstraction from a model to another one, or from one level to another one (Hill 1996). Complex transformations may be achieved by pipelining basic ones (Zef Hemel et al. 2008).

(Favre et al. 2006) explain how reverse engineering can be used to recover knowledge of
the legacy domain. In the context of this paper the legacy is the three model implementations cited above. Using informal reverse engineering, design recovery and restructuring as defined by (Chikofsky and Cross 1990) the following domain definition and architecture proposal have been obtained.

OVERVIEW OF THE SPECIFIC DOMAIN: MECHANISTIC MODELING IN AGRONOMY

Mechanistic modeling in agronomy is a widespread approach in plant modeling (Marcelis et al. 1998). Its origins are found in PBM's approach commonly called crop models. Mechanistic modeling encompasses also the FSPMs (Functional Structural Plant Model) (De Reffye and Hu 2003). The main difference between these modeling approaches is that crop models ‘provide a synthesis of knowledge in a simplified form’ (Brisson et al. 2006). Whereas in FSPM the organs and processes are considered with great detail (Cournède et al. 2008). Though a powerful tool for research, FSPM remains unsuited for DSS that is why we will have our main focus on crop models.

Basically PBM's in agronomy are relying on the modeling of the different physical, chemical or biological processes occurring in the real system. Those processes are responsible for the changes in the system structure and the exchange of matters and/or energy between its different parts. They are interdependent meaning they use each other’s outputs to produce their own. They are time dependent: with a time scale of their own. They rely on a variable plant structure (big leaf, stratified canopy or detailed topological representation). They are ruled by mathematical and conditional expressions.

Flowcharts representations are quite common in mechanistic modeling in agronomy (Lawless et al. 2005; Willocquet et al. 2008; Hakojarvi et al. 2010). In most cases, the processes are considered sequentially and always are in the context of our DSS. It is an important point meaning that the models we use may be considered as workflows. That is why we focus our study on the processes with the intent of producing a generic platform permitting the design of mechanistic models with a flow-like representation. Thus we consider a simple plant data structure as being a common blackboard for the different processes.

STATIC MODEL OF THE CROP SYSTEM

Decomposing the plant structure to get a hierarchical view is straightforward for computer engineers (ep. Figure 1). This kind of class diagram describing the plant state is quite common but as stated above adding behavior to those classes is not possible. Depending on the choices of modelers, transpiration may be applied at the scale of the stomatas, leaf, plant or even to the field. Thus we want to distinguish the plant data view from the associated dynamic processes.

Figure 1: Class Diagram of the Plant Data Structure
DEALING WITH THE DYNAMICS

In this section we present the metamodel (cp. Figure 2) associated to the domain of mechanistic modeling in agronomy stressing out the processes view and pinpointing its main features in the following.

Main Characteristics of the Processes

Regardless of its biological meaning, each process may be characterized by its inputs, its state variables, its parameters and mathematical expressions. All of them are used to produce outputs for each time step specific to the process (cp. Figure 3). This kind of representation is quite common in the world of modeling (Zeigler 1976).

Hierarchical Decomposition of the Processes

As in DEVS where we find the notion of composed models, we consider the concept of hierarchical decomposition of processes. A process is either a complex one, its sole role is to organize logically its sub-processes, or an atomic one, which contains mathematical and logical expressions, state variables and parameters, to process its inputs and produce its outputs. The use of the composite design pattern is then well suited to represent this structure. It will also prove to be of interest for initializing the simulator. Parsing the whole hierarchy of the processes is in this case an easy thing and will permit the flattening of the workflow.

Data: Expressing more than a Value

In a general way modeling is about using data (inputs) to produce data (outputs) as a result of data (state variables and parameters). When thinking about data the first thing coming in mind is values. But it should not be restricted to this. There is a set of expert knowledge that is linked to data with regards to its type, its description, its units and acceptable range of values. We consider those as metadata which is useful for:

- Reusability: each process and its associated data contain sufficient information to inform the modeler of its behavior and the needed inputs and parameters for its execution.
- Linking assistance: as inputs and outputs have their own units, it is possible to propose to link inputs and outputs which are present at same level of the model hierarchy.
- Conversion assistance: the Adapter class (see below) relies on the metadata to propose a fine unit and scale adaptation.
Multi-Scale Modeling

Thanks to this separation between the dynamic aspects and the plant data structure it is possible to integrate processes at different spatial scales in a given model and also to reuse a process and to apply it at a different scale with minor changes in the process configuration (e.g. from leaf to field).

As for the time scale we may integrate processes having different time dependencies with a constraint on the time steps between a given model and the one containing it. The lower level model time step has to be a divisor of the time step of the upper level model.

Adapting without Changing

In the scope of reusability and automation, we integrate the adapter concept. Reusing models from different sources and plugging them might result in having to link an input and an output -which means they represent the same information- but their units might be different. This leads to the need to convert the output to produce data as expected by the second model. Rather than changing the first process to have the expected data produced with the risk to have a cascade of changes to realize, we introduce an adapter. The Adapter class is in charge of the data conversion and also of decomposing or aggregating data from processes having different time steps. As the object standing for the output is unaware of the existing adapter the Observer design pattern is used to notify the adapter of changes done to the output value. It is an essential design choice for enabling the aggregation of values produced at a shorter time step than the one of the process using those values as a single input.

Flowcharts representations and design assistance under the modeler supervision are naturally leading us to considering a visual tool for the design of mechanistic models.

VISUAL TOOLS FOR DESIGNING MECHANISTIC MODELS

Existing Tools

A lot of tools already exist in the world of visual modeling, some unspecific with regards to the domain of modeling like Simulink®, Simile (Muetzelfeldt and Massheder 2003). Different tools have been developed based on Modelica (Mattson and Elmqvist 1997) like MathModelica® or MapleSim®. Some have been developed in the scope of modeling in agronomy like the OpenAlea platform (Pradal et al. 2008).

The previous tools do not fulfill our needs given that they are either dedicated to the component approach with the reuse of models being developed by different teams under different technologies. So that one needs to either develop her/his own model and integrate it in the platform or reuse existing components. Or even to use a GUI which forces her/him to explicitly specify the different elements of the model. This step is laboriously done and results in a graph of models hardly readable. Even if some of these tools may be used to generate Java code embedding scientific code written in another language they remain unsuited for industrial use. For instance the Matlab® compiler runtime is thread locking which means facing bottleneck issues in a multi-user environment. Moreover embedding code has its constraints. For example when willing to add to the model outputs variables already computed it is required to modify the scientific model implementation. Such modifications imply to validate the model again. Whereas it is easily done using configuration file and the Observer design pattern in a full Java implementation. We may add to those elements the need to have full control on the code produced for optimization, persistence or log purposes.

Our Proposal

We wish through our proposal to define a design environment permitting to specify a whole model with visual operations inducing transformations to the underlying object model. The aim is to win the modelers support with an easy to learn tool, retaining the full power of mathematical expressivity and logical expressions for the atomic models, and to grant them with an easy way of combining the processes and linking them with visual elements.

Figure 4 is a snapshot of what the GUI should be. It shows as an example how the leaf dynamic process of our wheat model could be defined. This process is relying on organic matter and daily temperature inputs (left side of the scheme). It is divided into three processes:

- Creation: detecting when a new leaf is produced and adding it to the plant data structure.
• Senescence: computing the loss of leaf area through aging. This process comes prior to the leaf growth because of remobilization of organic matter (part of the matter of the dying leaf is reused for the growth of the younger ones).

• Growth: depending on the available organic matter and daily temperature computes the gain of leaf area.

Figure 4: Scheme of the Proposed GUI of the Mechanistic Model Factory

The thick arrows show the call sequence of the processes. The links with a circle in their middle stand for linked input and output and an optional adapter. Unlinked input/output is shown as a specific icon linked to a model it is associated with. On the right side of the screen the whole model hierarchy is shown in a browser like fashion. This way the modeler can easily parse the model to link an input/output of one of the processes currently being edited with its counterpart at another level of the hierarchy. Adding the input/output to the upper level models for the sake of a clean linking is automatic.

Some Underlying Mechanisms

Any modification on the diagram has consequences on the underlying class diagram. Should the user create a new model via the bottom right corner tool panel, it results in generating a new class inheriting from the CompositeModel or AtomicModel class depending on the choice she/he makes. Once inserted in the call sequence the generated class is associated to the upper level model. The following (Code 1) is an example of code generated from the proposed leaf dynamic model (cp. Figure 4).

**Code 1:**

```
public class LeafModel extends CompositeModel{
    private List<Data<?>> inputs;
    private List<Data<?>> outputs;
    private ArrayList<Model> subModels;

    public LeafModel(){
        inputs.add(new LeafCarbonInput());
        inputs.add(new LeafDailyTemperatureInput());
        subModels.add(new LeafCreationModel());
        subModels.add(new LeafSenescenceModel());
        subModels.add(new LeafGrowthModel());
    }
}
```

Let us now imagine that our leaf dynamic model is linked to a photosynthesis model which computes the organic matter produced at an hourly time step whereas the former uses a daily time step. To link the carbon output of the latter to the carbon input of the former we need an adapter which will sum up the 24 outputs of the photosynthesis model. The values are notified to the adapter thanks to the use
of the observer design pattern. The resulting (simplified) generated code is as follows (Code 2).

**Code 2:**

```java
public class CarbonAdapter extends Adapter {
    private Data input = LeafCarbonInput.getInstance();
    private Data output = PhotosynthesisOutput.getInstance();

    private int[] value;
    private int timeRatio = 24;
    public void transform() {
        int convertedValue = 0;
        for (int i = 0; i < 24; i++) {
            convertedValue += value[i];
        }
        input.setValue(convertedValue);
    }
}
```

Hand-coding the `Adapter` class for any link between input and output requiring conversions would prove to be a laborious task. Conversions formulas will be automatically generated relying on the metric system and the specified units of the inputs and outputs and proposed to the modeler. She/he may agree upon them or adapt them to her/his specific needs. In a similar fashion aggregation and disaggregation operations will be proposed when suited. The adapter implementation will then be automatically generated.

**A Suitable Environment: the Eclipse Platform**

Given the objective of building a crop model factory relying on MDE techniques and our technical context (JEE development), the Eclipse Platform seems to be the most suitable environment to build the factory. Using the Eclipse Modeling Framework (EMF) (Steinberg et al. 2008) and the Graphical Modeling Framework (GMF), a first tentative user interface has been produced which seems promising though further exploration of customization possibilities still needs to be done. The transformations between the model at design level and the model dedicated to simulations are still to be addressed. QVT0 or ATL will most probably be used. But the simulation model will also require specific code generation for the mathematical and logical expressions at the atomic level.

**CONCLUSION**

In this paper we made a first proposal of a metamodel with the intent to create a mechanistic model factory in the field of agronomy. Excluding FSPM from our study because of its lack of adequacy for decision support in crop management, we have been able to narrow down the domain to crop models. This restriction allows the definition of our metamodel which relies on a first informal reverse engineering and refactoring of three model implementations covering most of the diversity of crop models.

The definition of the graphical interface with boxes and lines seems natural given the PBM nature of crop models and the way modelers adopt flowchart representations for the design of their models. The focus of this study is not on the graphical formalism of the visual tool. We want to keep it as simple as possible to facilitate its adoption by modelers and to adapt its interface depending on the modelers’ needs. Automation of processes which can be cumbersome when hand-coded, like linking models or converting data should favor its adoption. The choice of using `Data` and `Adapter` classes makes possible to achieve these operations without additional effort on the modelers’ part. Code generation for conversion is easily done given the information provided by the `Data` classes and the possible tuning by the modelers. As linking models and conversion are the main operations achieved when reusing models, this proposal favors the reuse of models formerly designed.

Not only do we propose a technical solution to the issue of having a double implementation of the same model in two different languages avoiding that way the problem of double development and maintenance cost. But we also propose a way to make mechanistic model design an easier thing for modelers through automation, easy reusability and multi-scale considerations. The definition of metadata associated with the different elements of the model offers also the possibility to have automatic and up-to-date documentation with regards to the scientific model.

The next step will be to test our metamodel on the three models we used to build this first proposal and refine it if needed. Improvement of the specifications of the interaction between the processes and the plant data structure is also
required. We have to state the way the data structure is integrated in the generated model implementation and how the modeler is interacting with it at the GUI level.

REFERENCES


USE OF THE BELIEF THEORY TO FORMALIZE AGENT DECISION-MAKING PROCESSES: APPLICATION TO CROP PLAN DECISION-MAKING

Patrick Taillardier
MTG Lab, UMR-IDEES 6228
1, rue Thomas Becket,
Mont-Saint-Aignan, France
E-mail: patrick.taillardier@univ-rouen.fr

Olivier Therond
INRA - UMR 1248 AGIR
Chemin de Borde Rouge
Castanet Tolosan, France
E-mail: olivier.therond@toulouse.inra.fr

KEYWORDS
Multi-Agent Simulation, Agent Behavior Formalization, Belief Theory, avian flu propagation.

ABSTRACT
The agent-based simulation is a powerful tool to study complex systems. It allows to take into account different levels of granularity, as well as the heterogeneity of the entities composing the system. One of the main issues raised by these simulations concerns the design of the agent behavior. Indeed, when the agent behavior is led by many conflicting needs, desires and necessities, its definition can be very complex. In order to address this issue, we propose a new formalism to define the agent decision making process. This formalism is based on the belief theory, which is a formal theory about reasoning that allows to manage information incompleteness, uncertainty and imprecision. An application of the approach is proposed in the context of a model dedicated to cropping plan decision-making.

INTRODUCTION
Agent-based simulations are now widely used to study complex systems. It allows to take into account different levels of granularity, as well as the heterogeneity of the entities composing the system. However, the problem of the agent design is still an open issue. Indeed, designing complex agents able to act “relevantly” in the simulation is a difficult task, in particular when their behavior is led by many conflicting needs and desires.

In the agent community, numerous formalisms were proposed to model the agent decision-making process: logical formalisms, probabilistic formalisms and modal formalism (see (Das 2008)). However, most of these formalisms are not of much use for agent-based simulations. A reason is their inadequacy to the simulation context: a formalism, to be used in simulation, has to allow thousands of agents to make a decision from many criteria in a short amount of time. Moreover, it has to be easily understandable and usable by domain experts that are often not computer-scientists.

In this paper, we propose a new probabilistic formalism that is particularly well-fitted for agent-based simulations. This formalism is based on the use of utility functions. A utility function is a function that maps a decision to a real number. The more a decision maximizes the utility function, the more it has a chance to be chosen by the agent. This type of formalisms is particularly well-fitted for agents whose decisions depend on numeric variables.

In order to compute the utility of each decision, we propose to use the belief theory. This theory allows to formalize reasoning. It can be used to make a decision between several alternatives according to a set of criteria. An advantage of this theory is that it allows to make a decision even with incompleteness, uncertainty and imprecision.

In the next section, we introduce the general context of our work.

CONTEXT: UTILITY FUNCTIONS AND AGENT DECISION-MAKING PROCESS

The use of utility functions for agent decision making is a classic approach that was used in many works (e.g. (Lang et al. 2002)). In this context, the utility of a decision is often linked to its expected outcomes: the expected utility of a decision is then computed from the resulting states of the decision’s possible outcomes and from their probability of happening (Von Neuman and Morgenstern 1947).

For some applications, the decision cannot be evaluated by a unique attribute (or criterion). The decision making process of the agents will then consist in solving a multi-criteria analysis problem: an agent has to make a decision according to a set of criteria that will represent its needs and desires. In the literature, several approaches were proposed to solve this type of multi-criteria decision-making problems.

A first family of approaches, called partial aggregation approaches, consists in comparing the different possible decisions by pair by the mean of outranking relations (Figueira et al. 2005; Behzadian et al. 2010).

Another family of approaches, called complete aggregation approaches, consists in aggregating all criteria in a single criterion (utility function), which is then used to make the decision (Jacquet-Lagreze and Siskos 1982).

A last family of approaches, which is highly interactive, consists in devising a preliminary solution and comparing it
with other possible solutions to determine the best one (Izigio 1978).

Partial aggregation approaches allow to address the problem of criterion incompatibility but lack clarity compared to complete aggregation approaches (Ben Mena 2000).

The approach we are interested in belongs to the complete aggregation approaches. It is built on the belief theory. In the next section, we describe this approach and its application for the agent decision making process.

**USE OF THE BELIEF THEORY TO FORMALIZE THE AGENT DECISION MAKING PROCESS**

**Multi-criteria decision making using the belief theory**

**Generality**

The belief theory, also called Dempster-Shafer theory, was proposed by Shafer in 1976 (Shafer, 1976). It is based on the Theory of Evidence introduced by Dempster (Dempster, 1967), which concerns the lower and upper probability distributions. It allows to manage incompleteness, uncertainty and imprecision of data. It has been used with success for many applications (e.g. (Olejnovici-Raimond and Mustiere 2008; Taillard et al. 2009)).

The belief theory first defines a frame of discernment, noted \( \Theta \). It is composed of a finite set of hypotheses corresponding to the potential solutions of the considered problem.

\[ \Theta = \{H_1, H_2, ..., H_N \} \]

From this frame of discernment, let us define the set of all possible assignments, noted \( 2^{\Theta} \):

\[ 2^{\Theta} = \{ \emptyset, \{H_1\}, \{H_2\}, ..., \{H_i\}, \{H_j\}, ..., \Theta \} \]

Each set \( \{H_i\}, \{H_j\} \) represents the proposition that the solution of the problem is one of the hypotheses of this set.

The belief theory is based on the basic belief assignment, i.e. a function that assigns to a proposition \( P \) with \( \Pr \in 2^{\Theta} \), a value named the basic belief mass (bmm), noted \( m(P) \). It represents how much a criterion \( j \) -called source of information-supports the proposition \( P \). The bmm is ranged between 0 and 1 and has to check the following property:

\[ \sum_{p \in 2^{\Theta}} m_j(P) = 1 \]

**Decision making approach**

In our agent decision making context, each hypothesis represents the fact that a decision of the set of decisions \( D \) is the best one. For example: \( \{H_1\} \): the best decision of \( D \) is \( d_1 \), \( \{H_2\} \): the best decision of \( D \) is \( d_2 \), \( \{H_3\} \): the best decision of \( D \) can be either \( d_1 \) or \( d_2 \), etc.

The decision making process is composed of four steps.

**Step 1**

This first step consists in initializing the basic belief masses. For this step, we propose to use the works of Appriou (Appriou 1991). He proposed to “specialize” the criteria for one hypothesis of the discernment frame. Thus, the criteria give one’s opinion only in favor of a hypothesis, in disfavor of it or do not give their opinion. For each hypothesis \( H_i \) of \( \Theta \), a subset \( S_i \) of \( 2^{\Theta} \) is defined:

\[ S_i = \{ \{H_i\}, \{-H_i\}, \Theta \} \]

- \( \{H_i\} \): this proposition means that the hypothesis \( H_i \) is true.
- \( \{-H_i\} \): this proposition means that the hypothesis \( H_i \) is false.
- \( \Theta \): this proposition means the ignorance (i.e. every hypothesis can be true).

Thus, the initialization of the basic belief masses consists in computing, for each criterion \( j \) and for each hypothesis \( H_i \) of \( \Theta \) the basic belief masses \( m_i^{H_j}(\{H_i\}) \), \( m_i^{H_j}(\{-H_i\}) \) and \( m_i^{H_j}(\Theta) \).

To compute all the bmm, belief functions have to be defined. A belief function is a function that returns a float value between 0 and 1 according to the value of a considered criterion for a given hypothesis. Let \( bf \) be a belief functions, \( j \) a criterion and \( H_i \) a decision of \( \Theta \). We note \( V_j^{H_i} \) the value of the criterion \( j \) for the hypothesis \( H_i \).

\[ bf(V_j^{H_i}) : \mathbb{R} \rightarrow [0,1] \]

Examples of belief functions are given in the application section.

**Step 2**

This step consists in combining criteria with each other. We propose to use the conjunctive operator introduced in (Smets and Kennes 1994) to provide a combined bmm synthesizing the knowledge from the different criteria. Let us consider two criteria \( C_1 \) and \( C_2 \). The conjunctive operator is defined as follows:

\[ \forall H_i \in \Theta, \forall P \in \{ \{H_i\}, \{-H_i\}, \Theta \}, m_{\Theta}^{V_j^{H_i}}(P) = \sum_{P' \cap P \neq \emptyset} m_{C_j}^{V_j^{H_i}}(P') \times m_{\Theta}^{V_j^{H_i}}(P') \]

The fusion of criteria can introduce a conflict, e.g. when one criterion assigns a bmm not null for the proposition \( \{H_i\} \) and another criterion assigns a bmm not null for the proposition \( \{-H_i\} \) (i.e. when \( P' \cap \Phi \)). This conflict will be taken into account in the decision.

For example, let \( \{C_1,C_2\} \) be a set of criteria, and \( H_i \) an hypothesis of \( \Theta \). Let the bmm be defined as follows:

\[ m_{C_1}^{V_j^{H_i}}(\{d_1\}) = 0.5, \ m_{C_1}^{V_j^{H_i}}(\{-d_1\}) = 0.3, \ m_{C_2}^{V_j^{H_i}}(\Theta) = 0.2 \]

\[ m_{C_1}^{V_j^{H_i}}(\{d_1\}) = 0.8, \ m_{C_2}^{V_j^{H_i}}(\{-d_1\}) = 0.1, \ m_{C_2}^{V_j^{H_i}}(\Theta) = 0.2 \]

The belief masses resulting after the fusion of \( C_1 \) and \( C_2 \) are equal to:

\[ m_{\Theta}^{V_j^{H_i}}(\{d_1\}) = m_{C_1}^{V_j^{H_i}}(\{d_1\}) \times m_{C_2}^{V_j^{H_i}}(\{d_1\}) = 0.66 \]

\[ m_{\Theta}^{V_j^{H_i}}(\{-d_1\}) = m_{C_1}^{V_j^{H_i}}(\{-d_1\}) \times m_{C_2}^{V_j^{H_i}}(\{-d_1\}) = 0.06 \]

\[ m_{\Theta}^{V_j^{H_i}}(\Theta) = m_{C_1}^{V_j^{H_i}}(\Theta) \times m_{C_2}^{V_j^{H_i}}(\Theta) = 0.04 \]

\[ m_{\Theta}^{V_j^{H_i}}(\{d_1\}) = m_{C_1}^{V_j^{H_i}}(\{d_1\}) \times m_{C_2}^{V_j^{H_i}}(\{d_1\}) = 0.24 \]
This conjunctive operator is commutative and associative. Thus, it is possible to combine the result of a previous fusion with the belief masses of another criterion.

Let $C$ be the criterion set. At the end of this step, for each decision $H_i$ of $\Theta$, we obtain the combined belief masses $m^c_i(\{H_i\})$, $m^c_i(\{-H_i\})$, $m^c_i(\Theta)$ and $m^c_i(\phi)$.

**Step 3**

This step consists in combining hypotheses with each other. This combination is interesting because it allows to take into account in the final ranking, the fact that some criteria reject some hypotheses ($-H_i$).

We propose to use the Dempster operator (Dempster 1967) to compute the belief masses resulting from the combination of two hypotheses $H_i$ and $H_j$:

$$\forall P \in 2^\Theta, m_i^{c,-i}(P) = \frac{1}{1 - m_i^{c,-i}(\phi)} \sum_{P' \ni P} m_i^{c,-i}(P') \times m_i^{c,-i}(P''')$$

The coefficient $\frac{1}{1 - m_i^{c,-i}(\phi)}$ is used to normalize the belief masses obtained. In the case of a total conflict ($m_i^{c,-i}(\phi) = 1$), no decision can be made.

For example, let $\Theta$ be composed of two hypotheses, $H_1$ and $H_2$ ($\Theta = \{H_1, H_2\}$, $\{-H_1\} = \{H_2\}$, $\{-H_2\} = \{H_1\}$). Let the belief masses be defined as follows:

- $m_{i1}(\{H_1\}) = 0.66$, $m_{i1}(\{-H_1\}) = 0.06$, $m_{i1}(\Theta) = 0.04$, $m_{i1}(\phi) = 0.24$.
- $m_{i2}(\{H_2\}) = 0.5$, $m_{i2}(\{-H_2\}) = 0.5$, $m_{i2}(\Theta) = 0$, $m_{i2}(\phi) = 0$

The belief masses resulting from the fusion of $H_1$ and $H_2$ are equal to:

$$m_{i1}(\{H_1\}) = m_{i1}(\{H_1\}) \times m_{i2}(\{H_2\}) \times m_{i1}(\{\Theta\}) + m_{i1}(\{-H_1\}) \times m_{i2}(\{H_2\}) \times m_{i1}(\{\Theta\})$$

$$+ m_{i1}(\{-H_1\}) \times m_{i2}(\{-H_2\}) \times m_{i1}(\{\Theta\}) + m_{i1}(\{H_1\}) \times m_{i2}(\{-H_2\}) \times m_{i1}(\{\Theta\})$$

$$+ m_{i1}(\{H_1\}) \times m_{i2}(\{\Theta\}) \times m_{i1}(\{\Theta\}) + m_{i1}(\{-H_1\}) \times m_{i2}(\{\Theta\}) \times m_{i1}(\{\Theta\}) = 0.27$$

$$m_{i1}(\{H_2\}) = \frac{1}{1 - m_{i1}(\phi)} \times m_{i1}(\{H_1\}) \times m_{i2}(\{-H_2\}) \times m_{i1}(\{\Theta\})$$

$$+ m_{i1}(\{-H_1\}) \times m_{i2}(\{H_2\}) \times m_{i1}(\{\Theta\}) \times m_{i1}(\{H_1\}) \times m_{i1}(\{\Theta\}) \times m_{i1}(\{\Theta\}) = 0.93$$

$$m_{i2}(\{H_1\}) = \frac{1}{1 - m_{i2}(\phi)} \times m_{i1}(\{H_1\}) \times m_{i2}(\{-H_2\}) \times m_{i1}(\{\Theta\})$$

$$+ m_{i1}(\{-H_1\}) \times m_{i2}(\{H_2\}) \times m_{i1}(\{H_1\}) \times m_{i1}(\{\Theta\}) \times m_{i1}(\{\Theta\}) = 0.04$$

$$m_{i2}(\{H_2\}) = \frac{1}{1 - m_{i2}(\phi)} \times m_{i1}(\{H_1\}) \times m_{i2}(\{-H_2\}) \times m_{i1}(\{\Theta\}) \times m_{i1}(\{\Theta\}) \times m_{i1}(\{\Theta\}) = 0.03$$

At the end of this step, a belief mass for each proposition $m_i^c(\{H_1\})$, $m_i^c(\{H_2\})$, $m_i^c(\{H_1, H_2\})$, ..., $m_i^c(\Theta)$ is computed.

**Step 4**

The last step consists in making the decision. We are only interested in the propositions that concern a unique hypothesis (one decision) and not a set of hypotheses. Thus, to evaluate each proposition we propose to use the pignistic probability (Smets 1990).

The pignistic probability of a proposition $A$ is computed by the following formulae:

$$P(A) = \sum_{B \ni A} m(B) \frac{|A|}{|B|}$$

The more a proposition maximizes this probability, the more the corresponding hypothesis is true. Thus, the decision making will be based on this probability: this probability will represent the utility of the decision.

For example, let $\Theta$ be composed of two hypotheses, $H_1$ and $H_2$ and the belief masses of all the propositions be defined as follows:

- $m_{11}^c(\{H_1\}) = 0.93$, $m_{11}^c(\{H_2\}) = 0.04$, $m_{11}^c(\Theta) = 0.03$

The resulting pignistic probabilities are:

$$P(\{H_1\}) = m_{11}^c(\{H_1\}) \times \frac{1}{2} + m_{11}^c(\Theta) \times \frac{1}{2} = 0.945$$

$$P(\{H_2\}) = m_{11}^c(\{H_2\}) \times \frac{1}{2} + m_{11}^c(\Theta) \times \frac{1}{2} = 0.055$$

Thus, $H_1$ has more chances to be true than $H_2$.

**Application of the belief theory to define the agent decision making process**

As presented in the previous section, the belief theory allows to make a decision from a set of possible decisions according to a set of criteria.

In order to use the belief theory to formalize the decision making process of an agent, the modeler has to define several elements:

- A set of criteria that allow to evaluate the different possible decisions.
- For each criterion: a belief function for the hypotheses “this decision is the best one”, “this decision is not the best one”, “ignorance”.

Remark that it is possible to decrease the complexity of the decision making computation by filtering the possible decisions: only decisions that are Pareto-optimal are kept.

For some agents, it will also be possible (or mandatory) to divide the decision making process into several subprocesses. This division can be used to decrease the complexity of the decision process or to use different sets of criteria that will correspond to different steps of reasoning. Indeed, for example, it is possible to divide the decision making process into two steps: the first one consisting of choosing a general objective for the agent (e.g. eating, sleeping) and a second consisting in choosing the best place to carry out this objective.
APPLICATION: CROPPING PLAN DECISION-MAKING.

The MELIA project
The MELIA (MELIA 2011) project aims at developing an agent-based platform for the simulation of the socio-environmental impacts of norms on the water resources. In particular, this project proposes to model the impacts of norms on the behavior of the farmers that are the most important water users in many regions.

Farmer agents and culture choice
The most important behavior of farmers will consist in allocating crops and crop management into their fields. This choice will have a deep impact on the profit of the farmer and on the quantity of water used.

Using the works of (Dury et al. 2010), we defined four criteria that are taken into account during the cropping plan decision-making:

- **Expected profit**: profit that can be expected.
- **Loss at worst**: money that will be lost considering the worst scenario (no plant grown).
- **Workload**: quantity of work.
- **Similarity to last cropping plan**: influence of the last cropping plan chosen.

The next sections describe in details each of these criteria, and in particular the belief functions defined for each of them.

Expected Profit
The first criterion concerns the profit that can be expected from a given cropping plan. The profit takes into account several elements:

- Expected crop production
- Agricultural product price (current price of the market)
- Variable cost
- Workforce price
- Equipment price (tractor, irrigation systems, …)
- Aid (e.g. European Union Aid)

![Figure 1. Belief functions for the profit criterion](image)

The belief functions are shown in Figure 1. These functions depend on the profit made (P). P can be negative or positive. If \( P \) is negative, the belief mass of the proposition “this decision is not the best one” will be higher than 0. If \( P \) is positive, it is the belief mass of the proposition “this decision is the best one” that will be higher than 0. \( P_{\text{max}} \) is the maximal profit that can be made according to the parcel size and the market price. \( D_{\text{max}} \) is the maximal deficit that can be made considering the worst scenario (no plant grown). Two attributes have to be defined for each farmer: \( V_{p1} \) and \( V_{p2} \). \( V_{p1} \) represents the greedy part of the farmer: the higher \( V_{p1} \) is, the more the farmer will try to make benefit at all cost. \( V_{p2} \) is the aversion of the farmer toward the deficit: the higher \( V_{p2} \) is, the more the farmer will tend to avoid deficits.

Loss at worst
This criterion concerns the loss (in terms of money) while considering the worst scenario (no plant grown). Its goal is to assess the risk taken by the farmer. The belief functions are shown in Figure 2. These functions depend on the money that can be lost (\( L \)). If \( L \) is higher than 0, the belief mass for the proposition “this decision is not the best one” will be higher than 0. \( L_{\text{max}} \) is the maximal loss that can be made according to the parcel size and the market price (variable cost). \( V_{l1} \) is computed from \( V_{p2} \) (see expected profit criterion) and from the probability that the worst outcome becomes true (it will depend on the type of culture).

![Figure 2. Belief functions for the loss criterion](image)

Workload
This criterion concerns the workload necessary to carry out the cropping plan. Indeed, the farmers seek to minimize this workload.

The belief functions are shown in Figure 3. These functions depend on the workload value (\( W \)). This value takes into account the quantity of works (in terms of hours of work) necessary to carry out the cropping plan. If \( W \) is higher than 0, the belief mass of the proposition “this decision is not the best one” will be higher than 0. \( W_{\text{max}} \) is the maximal value that can be reach for the workload value. \( V_{w1} \) is the aversion of the farmer toward the work: the higher \( V_{w1} \) is, the more the farmer will tend to avoid working.

![Figure 3. Belief functions for the workload criterion](image)

Similarity to last cropping plan
This criterion concerns influence of the last cropping plan. Indeed, farmers tend to seek for stabilized cropping plan.

The belief functions are shown in Figure 4. These functions depend on the similarity value compare to the last cropping plan (\( S \)).
The belief masse for the proposition “this decision is the best one” will be higher or equal to 0 if the similarity is higher than a given threshold $S_1$. If the similarity is equals or higher than a given threshold $S_2$, the belief masse of this proposition will be equal to $V_{spl}$.

\[ m_{spl}^H(H) \quad m_{spl}^0((-H)) \quad m_{spl}^0(\emptyset) \]

**Figure 4.** Belief functions for the similarity to last cropping plan criterion

$S_1$ and $S_2$ are two thresholds that allow to integrate a fuzzy aspect in the decision. One attribute has to be defined for each farmer: $V_{spl}$. $V_{spl}$ concerns the conservative part of the farmer: the higher $V_{spl}$ is, the more the farmer will tend to make the same culture choices.

**Conclusion**

Our formalism allowed us to simply formalize the raw knowledge provided by the interviews and by the field experts. In particular, it offered us a powerful tool to aggregate the different motivations of the farmers (make profit, avoid loss ...).

A first prototype of the MAELIA model is currently under development with the GAMA simulation platform (Tailandier et al. 2010). This model will take into account more than ten thousands farmers and thus will allow us to test the scalability of our formalism.

**CONCLUSION**

In this paper, we proposed to use the belief theory to formalize the agent decision making process in agent-based simulations. We present an application of this formalism for a simulation dedicated to cropping plan decision-making.

A key issue in the use of our formalism concerns the definition of the belief functions. In some applications, the definition can be based on expert knowledge. In other applications where knowledge is lacking, machine learning techniques can be used to build automatically these functions. In this context, we propose to develop methods to learn them directly through a participatory approach. This approach could be based on the one proposed by (Tailandier and Buard 2009).

**REFERENCES**


**AUTHOR BIOGRAPHY**

**PATRICK TAILLANDIER** graduated in artificial intelligence from the University of Lyon 1 (France) in 2005 and received his PhD degree in 2008 at the University Paris Est (COGIT lab – IGN). After working two years for the MSI research team (IFI – Hanoi, Vietnam) and one year for the SMAC research team (IRIT – Toulouse, France), he was recruited in 2011 as an associate professor by the University of Rouen.
INDUSTRIAL OPTIMIZATION
INVESTIGATING THE PERFECT SEPARATION ON BINARY OUTCOMES CODED IN DECISION MAKING PATTERNS FOR PACE RATING

Peeradaech Suwittayaruk  
Dirk Van Gouberman  
Ghent University  
Technologiepark 903  
B-9052 Ghent-Zwijnaarde, Belgium  
E-mail: peeradaech.suwittayaruk@ugent.be,  
dirk@vangouberman.com

KEYWORDS  
Pace rating, decision making pattern, binary outcomes sensitivity, specificity, a cut off line

ABSTRACT

In order to survive and to be successful in the long run, the key to business is designing, executing and managing operational activities faster, better and more efficiently than the competition. Hence, it is necessary to have good time standards for all activities in manufacturing organizations. Pace rating is an important element to compute standard time. Since the beginning of nineteenth century, pace rating has always been treated as difficult, subjective and even controversial. Although a tool that can simulate and visualize response selection and execution, showing decision making patterns on pace rating has been developed to help Industrial Engineers (IEs) perform pace rating, a decision making pattern has not been studied and analyzed in detail. This paper presents a way to assess accuracy of pace rating by investigating the perfect separation on binary outcomes coded in decision making patterns for pace rating. The application of the interactive dot diagram shows sensitivity, specificity and a cut off line.

INTRODUCTION

Work measurement or time study is one of the oldest tools used to measure and control the amount of time required to perform a specific task. Standard times are important pieces of information needed to carry out many functions in manufacturing organizations such as planning, scheduling, cost efficiency and budget calculations. Direct time study is the most widely spread work measurement technique (Barnes 1980). It is used when manufacturing companies need standard times with accuracy and high confidence. The standard time is composed of two elements as shown in Equation (1).

\[ \text{Standard time} = \text{normal time} + \text{allowances} \]  
\[ \text{Normal time} = \text{observed time} \times (\text{PR}/100) \]  
(1)  
(2)

Allowances are additional times to compensate for personal time, fatigue and delays while normal time can be computed as in Equation (2) above.

Pace rating (PR) is an important element to compute standard time. Pace rating is a kind of performance rating that is defined as the process of rating work’s performance by analyzing the actual task being performed while comparing the observed performance quantitatively with the industrial engineer’s concept of normal performance (Aft 2000). However, the traditional method of determining the rate of working is subjective and is the most challenging aspect of work measurement (Meyers and Stewart 2002). Among the different ways, direct time measurement with pace rating is recognized as being the fastest method because only the operator’s speed of movements is rated.

The basic purpose of determining the work rate is to normalize or adjust the mean observed time for each work element being performed in order to develop the normal time. The quality of the standard time is heavily dependent on the evaluation of the work rate. In this case, rate of working is compared between a worker’s actual performance and a qualified worker performing a task at a standard performance. Pace rating has always been treated as difficult, subjective, challenging and even controversial. (Barnes 1980; Miller and Schmidt 1990; Kanaway 1992; Meyers and stewart 2002; Niebel and Freivalds 2003; Groover 2007).

Pace rating has always been one of the main responsibilities of industrial engineers (IEs). Traditionally, IEs are trained to remember the speed of hand motions at 100% standard pace which corresponds to dealing 52 cards into four equal stacks in 30 seconds. Afterwards, when rating the pace of an actual work, the IEs must compare the observed pace of working with this reference in his or her mind. This visual method has several disadvantages and weak points: it is subjective, not reproducible, has a low confidence and one needs a long time for learning how to rate in a consistent way (Das et al. 1991).

![Figure 1: A pace rating software](image)

145
Because of the many disadvantages, Van Goubergen and Vancauwenberghe (2006) proposed a new idea of pace rating. Based on video technology, a reference video and an actual video of the work method being studied are shown at the same time on a screen. By adjusting the speed of the reference video, IEs or workers without extensive time study knowledge can synchronize the motion patterns in both videos and thus quantify the actual pace of the method under study.

As Van Goubergen and Vancauwenberghe (2006) only presented the basic idea of synchronizing the reference and actual videos, Suwittayarak and Van Goubergen (2011) have developed a new software (Figure 1) based on concepts of discrete-event dynamic systems that can simulate and visualize response selection and execution by showing decision making patterns as displayed in Figure 2.

Figure 2: A decision making pattern

The starting point of the reference video was set off at a minimum tempo difference of 20%. Adjusting the speed of the reference video was programmed so it can be changed with 5% increments and 5% decrements. Three buttons (a + button, a - button and a stop button) are available to make decisions with regard to rating the pace. A series of clicks on the three buttons are the basic idea on which the software is based. The time elapsed before each click (time factor) is recorded in order to compute and simulate the data series of rated paces. The time is plotted along the X-axis and the pace level is indicated on the Y-axis.

As the final response required in the information processing is the tempo of the reference video being equal to the actual video, the basic idea of clicking the mouse during the decision making process depends on the following conditions.
- The + button is activated when tempo of the reference video is less than the actual video (a_t).
- The - button is activated when tempo of the reference video is more than the actual video (a_t).
- The stop button is activated when tempo of the reference video equals to the actual video (a_t).

Paces sequentially evaluated with this visual and reasonable procedure can serve as a model and simulate how pace rating is processed in the human brain of the IE, as illustrated in Figure 2. In addition, documentation of a decision making pattern can serve as an aid to communicate between management and union.

Until now, previous work has only focused on using the final response as the rated pace (Suwittayarak and Van Goubergen 2011). Accuracy of rating by studying decision making patterns obtained from participants still needs to be investigated.

This paper presents a way to assess accuracy of pace rating by utilizing decision making patterns. Our knowledge of the assessment of rating accuracy is largely based on sensitivity, specificity and a cut off line calculated by investigating the perfect separation on binary outcomes that are coded in decision making patterns.

HOW TO INVESTIGATE THE PERFECT SEPARATION ON BINARY OUTCOMES

To investigate the perfect separation on binary outcomes, initially, the decision making patterns are analyzed in this research by using the Receiver Operating Characteristic (ROC) tool. ROC is a standard tool that is very useful for assessing and comparing the predictive accuracy of decision making under uncertain conditions, typically used on signal detection studies (Green and Swets 1988; Macmillan and Creelman 2005), psychophysics (Gescheider 1997; Krzanzowski and Hand 2009; Kingdom and Prins 2010), and medical research (Altman 1991; Zwie and Campbell 1993). ROC considers decision making resulting in two responses: signal (yes) and no signal (no).

Figure 3: The distribution of the two populations

Figure 3 displays the basic idea of overlapping of the two populations, showing four possible outcomes:
- Hit rate: saying there is a signal when the signal is present.
- False alarm: saying there is a signal when no signal is present.
- Correction rejection rate: saying there is no signal when no signal is present.
- Miss: saying there is no signal when the signal is present.

The main objective to use this method is to compute the perfect separation between the two groups by maximizing two values: sensitivity and specificity. In order to
investigate the perfect separation on two populations, more correct yes and no responses (more hit rate and more correction rejection rate) and less errors (less false alarm and miss) are the basic idea on which the method is based. Sensitivity is the probability of yes responses that are correctly evaluated by the rater, also called the “hit rate” marking the true yes response rate (P[yes|yes]). Specificity is defined as the probability of no responses that are correctly assessed, also named the “correct rejection rate” pointing out the true no response rate (P[no|no]). Sensitivity and specificity are expressed as percentages.

Figure 4: Coding binary outcomes on decision making patterns

In order to understand the basic idea of ROC, Figure 4 simulates ten identical decision making patterns: the real pace is 100% but 105% is a threshold of a faster point. In this case, the threshold is coded as the yes (1) response. On the other hand, the rest are coded as a series of no (0) responses. This data was analyzed with the DeLong’s method using MedCalc V.11.2 software. The obtained ROC curve is illustrated in Figure 5 where sensitivity (the true yes response rate) is plotted in function of 100-specificity (the false yes response rate). The ROC plot passes through the upper left corner with 100% sensitivity, 100% specificity and a criterion > 100 %. Generally speaking, a ROC plot should always be close to the upper left corner, clearly indicating a high accuracy of the rating. In practice, the false rates threshold value may be set to 10% which results in the requirement that both sensitivity and specificity should be at least 90%.

Figure 5: The ROC curve showing 100% sensitivity, 100% specificity and a criterion > 100 %

To obtain more and clear information, a second type of chart, namely an interactive dot diagram, is used to simulate the data of the yes (1) and no (0) responses as dots (Figure 6 is an example based on the same data set as Figure 4). The binary outcomes coded (the yes (1) and no (0) responses) is plotted along the X-axis and the pace level is indicated on the Y-axis. Originally, a horizontal line is drawn as a cut off line corresponding to the best separation between the two groups based on minimum false no response and false yes response values (here at 100%).

The sensitivity and specificity as shown on the ROC curve are also displayed in the interactive dot diagram. On this diagram we can see the perfect separation of the yes responses from the no responses, without any overlap for the two populations. In this example, the real pace is 100%, and 105% is the threshold of the faster point. The cut off line at 100% indicates the resulting value of the rated pace with 100% sensitivity and 100% specificity. Contrary to the simple data set we just used to introduce the diagrams. In this study, however, we hypothesized that the perfect separation between two groups is impossible to occur in practice.

Figure 6: The simulation of the yes (1) and no (0) responses showing the perfect separation between two groups

CASE APPLICATIONS

In order to test the method to investigate the perfect separation on binary outcomes, two experiments were performed to simulate response selection and execution showing decision making patterns for pace rating based on comparing two videos. A reference video, analyzed and normalized with MTM-1 at 100% pace using Ulead VideoStudio version 9, was chosen to be used in the experiments as the reference video, displayed in Figure 7 on the right side. The video contains motions such as reaching, (easy) grasping, moving, positioning, and releasing movements with the same distance. MTM-1 is a predetermined time standards technique that provides the highest accuracy compared to other similar systems. It gives a normal time based on predetermined time values at pace 100% for fundamental motion such as: reaching, moving, turning, apply pressure, grasping, positioning, releasing, disengaging, eye movements, and body movement (Karger and Bayha 1987).

Two actual videos were chosen from two different sources to conduct the experiments:
1. The actual video 1 showing an easy motion pattern was analyzed and developed with MTM-1 by the authors using Ulead VideoStudio version 9 as shown on the left side Figure 7(a). This video consists of motions over the same distance as the reference video.

2. The actual video 2, showing hand movements with complex motion patterns as displayed in the left side of Figure 7(b) was rated at 100% pace using the traditional way and was selected from the TMI pace rating video collection (Wattmough 1975).

(a) Set 1 showing an easy motion pattern as an actual video

(b) Set 2 showing a complex motion pattern as an actual video

Figure 7: Case applications

For conducting the experiments, twenty graduate Industrial Engineering and Operations Research students at Ghent University, Belgium were selected as participants. They had background in pace rating as they had taken a graduate work measurement and method engineering course.

The following instructions were given to the participants for conducting the experiment: the reference and actual videos are shown at the same time without giving any quantitative information on the pace. The participants are asked to compare both videos with regard to pace of working. As a starting point, the reference video is set off at a lower pace (20%). When the pace is perceived by the participants as being lower, the + button (a3) needs to be clicked. The speed of the reference video will now increase. The same question is asked repeatedly until the pace is judged as being higher. Finally, the stop button (a3) needs to be pushed in order to end the experiment. Forty decision making patterns (twenty per each set) were collected as data. Figure 8 illustrates an example of a participant. The time is plotted along the X-axis and the rated pace level is indicated on the Y-axis.

In this case, a decision making pattern represents only one yes response and a series of no responses. The yes response was coded 1 while the no responses were coded 0 as shown in Figure 8. These binary outcomes are used to generate the sensitivity and specificity which are the two outcomes of ROC.

![Rated value](image)

Figure 9 The interactive dot diagram of set 1

![Rated value](image)

Figure 10 The interactive dot diagram of set 2

The data with regard to the decision making patterns from the experiments were computed using MedCalc V.11.2 with the DeLong’s method (DeLong et al. 1988; Schoonjans 2010). All data points were considered: the yes response is the stop button activated while the no responses are the + button activated as exhibited in Figure 8. The data was analyzed using the interactive dot diagram which is used to
discriminate yes (1) responses from no (0) responses. Having computed the data, we obtained both high sensitivity and specificity values as can be seen in Figure 9 and 10.

The two corresponding test characteristics (sensitivity and specificity) are indicated at the right side of the diagrams. The two figures also display the cut off line of each set. Interestingly, both of them almost reached the real pace (100%). Clearly, sensitivity and specificity obtained from set 1 are higher than set 2. Our evidence would seem to imply that when rating with an actual video showing a complex motion pattern is more difficult than rating with an easy motion pattern: the cut off line of set 1 is closer to the real pace (100 %) compared to set 2.

CONCLUSIONS AND FURTHER RESEARCH

The purpose of this paper is to present a way to assess accuracy of pace rating by utilizing decision making patterns. Decision making patterns generated by multiple participants can be used to compute the cut off line representing the rated pace based on the interactive dot diagram. The most important requirement is that both sensitivity and specificity values need to be high enough. We believe that our method could be used in discriminating yes(1) responses from no(0) responses to obtain sensitivity, specificity and the cut off line. These results widen our knowledge of investigating the perfect separation on binary outcomes coded in decision making patterns for pace rating.

As hypothesized, clearly, the perfect separation between two groups with 100% sensitivity and 100% specificity is impossible to occur in practice. In addition, the study indicates that apparently rating on clearly different motion patterns has a significant impact on the sensitivity and specificity values as well as the cut off lines. And finally, until now we obtained sensitivity, specificity values and the cut off lines based on experiments with only two different motion patterns. Important research questions we are still interested in are related to determining a difficulty index for rating based on motion pattern compatibility between the reference and actual videos. Additional research and experiments are needed to assess difficulty indexes when three or more different motion patterns are involved, using the interactive dot diagrams.

REFERENCES


BIOGRAPHIES

PEERADAECH SUWITTAYARUK is a PhD student in Industrial Engineering and Operations Research at the department of Industrial Management, Ghent University(Belgium). His research interests lie in work measurement, motion study, ergonomics and lean manufacturing. In his PhD he is researching on improving work measurement practice for method improvement and process design using video analysis.

DIRK VAN GOUBERGEN is a professor at the department of Industrial Management at Ghent University. He is an expert in the area of work measurement and method engineering, productivity improvement and lean manufacturing. Additionally, he has more than fifteen years of international experience as a trainer/consultant in the area of lean manufacturing within different industrials throughout Europe. He has a M.S. degree in Mechanical Engineering from the Royal Military Academy (Brussels-Belgium), a M.S. in Industrial Management from Ghent University and a PhD in Industrial Engineering from Ghent University. He is a senior member of the Institute of Industrial Engineering(IIE).
MODELING AND SIMULATING A TEXTILE PRODUCTION SYSTEM

Sérgio Oliveira, José A. Oliveira, Luis S. Dias, Guilherme Pereira
Algoritmi Research Centre, Universidade do Minho
Campus de Gualtar
4710-057 Braga – P, Portugal
E-mail: {zan@lsd@gui@dps.uminho.pt}

KEYWORDS
Modeling, Simulation, Scheduling, Heuristics, Optimization, Arena.

ABSTRACT

This paper presents a study for a production scheduling problem in a textile company, specifically in the weaving preparation area. Basically, the processing orders can be considered as sequential working steps through three operations (charging - weaving - discharging), and the goal is to minimize time variation and to avoid delays. The machine utilization should be as higher as possible due to short delivering deadlines. The production unit has got 4 of these weaving machines functioning at the same time. Four dispatching rules were tested in order to find the best solution. The optimization procedure highlighted some interesting issues that are discussed in this paper.

INTRODUCTION

This work focused mainly on the production programming problem in the weaving preparation area of a textile company. The production function became a main concern faced by managers once new problems arise due to the huge number of products and the uncertainty on demand. These two factors reduce the company productivity and competitiveness. Nowadays, a large number of companies are changing their production systems to this new reality. There are several tools in the market to support the decision to plan and control the production. Programming the production is an important concern to the companies because they want to maximize the utilization of their resources, increasing productivity without compromising flexibility. However, this is a well known hard problem with a high level of complexity, where a mathematical solution could not be obtained in a short time period.

In this paper we present a study to solve this production scheduling problem. We present a decision support system based on heuristic rules and a simulation model. The main objective of the production schedule is to define the sequence of the jobs that involve three operations (loading – weaving – unloading) in order to minimize time variation and to avoid delays. The machine utilization should be as higher as possible due to short delivering deadlines. The production unit has got 4 of these weaving machines, functioning as parallel machines.

A literature review has been performed in order to identify related heuristics to schedule the orders. The selected and applied heuristics were: SPT (Shortest Processing Time), NOB (Number Of Bobbins), EDD (Earliest Due Date) and FAP (Family Articles Processing). Based on those dispatching rules, manufacturing orders were sorted and performance evaluated accordingly.

Results were evaluated in a simulation model using Arena software, in order to make the acceptance or rejection of heuristics considering the numerical results and the model animation helped model validation by the company decision makers.

THE PROBLEM

The main problem is related with a process composed by three operations: load an order, make a fabric, unload the order. The shop floor has four parallel (identical) machines. In each machine there are two devices to load an order. Loading an order means to put about 740 bobbins with the cotton thread in the machine. Figure 1 shows a view of the warehouse, where it is possible to see several pallets with bobbins of cotton thread.

Figure 1: Bobbins of cotton thread

Figure 2 shows the textile machine to produce the fabric. The machine has two device loaders - this allows to use one of them with one order, while it is possible to unload the previous order and also to prepare (to load) the next order.
Figure 2: Textile machine

Figure 3 shows a representation for the machine and the two device loaders which will be used in the simulation package to animate the simulation.

Figure 3: Machine and devices loaders

Figure 4 represents the machine device loader with several bobbins with several colors to produce one pattern.

Figure 4: Device loader with different colours

Figure 5 presents the layout of the shop floor where four machines are available, the respective device loaders, a warehouse of raw material, and two workers per machine. In each machine there are a skilled hand to operate the machine and an auxiliary worker which load and unload the machine with the orders. Figure 5 is used in the simulation package to animate the simulation.

Figure 5: The layout

There are six different states to be considered in this process for the machine and its two devices. The machine itself could be 1) waiting for the next order; 2) involved in a setup procedure; 3) operating to produce the fabric. On the other hand, the device loaders could be 4) loading the set of bobbins for the next order; 5) busy with loaded order; 6) unloading a finished order. Figure 6 shows a Gantt Chart to present the several states of the machine and the devices.

Figure 6: Gantt chart

In this small example we present the sequence of four orders (OF i) and also the previous one (OF n-1). It is possible to verify that when OF 1 is finished the machine must be idle until loading of OF 2 on Device 2 is completed. In the meantime, to produce OF 3 the machine does not need to wait for the conclusion of loading order in the Device 1. The operating time of each order is variable and depends on the quantity (in meters) of fabric to be produced. Also, the time to unload an order and to load the next order in a device depends on the number of bobbins to be changed in the device. If a consecutive pair of orders prepared in the same device share the same color (bobbin) it is not necessary to remove that bobbin from the device to replace it again in the device. This property allows to save time on both unloading and loading operations in the device.

This real problem could be considered as a generalization of the well known problem of scheduling operations in a flexible manufacturing system where it is necessary to change the tools of a machine. The particular situation of one machine could be modeled as a traveling salesman problem, where the sequence of orders could minimize the time to change the tools on the machine.

MODELING THE REAL PROBLEM

This real problem could be modeled as a deterministic combinatorial optimization problem, in particularly as a variant of the multiple traveling salesman problem (mTSP), which consists of determining a set of routes for m salesmen who all start from and turn back to a home city.

The mTSP can be considered as a relaxation of the vehicle routing problem (VRP), with the capacity restrictions removed (Bektas 2006). This means that all the formulations and solution approaches proposed for the VRP are also valid and applicable to the mTSP, by assigning sufficiently large capacities to the salesmen (vehicles). The mTSP is a NP-complete problem (Husban 1989). To find an optimal solution to the mTSP using exhaustive search is only possible for a very small number of nodes. An alternative to find solutions for this kind of problems is using heuristics, such as evolutionary algorithms (Fogel 1990), simulated
annealing (Song et al. 2003), tabu search (Ryan et al. 1998), ant systems (Pan and Wang 2006), etc. The company is facing new difficulties related with the huge number of different (type of) products that are produced and the uncertainty on the demand. This known variability made the authors decide to study the real problem using a simulation model. Indeed, the system under investigation is complex and it is not possible to find analytical solutions. Nevertheless a deterministic model is not enough to represent the real problem, due to demand fluctuations, and design changes must be considered. Taking into account the complex stochastic characteristics of the system, simulation is the most suitable tool to predict system behavior and is a powerful operations research technique to use. In particular, discrete event simulation has proved to be a useful tool for evaluating the performance of such systems.

The current manufacturing system is analyzed by a simulation model emphasizing the bottlenecks and the poorly utilization of the machines. The main idea is to develop a simulation optimization based decision support system where simulation outputs would validate analytical results, incorporating variability in the process and allowing important sensitivity analysis of the solutions found. However, a simple evaluation of performance is often insufficient and a more exploratory process may be needed in the form of simulation optimization. As stated by Ólafsson and Kim (2002) simulation optimization is the process of finding the best values of some decision variables for a system where the performance is evaluated based on the output of a simulation model of this system. However, the major drawback of simulation for practical applications is that it is computationally time consuming. There has been a great deal of work on simulation optimization in the research literature, and more recently optimization routines has been incorporated into several commercial simulation packages. Fu (2002) presents an exhaustive tutorial that summarizes the existent approaches and provides a discussion contrasting these approaches with the algorithms implemented in commercial software.

**SIMULATION TOOL SELECTION**

According to Dias et al. (2011) most of scientific works related to tools comparison/reviews only analyze a small set of tools and usually evaluating several parameters separately avoiding to make a final judgment due to the subjective nature of such task.

An extensive review about simulation software could be found in Dias et al. (2011):

> “In the Industrial Engineering Magazine (1993/July) there is a list of 45 commercial simulation software products. The sixth biannual edition of simulation software compiled by James J. Swain in 2003 identifies about 60 commercial simulation products, 55 in 2005 and 48 in 2009 (Swain, 1991-2009). The annual 2004 SCS edition – (M&S Resource Directory) lists 60 simulation products. In the (Simulation Education Homepage - Simulation tools list by William Yureck) there were more than 200 simulation products, incl. non-commercial tools.”

The study produced by Dias et al. (2011) started with Swain’s list, removing non discrete event simulation environments, and adding some tools found in more than one list sources. Dias et al. (2011) support that the three most popular simulation packages are Arena, Simul8 and Witness. Figure 7 presents the ranking presented by Dias et al. (2011).

The authors believe that the Top 10 “popular” simulation commercial tools are included in that list (of 19). As well as it is most probable that this list includes the top 10 “most used” and “best” contemporary simulation tools.

![Figure 7: Popularity ranking (Dias et al. 2011)](image)

The chart in Figure 7 can help to visualize the strengths and weaknesses of each tool, in a comparative analysis.

**THE MODEL**

Figure 7, showing an interesting simulation software popularity ranking moved us to choose Arena Software from Rockwell Automation, Inc to construct a model to this real problem. Arena is built over the SIMAN simulation language. Arena is based in the process simplification through discrete events. After creating a simulation model graphically using base Arena models and the primitive building blocks and establishing links between blocks (to represent the process logic), Arena automatically generates the underlying SIMAN model used to perform simulation runs. In Arena package it is not necessary to develop code, since the modeling process is graphical. The Arena modeling system is a flexible and powerful tool that allows analysts to create animated simulation models that accurately represent virtually any system. The Arena Professional Edition also includes the functionality of OptQuest, for optimizing systems.

To implement this model, it was necessary to collect real data from the company, particularly to define operations processing times. To aid this task, it was collected from the
real process registries using MS Access equivalent to one month and later the data was compiled in a spreadsheet. Using ARENA INPUT ANALYZER functionality the elementary processing times were fitted to statistical distributions, based on historical data. Figure 8 shows expressions of modeled tasks, detailing one of them (Charging) as output of Arena.

<table>
<thead>
<tr>
<th>Task</th>
<th>Expression</th>
<th>% Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search yarn</td>
<td>1.5 + LOGN (3.16, 2.24)</td>
<td>5.50%</td>
</tr>
<tr>
<td>Charging</td>
<td>TRIA(6.5,11.20.5)</td>
<td>1.20%</td>
</tr>
<tr>
<td>Splice</td>
<td>6.5 + 13 * BETA (1.06, 1.3)</td>
<td>0.57%</td>
</tr>
<tr>
<td>Mince Comb</td>
<td>0.5 + WEIB (1.63,1.75)</td>
<td>0.029%</td>
</tr>
<tr>
<td>Weaving</td>
<td>1.5 + GAMM (0.582,4.14)</td>
<td>0.40%</td>
</tr>
<tr>
<td>Pass</td>
<td>8.5 + 12 * BETA (2.99,1.92)</td>
<td>0.70%</td>
</tr>
<tr>
<td>Discharging</td>
<td>1.5 + 9 * BETA (1.88,2.09)</td>
<td>0.64%</td>
</tr>
</tbody>
</table>

![Figure 8: Arena Input Analyzer usage](image)

It was developed the communication between Arena and the enterprise resource planning (ERP) system to load the existent orders. To import data from the spreadsheet (MS Excel) to the Arena, we developed a Visual Basic Application routine, presented at Table 1.

**Table 1: Visual Basic Application routine**

```vba
Private Sub ModelLogic_RunBeginSimulation()
Dim m As Model
Set m = ThisDocument.Model
Dim FileToOpen As String
Dim ArenaDir As String
Set XL = GetObject pervous Excel Application"
ArenaDir = Mid(m.FullName, 1, Len(m.FullName) - Len(m.Name))
FileToOpen = ArenaDir & "testarena.xlsx"
XL.Workbooks.Open FileToOpen
End Sub
```

In Figure 9 there is a “cloud” of the Arena Model (network of building blocks), with the loading area (from ERP system), the splitting area (to the four machines) based on the selected heuristic, and control logic for each machine.

![Figure 9: Arena Model](image)

The simulation model developed in Arena makes use of the tool animation feature to better discuss and validate results with decision makers. Figure 10 shows one screenshot of the animation area, constructed accordingly to the real layout as shown in Figure 5.

![Figure 10: Arena Model Animation Screenshot](image)

**THE EXPERIMENTAL RESULTS**

To study the several important factors of performance four orders dispatching rules were selected: SPT (Shortest Processing Time), NOB (Number Of Bobbins), EDD (Earliest Due Date) and FAP (Family Articles Processing). The main outputs considered to evaluate the performance of the system are machine utilization, linear metre per hour of fabric, woven fabric per hour, man power utilization and production time.

Table 2 presents production outputs retrieved from computational experiments preformed with simulation software which allow us to compare the selected dispatching rules.

<table>
<thead>
<tr>
<th>Productivity (metre/hour)</th>
<th>Real Prod</th>
<th>Real Sim</th>
<th>FAP</th>
<th>SPT</th>
<th>EDD</th>
<th>NOB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>150</td>
<td>150</td>
<td>210</td>
<td>192</td>
<td>189</td>
<td>186</td>
</tr>
<tr>
<td>Productivity (woven fabric/hour)</td>
<td>0.24</td>
<td>0.24</td>
<td>0.34</td>
<td>0.31</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>Machine Occupation</td>
<td>52%</td>
<td>46.5%</td>
<td>63.3%</td>
<td>59.75%</td>
<td>58.75%</td>
<td>58.25%</td>
</tr>
<tr>
<td>Manpower Occupation</td>
<td>-</td>
<td>-</td>
<td>75%</td>
<td>93.25%</td>
<td>92.5%</td>
<td>91.25%</td>
</tr>
<tr>
<td>Production time (hours)</td>
<td>7380</td>
<td>7380</td>
<td>5276</td>
<td>5768</td>
<td>5864</td>
<td>5948</td>
</tr>
</tbody>
</table>

The results presented at Table 2 shows a 5.5% less utilization of the machine in the simulated real production. Besides this short difference validates the model, it could be related with small operations that exit in the real process and they are not included. The FAP dispatch rule presents a machine utilization rate of about 63%. This simulation result foresees an increase around 11% comparing with the achieved value in the real situation, and 16% better than the simulated real production. With this option the productivity increase 60 metres per hour. The man power utilization is also increased significantly.
comparing with others dispatching rules. Although there is an important disadvantage of this rule – it does not take into account the due dates of the orders, and can therefore generate delays to deliver the orders. The SPT rule achieved an improvement of 7% in terms of utilization rate and increases productivity in 42 meters per hour. EDD and NOB have similar results and increase productivity in 36 meters per hour. Globally the FAP rule seems to be the best option to be implemented but still needs a plan that consider similar products to be jointly sequenced.

CONCLUSIONS AND FURTHER WORK

This textile company must be able to adapt to its customers’ ever-changing needs and improve the quality of its products in order to survive. It is important that the company responds quickly to rapid changes in demand fluctuations, and design changes. These needs have forced to put emphasis on automated systems to improve productivity. Many systems in areas such as manufacturing, supply chain management, financial management, are too complex to be modelled analytically. Discrete event simulation has long been a useful tool for evaluating the performance of such systems. The goal of this paper was to provide a formal model to one important problem in production management, specifically the one related to constrained resources utilization and its implementation. The more suitable orders dispatching rule is Family Articles Processing since benefits on lower times to load and unload orders increased the productivity about 11% allowing the production of more 60 meters of fabric per hour. Finally, it can be stated that the state of the art of simulation adopted in this paper as a DSS for modelling real production system can prompt management to compare the existing system and the proposed new process, and to find near-optimum values of the decision variables. It is shown that the proposed DSS is used effectively to improve the production rate, and results show a substantial decrease on the overall production time. This research also demonstrates how simulation modeling can be used to design and optimize real production systems. Shortly, we intend to explore the optimization package OptQuest and to compare with more dispatching rules to sequence the costumers orders. In a further research we intend to explore more heuristics from combinatorial (discrete) optimization to be employed in the simulation models.

REFERENCES


BIOGRAPHIES

SÉRGIO OLIVEIRA was born 1980 in Vale S. Cosme, Portugal. He holds a MSc in Industrial Engineering from University of Minho. His main interests are Industrial System Planning and Programming, Simulation and Lean Manufacturing.

JOSÉ A. OLIVEIRA was born 1966 in Matosinhos, Portugal. He studied Mechanical Engineering at the University of Porto, Portugal. He graduated with a Ph.D. in Production and Systems Engineering at University of Minho, Portugal. His main research interests are Optimization with Heuristic Methods in Systems Engineering.

LUÍS DIAS was born in 1970 in Vila Nova de Foz Côa, Portugal. He graduated in Computer Science and Systems Engineering at the University of Minho, Portugal. He holds an MSc degree in Informatics Engineering and a PhD degree in Production and Systems Engineering from the University of Minho, Portugal. His main research interests are Simulation, Operational Research and Systems Visual Modeling.

GUILHERME PEREIRA was born 1961 in Porto, Portugal. He graduated in Industrial Engineering and Management at the University of Minho, Portugal. He holds an MSc degree in Operational Research and a PhD degree in Manufacturing and Mechanical Engineering from the University of Birmingham, UK. His main research interests are Operational Research and Simulation.
A SIMULATION OF ALLOCATION OF THE POWER RESERVE IN THE POWER SYSTEM AND MICROGRID CONTAINING RENEWABLES

Eugeniusz M. Sroczan
State Higher Vocational School in Gniezno
ul. S. Wyszyńskiego 38
62-200 Gniezno, Poland
E-mail: eugeniusz.sroczan@put.poznan.pl

KEYWORDS
Renewable energy, interactive program, decision support system, interactive simulation, optimisation.

ABSTRACT

In order to solve the problem of allocation of the power system reserve the dedicated simulator was developed. It has been working in interactive mode and – as a result – the queue of load of committing power plant is calculated. As input data for simulation, except demanded power forecast, the characteristics of cost increasing versus power of each plant and statistical data from a given wind farm are applied. The developed simulator enable to consider and assess the influence of the generating structure in the microgrid for the power system especially in the phase of planning the development as well as during the exploitation.

INTRODUCTION

Electricity generated by photovoltaic plants, wind farms and hydro-plants, installed in local microgrid (MD) system and supplied with renewable sources of energy, depends on sun irradiation or wind speed as well as water inflow to hydro-plant. Therefore, in the case of insufficient capability of renewable sources, the energy should be supported by thermal power plants committing in the power system (PS). That means that a microgrid is importing the energy and takes effect in the increasing cost of energy generated in the PS. In other case, the energy is exported or stored in PS storage hydroelectric plant. Inside the microgrid, demanded electricity generation management (EGM) and demand side management (DSM) should be applied.

That manner of electricity generation in power system requires a system reserve of power. It should be allocated in the PS power plants characterized with some parameters affecting the minimal set-up time, short time of full load and capability to cover the load increase with a reasonably low cost.

The work of power plants which generate electricity and are connected to microgrid is specific due to naturally limited resources of power and energy which results from renewable kinds of primary energy (PES) denoted to given plant (Hunt 1982, Kacejko 2009, Karëk 2010, Sroczan 2008). Microgrid generation of electricity is characterized as dispersed (or decentralized) generation (DG) due to small rated power of the plants and the form of realisation of diversity the primary energy processed to electricity. In general, DG, as a dispersed structure, has the task to convert the primary energy, in the place where it exists, with use of different technologies of conversion into electricity demanded by local energy market (LEM). From technical point of view, microgrid – LEM is a part of power system and consists of a set of power plants: energy sources, grid and consumers.

The discussed LEM consist of the plants converting renewable: hydropower plants (HPP), photovoltaic power plants (PVPP) and wind power plants (WPP) committing, standard thermal power plants (TPP) as well as combined heat and power plants (C-HPP) fired with gas, coal or biomass.

THE PURPOSE AND THE SCOPE OF THE SIMULATION

The results of the simulation supports the decision concerning the strategy of the power balance. The power supplies the microgrid with regard to plant operation, at optimal cost, in order to meet the power system (PS) balance between demanded load \( P_d \) and generated power \( P_g \) in each \( t \) – moment of time (Sroczan 2008, Schwepe 1998). The flow of information and energy, necessary for simulation, is shown in figure 1. Similar approach to model the utility (for utility computing purposes) is presented in (Heckman at al. 2009).

![Figure 1: The area of simulated processes – structure and flows of information and energy in power system](image-url)
The developed simulator is a tool supporting the decision of optimal operation and balance of the power and energy production on LEM. Calculations are lead with regard to consumers’ demand, as well as the technical, economic, ecological, hydraulic (for hydroelectric plants) and power system boundaries and attributes (fig. 2). Generating the electricity on demand may cause significant increase in end-user energy prices. In addition, this mode is being strongly curtailed by weather conditions and consumers’ behaviour or PS phenomena (Bober and Kaproń 2009, Kacejko and Pijarski 2009, Karki at al. 2010).

A demand side management system (DSM) operates as an interface between the microgrid and the outside energy network. The interaction between energy end-users, supplied from microgrid, and state electricity system PS (fig. 1, fig. 2) may be managed by an energy market with use of advanced metering management (AMM). The main idea presented by (Billewicz 2009). This kind of operation requires the determined degree of the automation of the arrangement of the receiving installation. The effect of DSM action is smoothing with the load curve through its reduction or the increase, according to kind of signals, sent in by distributor to the supplier and customer.

![Figure 2: The structure of limitations and relations tree describing the influence of changes the load in microgrid affecting the attributes of power stations in power system](image)

The cost of electricity in the power system defined for the energy end-user, depends on the level of demanded power and the structure of committing power plants connected to local power network and, presented case, for the microgrid as the subsystem of PS.

**STRUCTURE OF SIMULATOR**

The flow diagram of the part developed structure of the simulator is shown in figure 3. The structure of the developed simulator consists of:

- preprocessor of the input data enabling the calculation of the change of demanded and generated power in PS;
- procedures for the assortment of the set of committing power plants with regard to actual energetic policy and wide range of attributes of committing power plants;
- procedures applying the classical attempt to optimize the cost of electricity with respect to the renewable sources (fig. 3).

When the power of given renewable source depends on weather conditions, the programme of economic load dispatch must reschedule the load queue of given plants. The lack of power in that source node is compensated by additional flow from adjacent nodes or PS (fig. 1). Therefore, the set of attributes $G \{H, E, C, T, S\}$ (fig. 2) of committing power plants and PS is embedded in the process of simulation. It is being activated, for the purposes of economic dispatch, above all at the stage of the modification of characteristics of committing units.

As it is shown in the fig. 2., the change of demanded power affects the change of load of committing units. In some plants (Kaproń and Wydra 2008) the efficiency of energy conversion is decreased, therefore the cost of electricity is raising. The optimal cost of energy is calculated with the use of queuing the load of each power plant.

**COST SIMULATION OF THE GENERATION OF ELECTRICITY**

It is assumed that the structure of power plants is fixed and fulfils the technical requirements as well as contains the renewable energy sources. The cost of electricity depends on the load profile of energy customers and available options of converting the renewable energy into electricity.

The simulated cost of electricity $C_{em}$ at time $t$ is defined in the way described in (Sroczan 2010) and include:
\[ C_{PS} = C_{og} + C_{vg} + C_{TC} \]

where: \( C_{og} \) – fixed cost of generation, \( C_{vg} \) – increase in the cost of the generation as a result of load change, \( C_{TC} \) – cost of transmissions losses and system services including rebuilding of the system power reserve.

The cost is optimal if in each time \( t \), the following relationship is fulfilled (Sroczen 2008):

\[ C_t = \min \left\{ \sum_{i} C_i (P_{gi}) \right\} \]

and

\[ \sum_{i} P_{gi} = P_d \]

\[ P_{\text{min}} \leq P_{gi} \leq P_{\text{max}} \quad i \in \omega \]

where: \( P_{gi} \) – generated, \( P_{\text{min}}, P_{\text{max}} \) – minimal and maximal power for \( \omega \) committing power plant, respectively, \( P_d \) – demanded power \( t \) – time range.

The value of power generated in hydropower plant is defined as:

\[ P_{HPP} = 9.81 \cdot Q \cdot H_1 \cdot \eta(Q, H) \] [kW]

where: \( Q \) [m³/s] – water flow, \( H_1 \) [m] – head of water, \( \eta(Q, H) \) – coefficient of turbine efficiency.

Generated value of power of wind power plant depends on the wind speed as follows:

\[ P_{WTP} = \eta_T \cdot C_t \cdot C_p \cdot (v_t \cdot v_i) \] [kW]

where:

\[ C_t = \frac{1}{2} \cdot \rho \cdot \pi \cdot D^2 \cdot 10^{-3} \]

\( \eta_T \) – total efficiency of generator and converter, \( C_f \) [\%] – efficiency coefficient, \( \rho \) – air mass [kg/m³], \( D \) [m] – rotor diameter, \( v_t \) [m/s] – instantaneous wind speed.

For purposes of load dispatch the corrected and guaranteed characteristics of photovoltaic plant are calculated as:

\[ P_{PV} = f[I_{	ext{irr}}(dt, cl, \vartheta)] \] [kW]

where: \( I_{	ext{irr}} \) – irradiation of PV panels, coefficients: \( dt \) – daytime, \( cl \) – cloudy, \( \vartheta \) – temperature of panels.

The equations (1) and (2) are used to solve the problem of optimal scheduling the load of power plants described by characteristics (5), (6) and (8).

**RESULTS OF SIMULATION**

The algorithm of calculations is based on the real costs of energy in given conditions with regard to demand and the updated characteristics of the renewable energy sources.

---

**CONCLUSIONS**

Simulation of the cost of generation in the microgrid (MG), treated as a structure containing renewable sources of primary energy and at the same time subset of PS in order to minimize the costs of energy with respect to renewable energy ratio was realized for model of committing power plants.

The process of the optimisation of the work of committing power plants considers the time period \( T \) in which all of the discussed units change the value of generated power with accordance to power demanded by end-users. The applying of the worked out simulator is defined as optimisation of generation cost with respect to the PES and PS constraints as well as supporting the AMM in the optimal flattening of the load of the PS.

**REFERENCES**


---

![Figure 4: Changeability of load a wind power station](image-url)
ENGINEERING
SIMULATION
SIMULATION AND EXPERIMENTAL TECHNIQUES FOR THE ANALYSIS OF LOCALISED-INCREMENTAL FORGING OPERATIONS

Ana M. Camacho*, Marta M. Marin, Claudio Bernal and Miguel A Sebastián
Department of Manufacturing Engineering
National Distance University of Spain (UNED)
E 28040, Madrid,
Spain
E-mail*: amcamacho@ind.uned.es

KEYWORDS
FEM simulation, Explicit and Implicit model, Experimental setup, Localised-Incremental Forging

ABSTRACT
Simulation of Localised-Incremental Forging operations is often difficult, especially when several strokes are necessary to obtain the final shape of the part. In this paper, two Finite Element methodologies are applied for the simulation and analysis of a simple localised operation: implicit and explicit numerical methodology by using Abaqus/Standard and Abaqus/Explicit solvers, respectively. The material considered is paraffin wax in order to compare the results experimentally. The most suitable mesh and geometry of the punch have been chosen among different ones according to the computation time, accuracy of results and forming capability. Once the mesh and the geometry of the punch are chosen, a simple operation is simulated by means of both implicit and explicit FE models. Results in forces, displacement profiles and residual strains and stresses are compared and they are in good agreement. Experimental tests were also conducted. Forces and geometrical profiles were compared among those obtained by experiments and simulations and the results show good correspondence. By comparing the results of the explicit FE model to those obtained by the implicit FE model as well as with experimental results, we can conclude that the explicit FE model is robust enough to use the same parameters in the initial model for further studies, where multiple-strokes operations will be required and explicit methodology is expected to be mandatory.

INTRODUCTION
Incremental bulk metal forming is being satisfactorily applied to several industrial manufacturing processes; some examples are orbital forming, incremental ring rolling, incremental forging and rotary swaging, among others. The flexibility, the lower forces compared to die-defined forming and the formability of the finished parts were some of the advantages highlighted in the paper by Groche et al. (2007) in the use of incremental forming; however, the authors also indicated that these technologies have some limitations that must be overcome mainly by an advanced design. Simulations by the Finite Element Method play an important role to cope this challenge, and new simulation strategies and methodologies are being developed by researchers in metal forming field. An example is the paper of Hirt et al. (2007), where a multi-mesh method designed ad hoc is implemented for its use in incremental bulk metal forming processes such as open die forging and ring rolling. This method uses a FE mesh which is fine in the deformation zone and coarse in the remaining areas. A second mesh, fine over the entire volume, is used to store computed values and to minimize the loss of accuracy.

On the other hand, Allwood and Utsunomiya (2006) realised a survey of flexible forming processes in Japan, dividing them into two groups: flexible bulk forming and flexible sheet forming ones; the main flexible processes were described and illustrated, including as well the capability and limitations of most of them. They stated that a rich source of inspiration in searching for novel flexible processes is to pursue the following approach: given a process design applied to one type of workpiece, how it could be applied to other types of workpiece.

On the basis of this principle, Novak et al. (2008) started to study a new orbital forging process based on the classical orbital forging technology by using the Marciniak press. In a more recent paper (Novak et al. 2010), they investigated the direction of material flow as a result of subsequent incremental deformation with a simple laboratory setup composed of only three anvils. They found that the changes in the direction of deformation in the narrow region in between the first and the second anvil can foster annihilation of dislocations and lead to the increase of workability.

Allwood and Utsunomiya (2006) also categorized the flexible forming processes taking into account whether they lead to a high state of hydrostatic stress in the workpiece or not, and whether the contact with tools is continuous or intermittent. According to this categorisation, it was found that of the areas that have not yet been explored, the broad category of flexible forging was a surprising gap.

Under this background, the application of incremental methodology to develop feasible flexible forging processes is the main aim of the so called Localised-Incremental Forging (LIF). The analysis of LIF operations is often difficult, especially when several strokes are necessary to obtain the final shape of the part. The use of Finite Element models is mandatory because incremental operations throughout a workpiece are quite difficult and most of the times impossible to analyse with conventional analytical methods. It is known that in general implicit FE models give more robust results than explicit FE models; however the former
are more time consuming, mainly when multiple strokes are needed and remeshing techniques are required. The first works realised to analyse LIF operations used an implicit methodology, as it was shown in the paper of Marin et al. (2008) and Camacho et al. (2010), where simple strokes where simulated. With the main goal to extend the analysis capability of LIF operations, an explicit FE model is developed in this paper and results are compared to those obtained by implicit methodology and experimental tests.

**METHODOLOGY**

**Approach**

A single-stroke cold forming operation by a punch is going to be analysed, studying the loading and unloading cycles. The geometry of the workpiece is shown in Figure 1, and it is kept constant throughout the study.

![Figure 1: Geometry and dimensions of the workpiece](image)

The material of the workpiece is paraffin wax in order to compare the results experimentally. Its flow curve is shown in Figure 2, where a softening behaviour is observed.

![Figure 2: Flow curve of the paraffin at room temperature](image)

Because of the intermittent contact of the tool with the workpiece in LIF processes, it is feasible to get good lubrication conditions; so a friction Coulomb coefficient of 0.05 is chosen for the FE models.

**Study of the mesh**

The first task to deal with is to chose a robust mesh whose element size is small enough to provide accurate results and large enough for obtaining an efficient computation time.

The geometry of the punch for this first analysis is shown in Figure 3.

![Figure 3: Geometry of the punch for the mesh study](image)

For this purpose four typologies of mesh are analysed and the element sizes of the different zones in each mesh are presented in Table 1. The punch depth is defined large enough in order to determine which mesh allows a higher level of deformation.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>Zone 1 (mm)</th>
<th>Zone 2 (mm)</th>
<th>Zone 3 (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mesh_1</td>
<td>2.5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>mesh_2</td>
<td>5.0</td>
<td>2.5</td>
<td>-</td>
</tr>
<tr>
<td>mesh_3</td>
<td>5.0</td>
<td>2.5</td>
<td>1.0</td>
</tr>
<tr>
<td>mesh_4</td>
<td>5.0</td>
<td>1.0</td>
<td>0.5</td>
</tr>
</tbody>
</table>

![Figure 4: Typologies of mesh: (a) mesh_1, (b) mesh_2, (c) mesh_3, (d) mesh_4](image)

Figure 4 shows the four meshes and the difference of the size depending on the zones.

**Study of the geometry of the punch**

In order to analyse the most suitable punch geometry, two groups of punches are studied. The first group (Figure 5) consist of flat-ended punches, whereas the second group (Figure 6) is distinguished for round-ended punches. Each group is composed of five cross sections depending on the corner radius \( r \), as it is shown in Figure 5 and 6.
\[
    u_{i+1} = u_i + c_{i+1} \tag{3}
\]

The explicit dynamics analysis procedure is based upon the implementation of the explicit central-difference integration rule:

\[
    \ddot{u}_{i+1/2} = \frac{\Delta t_{i+1} + \Delta t_i}{2} \tag{4}
\]

\[
    u_{i+1} = u_i + \Delta t_{i+1} \dot{u}_{i+1/2} \tag{5}
\]

But the key to the computational efficiency of the explicit procedure is the use of diagonal element mass matrices because the accelerations at the beginning of the increment are computed by:

\[
    \ddot{u}_i = (M)^{-1}(P_i - I_i) \tag{6}
\]

The explicit procedure requires no iterations and no tangent stiffness matrix. Because the nature of metal forming problems is quasi-static, in this paper the simulation with explicit methodology is accomplished by mass scaling in order to increase the forming speed artificially to obtain an economical solution.

**Experimental tests**

Experimental tests have been conducted on the AMINO incremental forming machine located at the Institute of Metal Forming (IBF) of the RWTH Aachen University. This machine is typically used for incremental sheet forming; forces were registered during the experimental test. Subsequently, the final shape was analysed using the ATOS 3D digitizer system at the IBF for comparison to simulation results.

**SIMULATION RESULTS**

**Selection of a suitable mesh**

The defined operation is simulated by Abaqus/Standard using the four meshes. The computation times for the analysis are given in Table 2.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>Computation time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mesh_1</td>
<td>401</td>
</tr>
<tr>
<td>mesh_2</td>
<td>380</td>
</tr>
<tr>
<td>mesh_3</td>
<td>1013</td>
</tr>
<tr>
<td>mesh_4</td>
<td>17504</td>
</tr>
</tbody>
</table>

The values of forces versus punch displacements are represented in Figure 7. Examining in detail these results the most suitable mesh seems to be the third one (mesh_3) because the results are identical to those obtained for mesh_4; nevertheless the calculation time is considerably less and the level of deformation is higher (the achieved reduction for mesh_4 is less than 10% whereas it is approximately 30% for mesh_3).
Selection of the punch geometry

For the first group of punches (flat-ended), computation times are given in Table 3 and forces in Figure 8.

Table 3: Computation times for the simulations with flat-ended punches

<table>
<thead>
<tr>
<th>Punch</th>
<th>Computation time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (r = 1 mm)</td>
<td>524</td>
</tr>
<tr>
<td>2 (r = 1.25 mm)</td>
<td>506</td>
</tr>
<tr>
<td>3 (r = 2 mm)</td>
<td>841</td>
</tr>
<tr>
<td>4 (r = 3.33 mm)</td>
<td>370</td>
</tr>
<tr>
<td>5 (r = 5 mm)</td>
<td>658</td>
</tr>
</tbody>
</table>

For this group, the FE models are not able to achieve 10% of reduction in the stroke. This is mainly due to the sharp edges of the punch that causes numerical contact problems at the punch - workpiece interface. In the real process this geometry could also leads to surface defects such as micro cracks due to the severe contact conditions as stated by Nowak et al. (2010), so they are refused.

Figure 8: Forces versus punch displacements for the simulations with flat-ended punches

For the second group of punches (round-ended), computation times are given in Table 4 and forces in Figure 9. As it is shown, the punches 1 and 4 require the shortest computation times. However, the geometry of punch 4 (as well as the geometry of punch 5) could lead to undesirable final shapes when used for multiple incremental operations with overlapping in order to obtain a geometry throughout the workpiece, as expected in LIF operations.

Table 4: Computation times for the simulations with round-ended punches

<table>
<thead>
<tr>
<th>Punch</th>
<th>Computation time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (r = 1 mm)</td>
<td>1013</td>
</tr>
<tr>
<td>2 (r = 1.25 mm)</td>
<td>1219</td>
</tr>
<tr>
<td>3 (r = 2 mm)</td>
<td>1956</td>
</tr>
<tr>
<td>4 (r = 3.33 mm)</td>
<td>800</td>
</tr>
<tr>
<td>5 (r = 5 mm)</td>
<td>3601</td>
</tr>
</tbody>
</table>

Due to the fact that geometry of punch 1 is the one that will produce the most accurate geometrical results when used in multiple strokes operations, this geometry is chosen as the most suitable among them. Form the FE analysis it is checked that reductions up to 30% can be achived by the numerical model and the computation time is acceptable.

Comparison between implicit and numerical results

Using the mesh and the punch chosen in the former analysis, a single-stroke operation of 10% of reduction is simulated with Abaqus/Standard and Abaqus/Explicit. Figure 10 shows a graph with the evolution of the forces while the punch is forming the workpiece and after removing the punch. Results are in good agreement.

Figure 10: Evolution of forces in FE analysis with Abaqus/Standard and Abaqus/Explicit

The final shape is as well one technological factor to consider in LIF processes. In order to evaluate the geometrical accuracy provided by the numerical models Figure 11 is included. In this figure the geometrical profile is represented for two different punch positions: when the punch is applying the maximum reduction and it is still in contact with the workpiece (Step_1), and when the punch is
removed enough from the surface to register any springback effect (Step 2). Results for both methodologies are in good accordance with each other in the contact zone with the end of the punch. However, some differences are found in the surface of the workpiece, where a bulging effect appear. The comparison with experimental tests will help to clarify if actually this effect takes place or if it is a numerical problem.

![Figure 11: Final shape obtained by the FE models when the punch is applying the maximum force and when the punch is removed](image)

Residual strains and stresses are also of interest in the analysis of LIF processes. In incremental bulk forming the deformation mechanism is complex mainly due to the fact that the strain paths are changing from one loading cycle to another (Groche et al., 2010). Residual equivalent plastic strains at the end of the unloading cycle (Step 2) are shown in Figure 12 for the implicit and explicit models. The results show an identical behaviour in the punch-workpiece interface, and small differences are observed in the surroundings. Maximum strains are induced in the edges in contact with the tool.

![Figure 12: Residual equivalent strains at the end of the unloading cycle](image)

Residual equivalent stresses are also evaluated. They are exhibited in Figure 13.

![Figure 13: Residual equivalent stresses at the end of the unloading cycle](image)

Good agreement is also found. In this case, maximum stresses are registered in the surroundings of the deformation zone, not strictly in the punch-workpiece interface; and the minimum values of stresses are found in the center of the deformation zone. This has to be taken into account in the application of multiple loading cycles throughout the workpiece because it can help to find the best overlapping level, combining the most stress affected zones with the less affected ones. Finally, residual plastic strain contours on the surface of the workpiece are shown in Figure 14 for both methodologies. The contours exhibit also a similar behaviour.

![Figure 14: Residual equivalent plastic strain contours on the surface of the workpiece: (a) Abaqus/Standard, (b) Abaqus/Explicit](image)

**Validation with experimental results**

Simulation results are compared to those obtained by experimental tests. Figure 15 shows the scanned surface by the ATOS 3D digitizer system.
Forces are measured by a load cell installed in the AMINO machine. The comparison of the forces with the Abaqus/Standard results is shown in Figure 16. This figure demonstrates that the prediction of forces by the FE analysis is quite good.

The comparison of the geometrical profile with the experimental test is shown in Figure 17. As it is seen, the geometrical results in the center indicate a less level of springback that the one obtained by the simulation and the bulging effect is not so pronounced. This demonstrate that the bulging effect showed in the simulations does not actually take place; it can be due to the contact algorithm, so special care must be taken when using this model for geometrical purposes.

CONCLUSIONS

The review of literature shows that limitations of incremental bulk forming must be overcome by advanced design, such as FE simulation. Based on the results of an implicit FE model, a robust FE explicit model for the simulation of LIF operations has been developed. The most suitable mesh and geometry of the punch have been chosen among different ones according to the computation time, accuracy of results and forming capability. Experimental test were also conducted to validate the results. By comparing the results of the explicit FE model to those obtained by the implicit FE model as well as with experimental results, we can conclude that the explicit FE model is robust enough to use the same parameters in the initial model for further studies, where multiple-strokes operations will be required. By analysing the residual strains and stresses, interesting information can be extracted in order to plan further forming strategies such as overlapping when multiple load cycles are applied.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Institute of Metal Forming of the RWTH Aachen University, especially Prof. Hirt, Dr. Bambach and A. Göttmann as well as the IBF staff for the kind supports; the authors also acknowledge the Ministry of Science and Innovation of Spain for funding this work via Project DPI2009-07300, and the Ministry of Education of Spain via National Human Resources Mobility Programme (Grant “José Castillejo”) of the National R&D&I Plan 2008-2011.

REFERENCES


Hibbitt, D., B. Karlsson and P. Sorensen. 2010. ABAQUS v6.10 User’s Manuals, Providence, RI.
AUTHOR BIOGRAPHY

ANA MARÍA CAMACHO was born in Ciudad Real, Spain. She studied at the University of Castilla-La Mancha (UCLM) where she received the M.Sc. Degree Industrial Engineering in 2001. In 2005 she obtained the Ph.D. Degree from the National Distance University of Spain (UNED). Nowadays she works as Associate Professor at the Department of Manufacturing Engineering of the UNED. Her research interest is focused in simulation of metal forming processes by the Finite Element Method and distance learning applications.
REAL-TIME SIMULATION OF AUTOMATED MECHANICAL TRANSMISSIONS

Marius Bataus  
Automotive Engineering Department  
University Politehnica of Bucharest  
313 Spl. Independentei  
Bucharest, Romania, 060042  
E-mail: marius.bataus@upb.ro

Daniela Vasiliiu  
Fluid Power Department  
University Politehnica of Bucharest  
313 Spl. Independentei  
Bucharest, Romania, 060042  
E-mail: vasiliiud@hydro.pub.ro

Nicolae Vasiliiu  
Fluid Power Department  
University Politehnica of Bucharest  
313 Spl. Independentei  
Bucharest, Romania, 060042  
E-mail: vasiliiu@fluid-power.pub.ro

KEYWORDS
Automotive Transmissions, Dual Clutch, Real-Time Simulation

ABSTRACT
The paper details with the state of the art in the field of modeling and simulation of automotive mechanical transmissions equipped with hydraulic control systems. The emphasis is put on real-time simulation of the complex system containing the control unit, the powertrain and the vehicle. Several investigation methods useful for model conversion from desktop to real-time simulation are discussed, and models of transmissions key components are adapted for real-time simulations. A complex model of a powertrain equipped with a dual clutch transmission that includes the clutch control hydraulics is used as an example. The model is implemented using the 1D multi-domain simulation software LMS Imagine.Lab Amesim and exported to a dSPACE platform using the Real-Time Workshop toolbox from Matlab.

INTRODUCTION
The automatic control of mechanical transmissions represents one of the most effective solutions for the improvement of comfort, dynamic performances and the reduction of fuel consumption and emissions of automotives. There are currently several types of automated mechanical transmissions in production or in research phase. The automated manual transmission (AMT) provides the best efficiency at low costs but suffers due to the power flow interruption at gearshift. In order to overcome this driving torque interruption, different devices (assist clutch, planetary gear set or flywheel) can be used to assist the transmission during the gearshift. Dual clutch transmissions (DCT) realize gearshifts by torque transfer from one clutch to another without traction force interruption. The automated mechanical transmission with planetary gear sets is an old concept that becomes feasible due to the increasing of engine torque and the improvement of clutch control. It can be obtained from an automatic transmission (AT) by eliminating the torque converter. Continuously variable transmissions (CVT) have a low efficiency but can allow the engine to operate at favorable torques and speeds. The diversity of the engine-transmission-vehicle combinations offered by the automotive manufacturers and the reduction of development times makes necessary the extensive use of Hardware-in-the-Loop (HiL) simulation for conception, testing and calibration of the control software and electronics components.

This paper aims to present typical modeling of automated mechanical transmissions and real-time simulation issues involved in these applications. Models of key components (clutches, synchronizers, hydraulic pistons) adapted for real-time are described and a realistic plant model including clutch control hydraulics and mechanics is developed. The model is implemented using the 1D multi-domain simulation platform LMS Imagine.Lab AMESim (which will be referred as AMESim). This platform is particularly suited for powertrain and hydraulic applications (Hayat et al. 2003; Bataus et al. 2010; Dragne et al. 2009). The AMESim RT (real-time) option enables the export of a model to a real-time environment such as dSPACE or xPC for use in HiL simulation. A dSPACE platform was used in order to evaluate the benchmark problems and to demonstrate the real-time performance of the complex powertrain model.

REAL-TIME SIMULATION DEMANDS
For a simulation to execute in real-time, the amount of time spent calculating the solution for a given time step (execution time) together with the amount of time spent processing inputs, outputs, and other tasks must be less than the length of that time step. This involves the limitation of the execution time for every time step. As a consequence, the variable-step solvers, which are preferred for desktop (offline) simulations, are unsuitable for real-time ones due to the temporary use of very small steps to accurately capture events that occur during the simulation. A fixed-step solver (implicit or explicit) must be used for this type of simulation. The model and the fixed-step solver must be tuned to allow the accurately capture of system dynamics without changing the step size. In order to produce accurate results, different fixed-step solver algorithms (implicit, explicit, lower/higher order, etc.) need different step size and computational effort per time step. The most used integration algorithms for real-time simulation are the explicit ones and the Euler one is preferred in the case of powertrains. Higher order algorithms (e.g. Runge-Kutta) can ensure a better precision at the expense of the longer execution times but only for a smoothly variation of the state variables in the sampling interval (Schuette and Wailerlmann 2005). This variation is impossible for the hydraulically operated powertrains due to Coulomb friction, gearshifts and low hydraulics volumes.

168
Promising results were obtained using implicit algorithms but with limited number of iterations (Schuette and Waeltermann 2005; Miller and Wendlandt 2010). After the solver type, a proper step size must be chosen:
- An increase time step offers more time to calculate the results but can lead to inaccurate results;
- A reduced time step improves the accuracy of the results but can lead to overruns.

Typical sampling rates of 0.5-1 ms are used for vehicle dynamics and powertrain application but can be necessary to use steps of 50-250 μs in the case of motorsports applications (e.g. F1 race cars) (Schuette and Waeltermann 2005; Waeltermann et al. 2004).

Often a trial and error process is used to find the combination of settings that allow the real-time simulation with acceptable accuracy. However, a number of existing analysis tools can allow the time step evaluation. If this combination of solver-step size cannot be found, it is necessary to simplify the model or use other techniques to accelerate the simulation.

The execution time can be reduced by using a local solver or multiple evaluations for the stiff systems parts (subsystems). An increased computation power and higher I/O processing speeds allows the runs of more demanding models.

MOVING FROM DESKTOP TO REAL-TIME SIMULATION

A real-time model is usually obtained by the simplification of the offline model and the appropriate selection of the model parameters.

To move from desktop simulation to real-time simulation on the chosen real-time hardware, the following items can be adjusted:
1. Solver type,
2. Number of solver iterations,
3. Step size,
4. Model size and fidelity.

The challenge is to find appropriate settings that provide accurate results (results sufficiently close to those obtained from desktop simulation) while permitting real-time simulation. Recommendations for this process are given in (Miller and Wendlandt 2010) and (Bataus et al. 2010).

If it is not possible to find a combination of settings that will allow real-time simulation on the chosen real-time platform, it is necessary to modify or eliminate the elements that are incompatible with this type of simulation. There are two classes of elements:
1. Elements that creates events (discontinuities) in the simulation;
2. Elements that have a very small time constant.

When an event takes place the solution changes almost instantaneous. This rapid change makes difficult for the fix-step solver to step over the event and to find the good solution on the other side where the system is defined by a new set of equations. Some examples of such elements representative for hydraulic controlled automated transmissions are: stick-slip friction, end stops, backlash and switches.

When an element or a group of elements have a very small time constant, a small step size it is necessary to accurately capture the system dynamics. Some representative examples for hydraulic controlled automated transmissions are: small masses (inertias) attached to stiff springs, hydraulic circuits with small volumes.

The analysis of step size during variable-step simulation can provide a rough idea of a step size that can be used to run the simulation.

In AMESim is possible to take advantage of the linear analysis facility. It gives in easy way the eigenvalues of a system (with real and imaginary parts) around a linearization time. These values can be used to establish the maximum step size for a fix-step solver. For explicit Euler fixed step solver (the most used for real-time simulation) the following conditions have to be verified at any time to have stable integration:
- Stable integration of undamped modes;

\[
f_x \geq \frac{(2 \cdot \pi \cdot f_j)^2}{2 \cdot R_i} \quad (1)
\]

- Stable integration of full damped modes;

\[
f_x \geq \frac{-R_i}{2} \quad (2)
\]

- Stable and no oscillating integration of full damped modes;

\[
f_x \geq -R_i \quad (3)
\]

Where:
- \( f_j \) is the frequency of the \( i \) mode (Hz);
- \( f_k \) is the Euler fixed step solver frequency (Hz);
- \( R_i \) is the real part of the \( i \) mode;
- \( \rho_i \) is the imaginary part of the \( i \) mode.

If the necessary step size is too small a simplification of the model is needed. A possible idea for model simplification is to use the power exchanges between system parts. The assumption is that in an energetic system the most important components to model accurately are those characterized by the largest magnitudes of energy or power flow. Therefore, is possible to simplify a given model by eliminating less energetic components, while trying to minimize the effect of the elimination on the overall energy flow.

An energy-based model reduction metric called activity was introduced for this purpose (Louca et al. 1997). The activity of an energetic element is defined as the time integral of the absolute value of the power flowing through it over a particular time-window for a particular input:

\[
A_i = \int P_i(t) \cdot dt \quad (4)
\]

In a bond-graph setting, were the elements exchanges flows and efforts, the power is:

\[
P_i = f_i \cdot e_i \quad (5)
\]

Where:
- \( e_i \) is the flow through the element \( i \);
- \( f_i \) is the effort across the element \( i \).
The activity index of element $i$ of a submodel ($AI_i$) is the ratio between the element activity and the total activity of the system:

$$AI_i = \frac{A_i}{\sum_{\text{SYSTEM}} A_i} \quad (6)$$

Another tool for model simplification implemented in AMESim is the State count. This facility can be used to identify the reason for a slow run. Convergence tests and error tests are made on each state variable at every integration step. For a particular step one variable will be the slowest to converge or will hold the biggest error. The State count identifies the state variable that as far as the integrator is concerned is the most difficult and a sorted list of variables can be produced.

If the model can be run with a fix-step solver, the next step is to evaluate the real-time capabilities of this model. The speed of simulation on the desktop can be used to estimate the execution time on the real-time target. Because of the multitude of factors that affect the execution time on the real-time target, comparing processor speed may not be sufficient. A better method is to measure the execution time during desktop simulation and then to determine the average execution time per step on the real-time platform for a given model. Knowing this conversion factor for one model makes it possible to estimate execution time on the real-time platform from the execution time during desktop simulation when testing other models.

**COMPONENTS MODELS FOR REAL-TIME SIMULATION**

The models used for the components (referred as submodels) must be compatible with real-time simulation. For the mechanical transmissions the most critical submodels are those of the clutch and the synchronizer. The modeling of the hydraulic control circuit for real-time applications is also difficult.

**Clutch model**

The friction modeling constitutes the base of all clutch models. Many examples of friction models are proposed: hyperbolic tangent model, Karnopp model, Reset-Integrator model, Dahl model, Bristle model, LuGre model etc. The model of friction must be chosen taking into account the purposes of use. In (Bataus 2011) it is shown that the clutch model base on Reset Integrator friction modeling is the most efficient in terms of execution time. The study was done using a 3 degree of freedom (dof) powertrain model (figure 1) and using parameters values from a middle class passenger car.

$$T = T_s \cdot \tanh \left( \frac{2\omega}{\omega_0} \right) \quad (7)$$

When engaged the synchronizer act as an angular spring-damper having the stiffness $k$ and the damping coefficient $b$.

$$T_s = k \cdot \omega_s + b \cdot \omega_t \quad (8)$$

**Synchronizer model**

The synchronizers cause discontinuities in the passage from synchronization to engaged (coupled) gear: the interaction between elements, initially due to friction is then realize by the contact of the dog-teeth. A number of modeling techniques for the synchronizer are available. The simplest use only the friction torque between the sleeve and the idle gear both for synchronization of the two velocities and the locking of the gear (LMS Imagine 2009). This 2 dof model can be satisfactory for fuel consumption studies but is not suitable for comfort ones. In order to transmit the maximum torque it needs a friction torque more than 10 times greater than the real one. This will produce an extremely reduced synchronization time.

Using a coupling logic that increase the friction torque when the velocities are matched it was possible to employ this model also for comfort studies (Bataus et al. 2005). However this solution cannot be used with a fix step solver with the usual step sizes for real-time applications.

Models with variable structures are also used. These models have 2 dof for uncoupled state and 1 dof for the coupled state (Galvagno et al. 2011). The drawback of the model is that it forms an integrated part with the rest of the system. Therefore, the transmission model equations have to be tailor-made for each configuration. Variations of this model are widely used since they allow efficient simulations.

An original submodel was developed for the synchronizer. The submodel has 4 ports (figure 2): one for the command and three for the mechanical connection to the shaft and the idle gear. The engagement and disengagement are commanded using as input a command signal com or a force $F_i$ applied to the sleeve. The port number is used as index for the torque $T$ and the angular speed $\omega$ passed at every connection port.

Figure 2: Synchronizer icon with input and output variables

The new model is build using hybrid modeling techniques (using continuous models triggered by discrete events) and ensures three phases: disengaged (no torque is transmitted), synchronization (the synchronizer is similar with a clutch) and engaged (the synchronizer is similar with a shaft).

In synchronization phase the transmitted torque is calculated from the synchronization torque $T_s$ using a hyperbolic tangent friction model.

$$T_s = T \cdot \tanh \left( \frac{2\omega_s}{\omega_0} \right) \quad (7)$$

When engaged the synchronizer act as an angular spring-damper having the stiffness $k$ and the damping coefficient $b$.

$$T_s = k \cdot \omega_s + b \cdot \omega_t \quad (8)$$
When compared with a detailed physical model it produces an error of only 3% for the coupling time, figure 3. The speed is dramatically improved, for 0.8 simulated seconds the computing time is reduced from 4.018 s to 0.036 s.

Figure 3: The speeds of coupled elements obtained with different synchronizer models

This synchronizer model makes possible the easy modeling of the most complex transmission architectures. Figure 4 shows the layout of the Volkswagen DSG 02E transmission. This complex DCT has two outputs shafts that combine the two partial transmissions (the first output shaft for gears 1, 2, 3 and 4; the second output shaft for gears 5, 6 and reverse).

Figure 4: Layout of DSG 02E

The transmission is modeled as a 4 dof system. This complexity level is similar or higher than that of other models used for gearshift dynamics and control (Galvagno et al. 2011; Kulkarni et al. 2007; Walker et al. 2011). The physical mechanical model includes: inertias, clutches, gear sets, synchronizers and final drive, figure 5.

Figure 5: AMESim simulation network of DSG 02E

**Piston model**

Since the usual hydraulic component models are not suitable for real-time simulation a special piston model is used. This piston model includes the velocity integration to suppress the need of the mass model (Alirand et al. 2005).

![Image of HCD piston model](image)

Figure 6: a) The standard HCD piston model; b) The real time piston model

The standard model use the piston velocity \(v_2\) (obtained from the mass model) to compute the flow rate \(Q_l\) and the pressure \(p_1\) to compute the spool force.

\[
Q_l = A_s \cdot v_2
\]  
\[
\sum F = A_s \cdot p_1
\]

Where \(A_s\) is the piston area.

The real time model use the flow rate (usually computed inside an orifice model) to compute the velocity and then the displacement. The pressure results from the force balance.

\[
v_2 = \frac{dx}{dt} = \frac{Q_l}{A_s}
\]

\[
p_1 = \frac{\sum F}{A_s}
\]

This submodel is used to model the hydraulic actuation of the clutches. The new hydraulic control systems of the transmission clutches are based on direct acting solenoids with high flow instead of pilot solenoids controlling flow.

171
control valves which have higher time delays (Robert Bosh GmbH 2004; Walker et al. 2011).

Figure 7 represents a simplified clutch hydraulic control system including one clutch piston and the proportional control valve. The clutch pack is modeled as a gap and end-stop. The line pressure is a function of a command signal. The complexity of the model is adequate for the study of transmission clutch control (Lucente et al. 2007; Walker et al. 2011).

![Figure 7: Simplified hydraulic system for clutch control](image)

A good correlation with experimental data given in (Lucente et al. 2007) is obtained.

**SIMULATION RESULTS**

Gearshift dynamics can only be simulated if the input and output torques of the transmission represent a real-life vehicle maneuver. Therefore, a global model that includes the engine, the transmission and the longitudinal dynamics of the vehicle was constructed. This model is used to obtain the reference results using an open-loop control. The model is then tuned for real-time simulation by applying the previously presented methods. This iterative process was done in AMESim using the dedicated tools: Linear analysis, State count and Activity index. The imposed target for the step size is 0.5 ms.

An offline test for accuracy is done. The results of the model with parameters tuned for real-time obtained with an Euler solver with 0.5 ms step size are compared with those of the reference model obtained with the standard variable step solver, figure 9.

![Figure 9: The vehicle horizontal acceleration obtained with different solvers](image)
The connection between the AMESim powertrain model and the traction control system (TCU) model implemented in Simulink is establish in order to export the model on a real-time platform.

The coupled AMESim-Simulink model is tested for the TCU shift logic. The test cycle consist of two launches from standstill at wide open throttle and at half engine load followed by brake to zero velocity. Figure 10 shows the results of this test. This plot shows a good variation of the engine speed during the start-up. Also, correct values for the engine speed at the gear changes are obtained both at up-shifts and down-shifts.

Finally the model was simulated in real-time with a sample rate of 2 kHz on the dSPACE RT platform equipped with ds1006 processor board (2.6 GHz). The results show a turnaround time of maximum 0.035 ms and no overruns. The model is fast enough to allow the use of complex control software for the transmission.

![Figure 10: Testing of the TCU shift logic](image)

Control Desk was used to conduct the experiment on the dSPACE real-time platform. This universal experiment software for electronic control unit development provides access to simulation platforms and connected bus systems, and can perform measurement, calibration and diagnostics on ECUs. It also allows the construction of friendly user interfaces for model control, data visualization and acquisition.

Figure 11 shows the result of a wide open throttle start-up as view in one of the dedicated Control Desk interface.

![Figure 11: Real-time simulation results (Control Desk Interface)](image)
This interface allows the visualization of:
- Vehicle velocity and longitudinal acceleration;
- Engine and transmission shafts speeds;
- Preselected gears;
- Clutches commands, actuation pressure and force;
- Synchronizers sates.
It is also use to control the simulation state (stop, pause or run), the model type of control (manual, commands cycle or velocity cycle), the acceleration and brake pedals in the manual mode and some model parameters (e.g. vehicle mass).

CONCLUSIONS

It was demonstrated that is possible to simulate in real-time high-fidelity models of hydraulically controlled automated mechanical transmissions coupled with complex vehicle dynamics models.
It is almost impossible to turn a complex offline model to a real-time one without proper tools. Depending on the simulation software a number of efficient analysis methods can be used for model simplification and parameter tuning in support of real-time simulation. For the presented application the most useful AMESim dedicated tools were: Linear analysis, Activity index and State count.
Because of the facilities offered by AMESim for the elaboration of the models and the extended use of Simulink for the development of the control the AMESim-Simulink connection is a useful tool for developing powertrain models for real-time simulation.
The dSpace real-time platform equipped with ds1006 processor board make possible the execution of very complex powertrain models and allows, in conjunction with the Control Desk software the proper control of the envisaged experiments.

REFERENCES

MODELING OF THE ADVANCED SPIN TRANSFER TORQUE MEMORY:
MACRO- AND MICROMAGNETIC SIMULATIONS

Alexander Makarov, Viktor Sverdlow, Dmitry Osintsev, Josef Weinbub, and Siegfried Selberherr

Institute for Microelectronics
Vienna University of Technology
1040, Vienna,
Austria

Email: {makarov|sverdlow|osintsev|weinbub|selberherr}@iue.tuwien.ac.at

KEYWORDS
STTRAM, MRAM, magnetic tunnel junction, macro and micromagnetic modeling

ABSTRACT

We study the dynamics of the switching process in a magnetic tunnel junction composed of 5 layers with the magnetization of the two side layers fixed. The magnetization of the middle free layer can be switched between the two stable configurations by passing the current through the tunnel junction in a certain direction. The dependence of the switching time on the parameters of the penta-layer structure is analyzed.

INTRODUCTION

Memory cells based on electric charge storage, such as flash memory, are rapidly approaching the physical limits of scalability. The increasing demand for mini-}

Figure 1: Schematic illustration different MTJ types: (a) penta-layer MTJ; (b) three-layer MTJ.

mization of microelectronic devices (e.g., MP3 players and mobile phones) stimulates a significant acceleration in exploring the new concepts for nonvolatile memory. Apart from good scalability, a new memory type must also exhibit low operating voltages, low power consumption, high operation speed, long retention time, high endurance, and a simple structure. (Kryder et al. 2009)

Several concepts were recently proposed and developed for potential replacement of the charge based memory. Some of the proposals are available as prototypes, such as carbon nanotube RAM (CNRAM) and copper bridge RAM (CBRAM), others already as products, e.g., phase change RAM (PCRAM), magnetoresis-

tive RAM (MRAM), and ferroelectric RAM (FRAM), while the technologies based on spin tunnel transfer RAM, racetrack memory (RTRAM), and resistive RAM (RRAM) are under intensive research.

The theoretical predictions (Slonczewski 1996; Slon-
czewski 2005) and the experiments (Braganca et al. 2005; Iwayama et al. 2008; Meng et al. 2006; Fuchs et al. 2005; Devolder et al. 2005) of spin transfer switching demonstrated that the spin transfer torque random access memory (STTRAM) is one of the promising can-
didates for future universal memory. STTRAM is characterized by small cell size (4F^2), fast access time (less than 10ns), high endurance (10^{10}), and long retention time.

The basic element of the STTRAM is a magnetic tunnel junction (MTJ). The three-layer MTJ (Fig.1b) represents a sandwich of two magnetic layers separated by a thin insulating spacer which forms a tunnel barrier. While magnetization of the pinned layer is fixed during the fabrication process, the magnetization direction of the free layer can be switched between the two states parallel and anti-parallel to the fixed magnetization direction. Switching between the two states occurs due to spin-polarized current flowing through the MTJ.

The spin-polarized current is only a fraction of the total charge current. Therefore high current densities are required to switch the magnetization direction of the free layer. The reduction of the current density re-

quired for switching and the increase of the switching speed are the most important challenges in STTRAM research. Several strategies have been proposed to de-

crease the switching time below a few nanoseconds: by pre-charging with a bias current (Devolder et al. 2005), by combining a spin-polarized current together with a small radio frequency field (Finocchio, Krivorotov, et al.
by applying a magnetic field perpendicular to the magnetization direction (Devoorder et al. 2006).

Measurements performed by (Fuchs et al. 2005) showed a decrease in the critical current density for the penta-layer magnetic tunnel junction shown in Fig.1a. The structure represents a magnetic tunnel junction composed of 5 layers, with the magnetization of the two side layers fixed. The spin torque enhancement in penta-layer structures results in a significantly lower critical switching current at a switching delay comparable to that in three-layer structures, which makes the penta-layer cells attractive for low power high performance memory applications. In this work we briefly describe an implementation of a penta-layer model in our micromagnetic simulation environment. It allows us to investigate the dynamics of the switching process in a penta-layer MTJ. Such a penta-layer structure was recently analyzed (Mojumdar et al. 2010) by using the ballistic Green’s function formalism combined with the soft magnetic layer dynamics based on the Landau-Lifshitz-Gilbert equation. The spin torque enhancement was found in the anti-parallel penta-layers (the magnetizations of the two fixed layers are anti-parallel) as compared to the three-layer structure. This enhancement manifests itself only under the dual barrier resonance tunneling conditions, when the current is high.

At the same time, the aligned penta-layer configuration, when the magnetizations of the two fixed layers are parallel to each other, was found to have a fairly low spin torque efficiency and, as a consequence, it demands high switching currents (Fuchs et al. 2005). This fact cannot be understood within the formalism employed in (Mojumdar et al. 2010). Indeed, the quantum effects leading to the double resonant conditions and high spin torque efficiency in the anti-parallel structure could be equally well applied to the parallel one. In order to clarify the issue, we performed extensive micromagnetic modeling of the penta-layer structure. In contrast to (Mojumdar et al. 2010) we employ the Slonczewski model (Slonczewski 1996; Slonczewski 2005) for the spin torque. The use of this model is justified in the structures with a free ferromagnetic layer thickness of a few nanometers. Indeed, the electron spins become aligned with the fixed magnetization at a distance approximately 1nm away from the interface (Datta et al. 2009). We investigated the structure CoFe/Cu/Py/AlOx/CoFe, where Py is Ni81Fe19.

**MODEL DESCRIPTION**

Our simulations are based on the magnetization dynamics described by the Landau-Lifshitz-Gilbert-Slonczewski equation:

\[
\frac{d\mathbf{m}}{dt} = -\frac{\gamma}{1+\alpha^2} \cdot ((m \times h_{eff}) + \alpha \cdot [m \times (m \times h_{eff})] + \frac{g\mu_B}{e\gamma M_a d} \cdot (g_1(\Theta_1) \cdot (\alpha \cdot (m \times p_1) - [m \times (m \times p_1)]) - g_2(\Theta_2) \cdot (\alpha \cdot (m \times p_2) - [m \times (m \times p_2)]))
\]  

Here, \(\gamma\) is the gyromagnetic ratio, \(\alpha\) is the Gilbert damping parameter, \(g\) is the gyromagnetic splitting factor, \(\mu_B\) is Bohrs magneton, \(j\) is the current density, \(e\) is the electron charge, \(d\) is the thickness of the free layer, \(m = M/M_a\) is the position dependent normalized vector of the magnetization in the free layer, \(p_1 = M_{p1}/M_{p1}\).
and $p_2 = M_{p2}/M_{sp2}$ are the normalized magnetizations in the first and second pinned layers, respectively. $M_s$, $M_{sp1}$, and $M_{sp2}$ are the saturation magnetizations of the free layer, the first pinned layer, and the second pinned layer, correspondingly. We use Slonczewski’s expressions for the gyromagnetic splitting factor in the MTJ with a dielectric layer (Slonczewski 2005)

$$g_1(\Theta) = 0.5 \cdot \eta [1 + \eta^2 \cdot \cos(\Theta)]^{-1}$$

and with a metal layer (Slonczewski 1996)

$$g_2(\Theta) = [-4 + (1 + \eta)^3 (3 + \cos(\Theta))/4\eta^{3/2}]^{-1}$$

between the ferromagnetic contacts, respectively. In the pentalayer structure the two spin torques are acting independently on the two opposite interfaces of the free ferromagnetic layer, provided its thickness is larger than the scale on which the electron spins entering into the ferromagnet become aligned to the ferromagnets magnetization. The local effective field is calculated as:

$$h_{eff} = h_{ext} + h_{ani} + h_{exch} + h_{demag} + h_{th} + h_{amp} + h_{ms}.$$  

Here, $h_{ext}$ is external field, $h_{ani}$ is anisotropic field, $h_{exch}$ is a exchange field, $h_{demag}$ is a demagnetizing field, $h_{th}$ is a thermal field, $h_{amp}$ is the Ampere field, and $h_{ms}$ is the magnetostatic coupling between the pinned layers and the free layer.

In the uniaxial anisotropy case the anisotropic field is (Miltat and Donahue 2007):

$$h_{ani} = \frac{2K_1}{\mu_0 M_s} (m \cdot u) u,$$

while for the cubic anisotropy it is calculated as:

$$h_{ani} = -\frac{2D}{\mu_0 M_s} m.$$  

Here, $D$ is the diagonal matrix with entries

$$D_{11} = K_1 (m_x^2 + m_z^2) + K_2 m_y^2 m_z^2,$$

$$D_{22} = K_1 (m_y^2 + m_z^2) + K_2 m_x^2 m_z^2,$$

$$D_{33} = K_1 (m_x^2 + m_y^2) + K_2 m_x^2 m_y^2.$$  

$K_1$ and $K_2$ are the material-dependent anisotropy coefficients, $u$ is the easy axis, $\mu_0$ is the magnetic constant. The exchange field is calculated as (Miltat and Donahue 2007):

$$h_{exch} = \frac{2A}{\mu_0 M_s} \sum_j ((m_j - m) / |r_j|^2).$$

Here, $A$ is the exchange constant.

For calculating the demagnetization field we used the method proposed in (Kákay 2005).

![Figure 3: Evolution of the magnetization of the free ferromagnetic layer during switching in a pentalayer structure: (a) from anti-parallel to parallel configuration; (b) from parallel to anti-parallel configuration.](image-url)

The thermal field is calculated as (Ito et al. 2006):

$$h_{th} = \sigma \cdot \sqrt{\frac{\alpha}{1 + \alpha^2}} \cdot \frac{2k_B T}{\gamma \Delta V \Delta t M_s}.$$  

Here, $\sigma$ is a Gaussian random uncorrelated function, $k_B$ is the Boltzmann constant, $\Delta V$ is the volume of cell, $\Delta t$ is the time step.

The eddy currents field is (Torres et al. 2003):

$$h_{amp,i} = \sum_{j=1..N} \frac{J_j}{4\pi} \times \int_{r_j} \frac{r_i - r_j}{r^3} dv.$$  

Here, $J_j$ is the current induced on every cell ($j : 1..N$). For evaluation of the integrals we used methods proposed by (Tomáš 1999).
RESULTS

All simulations are performed for the nanopillar structure proposed in (Fuchs et al. 2005). The geometry of the nanopillar is defined as CoFe(8nm)/AlOx(0.7nm)/Py(4nm)/Cu(6nm)/CoFe(5nm), with an elliptical cross-section (major axes are 90nm and 35nm, correspondingly). The other parameters of our simulations are: $T=77K$, $\gamma = 2.3245 \cdot 10^{7} m/(A-s)$, $\alpha = 0.01$, $A = 1.3 \cdot 10^{-11} J/m$, $M_s = 644 \cdot 10^3 A/m$, $M_{sp} = 1.15 \cdot 10^4 A/m$, and $\eta_1 = 0.3$ and $\eta_2 = 0.35$ for the MTJ with the dielectric spacer and the metal spacer, respectively. We simulated the switching process under an applied spin current with the density fixed at $j = 0.1 \cdot 10^9 A/cm^2$ and an external magnetic field $h_{ext} = 26 mT$ applied along the negative direction of the $x$ axis. Simulation results for macro-spin approximation are shown in Fig.2 and Fig.3. The direction of the free layer magnetization is indicated with respect to the magnetization of the pinned layer in the CoFe(8nm)/AlOx(0.7nm)/Py(4nm) MTJ. Fig.3a demonstrates good agreement of the evolution of the magnetization in the free magnetic layer during switching with the results for an ideal elliptical cross-section at 77K reported in (Finocchio et al. 2007) and obtained by using micromagnetic modeling.

The limitation of the macro-spin approximation is that it cannot take into account the exchange field and the field of the eddy currents. It also does not allow to calculate accurately the magnetostatic coupling between the pinned layers and the free layer, and the demagnetization field. Without these fields it is difficult to obtain reliable results.

A snapshots of the eddy currents field is illustrated in Fig.4a. Fig.4b demonstrates the field of the magnetostatic coupling between the pinned layers and the free layer.

In the following we investigate the switching process from the anti-parallel to the parallel state with full micromagnetic simulation and for two types of initial conditions for the magnetization (Fig.5). In the case of unidirectional initial magnetization parallel to the $x$ axis we obtain practically the same result as for our macro-spin model. However, in the case, when the magnetization was allowed first to relax, the switching time is two times longer than in the case of unidirectional initial magnetization. This result demonstrates the importance of properly accounting for the magnetization relaxation under the influence of the external fields to correctly obtain the initial magnetization.

CONCLUSION

Magnetic tunnel junctions with the magnetization of the two side layers fixed are studied by means of a macro-spin approximation and the extensive micromagnetic calculations. Our results demonstrate that, despite all the limitations of a macro-spin model, it can provide fast and relatively accurate results for a certain set of parameters. We also show the importance of properly accounting for the magnetization relaxation under the influence of the external fields. Our simulation environment is thus perfectly suited for optimization of STTRAM cells.
ACKNOWLEDGEMENTS

This research is supported by the European Research Council through the grant #247056 MOSILSPIN.

REFERENCES


A SIMULATION STUDY OF DATA PROCESSING ACTIVITIES IN CMS PHYSICS ANALYSIS

Ciprian Dobre
University POLITEHNICA of Bucharest
Spl. Independentei, 313, Bucharest, Romania
E-mail: ciprian.dobre@cs.pub.ro

KEYWORDS
Modeling and simulation, evaluation, large scale distributed systems, LHC experiments, CMS.

ABSTRACT
The scale, complexity and worldwide geographical spread of the Large Hadron Collider (LHC) computing and data analysis problems are unprecedented in scientific research. The complexity of processing and accessing this data is increased substantially by the size and global span of the major experiments, combined with the limited wide area network bandwidth available. This paper discusses the latest generation of the MONARC (MODels of Networked Analysis at Regional Centers) simulation framework, as a design and modeling tool for large scale distributed systems applied to High-Energy Physics (HEP) experiments. We present a simulation study designed to evaluate the capabilities of the current real-world distributed infrastructures deployed to support existing LHC physics analysis processes and the means by which the experiments bands together to meet the technical challenges posed by the storage, access and computing requirements of LHC data analysis. The Compact Muon Solenoid (CMS) experiment, in particular, uses a general-purpose detector to investigate a wide range of physics. We present a simulation study designed to evaluate the capability of its underlying distributed processing infrastructure to support the physics analysis processes. The results, made possible by the MONARC’s model, demonstrate that the LHC infrastructures are well suited to support the data processes envisioned by the CMS computing model.

1. INTRODUCTION

Modeling and simulation were seen for a long time as viable solutions to develop new algorithms and technologies and to enable the enhancement of large-scale distributed systems, where analytical validations are prohibited by the scale of the encountered problems. The use of discrete-event simulators in the design and development of large scale distributed systems is appealing due to their efficiency and scalability.

The Large Hadron Collider (LHC) is a gigant particle accelerator consisting of a circular tunnel with a circumference of 27 km (the largest in the world), around which beams of protons and anti-protons (and heavy ions such as lead nuclei) are accelerated in opposite directions to nearly the speed of light (LHC Experiments, 2011). At four points on the ring of the accelerator, the beams of particle and anti-particles cross and collide with each other at extremely high energies, close to the energies of the first split seconds after the Big Bang, to produce other kind of particles. The experiments building each detector are ALICE, ATLAS, CMS and LHCb, and each is designed to study a different area of particle physics.

The scientific wealth of the experiments presents new problems in data access, processing and distribution, and collaboration across national and international networks, on a scale unprecedented in the history of science. The information technology challenges are introduced by the need to provide rapid access to data subsets drawn from the massive data stores. Approximately 10-14 Petabytes (10^13 TB or 10^12 B) of data need to be handled and store, and it is expected that the volume of the data will increase, according to specialists, by a factor of two after 5 years.

The size of the LHC experiments and the unprecedented scale of data resulted in the need to look at resources outside of CERN. From the beginning it was clear that to process all the data centrally at CERN was not a practical or viable solution. Instead, physicists from all over the world offered their own existing resources to be used in the experiments. Today all LHC experiments are embracing the hierarchical distribution model, according to which facilities from all around the world are putting together resources in order to provide the necessary computing power and data storage space needed for the experiments (Legrand, et al, 2005). According to this model the system is composed of an assembly of distributed computing resources, concentrated in a hierarchy of centers called Tiers, where Tier0 is CERN, Tier1s are the major computing centers which provide a safe data storage, likely in the form of a mass storage system (MSS), and Tier2s are smaller regional computing centers.

As the LHC experiments are currently well underway physicists are interested in evaluating the capability of the currently deployed (networking and computational) resources to handle the large amount of data and processing requirements. The difficulty in simulating the running conditions of the physics experiments comes from the large amount of resources involved in the analysis procedures, as specified by the computing models (Bonacorsi, et al, 2007). The evaluation of such complex simulations is hard to accomplish using existing simulators (Dias, et al, 2011). SimGrid (Casanova, et al, 2008) is a simulation toolkit that provides core functionalities for the evaluation of scheduling algorithms in distributed applications in a heterogeneous, computational Grid environment. It aims at providing the right model and level of abstraction for studying Grid-based scheduling algorithms and generates correct and accurate simulation results. GridSim (Buyya&Murshed, 2002) is a grid simulation toolkit developed to investigate effective resource allocation techniques based on computational economy. OptorSim (Cameron, et al, 2003) is a Data Grid simulator designed specifically for testing optimization techniques to access data in Grid environments. OptorSim adopts a Grid structure based on a simplification of the
architecture proposed by the EU DataGrid project. Given a replication algorithm and a Grid configuration as an input, it runs various activities over its resources.  
Such simulators were developed for particular classes of experiments. They all support, to some extent, the simulation of data transfer and replication techniques. However, they do not present general models that allow the evaluation of replication in the wider context of different architectures encountered in case of distributed systems. The simulation instruments tend to narrow the range of simulation scenarios to specific subjects, such as scheduling or data replication.  
MONARC 2, using highly advanced technologies to cope with the simulation of large amount of resources and applications, such as the ones described in the computing model of the CMS experiments (Bonacorsi, et al., 2007), is able to successfully test the running conditions of the LHC experiments.  
In this paper we present experiments designed to evaluate the capability of the current real-world distributed infrastructure to support existing physics analysis processes and the means by which the experiments bands together to meet the technical challenges posed by the storage, access and computing requirements of LHC data analysis within the CMS experiment.  
The rest of the paper is structured as follows. Section 2 gives a analysis of related work. In Section 3 we present the MONARC simulation model. Section 4 presents implementation details and results of the experiments designed to evaluate the running conditions of the CMS experiment. Finally, in Section 5 we give conclusions and present future work.  

2. MONARC SIMULATION FRAMEWORK  
MONARC 2 (Dobre&Cristea, 2007) is built based on a process-oriented approach for discrete event simulation, which is well suited to describe concurrent running programs, network traffic as well as all the stochastic arrival patterns, specific for such type of simulation. Threaded objects or "Active Objects" (having an execution thread, program counter, stack...) allow a natural way to map the specific behavior of distributed data processing into the simulation program.  

![Diagram](image)

Figure 1: The Regional center model.  
In order to provide a realistic simulation, all the components of the system and their interactions were abstracted. The chosen model is equivalent to the simulated system in all the important aspects. A first set of components was created for describing the physical resources of the distributed system under simulation. The largest one is the regional center (see Figure 1), which contains a farm of processing nodes (CPU units), database servers and mass storage units, as well as one or more local and wide area networks. Another set of components model the behavior of the applications and their interaction with users. Such components are the “Users” or “Activity” objects which are used to generate data processing jobs based on different scenarios. The job is another basic component, simulated with the aid of an active object, and scheduled for execution on a CPU unit by a “Job Scheduler” object.  
With this structure it is possible to build a wide range of models, from the very centralized to the distributed system models, with an almost arbitrary level of complexity (multiple regional centers, each with different hardware configuration and possibly different sets of replicated data).  

3. CASE STUDIES FOR THE LHC EXPERIMENTS  
The hierarchical distribution architecture is well mapped on the proposed simulation model. The simulation model allows the simulation of this type of organization. In the computing model of CMS (Bonacorsi, et al., 2007) the collections of processing nodes, data warehouses and networking entities are organized in what is called regional centers. The network simulation model allows these regional centers to be connected in arbitrary architectures, including the hierarchical model proposed by the physics experiments. Special designed job models designed to imitate the behaviour of the running LHC conditions are also integrated into the simulation model.  
These elements allow the easy construction of simulation experiments designed to test the running conditions of the LHC experiments, as envisioned in the computing models. The general concept developed by the CMS experiments is a hierarchy of distributed Regional Centers working in close coordination with the main center at CERN. This simulation study follows this concept and describes several major activities; mainly the data transfer on WAN between the Tier 0 (T0, the first processing layer is CERN) and a number of several Tier 1 (or T1) Regional Centers. The topology describing the connectivity of the Regional Centers is presented in Figure 2.  

![Diagram](image)

Figure 2: The simulation scenario resembling the CMS computing model.
We assume that the five T1 Regional Centers in Europe are connected independently, by two networks: GEANT (external to CERN) and LHCOPN (within CERN). In a simplified model this can be approximated with two “mega-routers” in which each T1 regional center is connected through a link. We also consider several transatlantic links connecting T0 with the regional centers in US, through the USLHCnet and ESNet “mega-routers”.

We first executed a series of simulation experiments designed to evaluate the function of the MONARC’s model, its capacity to handle the scenario and conditions of the running CMS experiments. These experiments are based on queuing models to evaluate the experiments.

In these experiments events produced in the LHC detector are transferred and processed at different regional centers. We first evaluated the behavior of the database as more events are concurrently served (the first series of experiments). Next we evaluated the behavior of the network as more events are concurrently transferred (the second series of experiments), and the use of uniform and non-uniform approaches to transfer the data. The analytical results allowed us to compare them against the obtained ones, thus validate the model and experiment.

In these experiments we considered a scenario consisting of several regional centers (see Figure 3) – a simplified version of the actual CMS experiments. We also evaluated the capability of the MONARC simulator to handle large scale experiments. As such, we successfully simulated for example 10000 concurrent jobs, each concurrently transferring and processing 100 events, with concurrent processors and databases.

800 MB of data are transferred, and the average CPU load at Caltech is around 80%. This is consistent with the results obtained in the simulation conducted for this case (Figure 4).

![Figure 3: The simplified scenario.](image)

![Figure 4: Validation results showing the average CPU load at Caltech, the throughput, and the total data transferred.](image)

In the first series of experiments we considered several jobs concurrently running at Caltech. Each job reads and processes 100 events from a database situated at UCSD (see Figure 3). These experiments were designed to evaluate the simulation model. The database at UCSD can serve data at a speed of 100 Mbps. The Caltech center contains 10 processing cores. The size of an event is approximately 400 KB (we actually used a normal distribution, so sizes are values in the range 300-500 KB). The processing of an event requires from 1.4 to 2.4 seconds (depending on the size).

Each job transfers an event, and processes it locally. The algorithm continues for the other 100 events. Under normal conditions, in an experiment involving 20 such jobs running concurrently, the experiment would show that approximately we continued with experiments by gradually increasing the number of jobs running concurrently within Caltech. The other parameters were kept constant. We were interested in the capability of the database server at UCSD to handle the large number of requests. As we increased the number of jobs, the load on the database also increased, up to a point where the delays started to affect the performance of the jobs. For example, Figure 5 (left) show the results for the CPU usage obtained in these experiments. The horizontal axis represents the number of concurrent jobs being used in the experiment. On the vertical axis the values represent the average CPU usage registered at Caltech. In the beginning, as more jobs run and process events concurrently, the CPU usage increases as expected.
At around 100 jobs the database reaches an internal bottleneck and starts serving events slower. Starting this point the time needed for the events to be transferred locally increases, so the overall CPU usage decreases. The results are also observed in the time needed to complete the simulation increases as more jobs rush concurrently to get the data from the database server at UCSD.

We next continued evaluating the network conditions. These experiments involved 10 databases located at UCSD, all capable to serve the events. This allow us to relieve the load on the database servers. But, as expected, in this case the network capacity becomes the limit. We executed a series of experiments by varying the number of concurrent jobs. The results in Figure 5 show the relation between the network saturation and the effect on the CPU usage, completion time and the throughput on the Internet 2 link. As the time to transfer an event increases with each experiment, due to the network link becoming a bottleneck, the CPU usage decreases. These results are sustained also by the completion and throughput values (center and right).

A comparison between the first series of experiments and the second one, for the same case of 5000 concurrent jobs running at Caltech and requesting events from the database(s) at UCSD, is presented in Figure 6. In the third series of experiments we evaluated different solutions to distribute the events. In these experiments the jobs run at Caltech and transfer events from three external regional centers (FNAL, CERN and UCSD) before processing them.

In the first series of experiments we considered a uniform distribution of events. For that each job requests a uniform event identifier with each iteration, normally distributed over the three considered centers. In the second case we considered three distinct sets of jobs that take data only from one distinct regional center from the three considered.
Figure 7: Results obtained for the throughput in case of uniform (left) and non-uniform (right) distributions.

Figure 8: Comparison between experiments using uniform (first three) and non-uniform (last three) distributions of events.
4. A SIMULATION STUDY FOR T0/T1 DATA REPLICATION & PRODUCTION ACTIVITIES

After the validation of the model we proceeded with the translation of the topology presented in Figure 2. The end-result simulation model is presented in Figure 9. It uses the entities available within the MONARC simulator. The values on the links represent the available bandwidth (in Gbps). For a better representation, this topology was simplified and the Tier2 centers were purposely ignored. For the WAN links we assumed the following RTT values (based on real-world monitored facts):

![Diagram of network topology](image)

Figure 9: The configuration used in the CMS modeling experiments.

<table>
<thead>
<tr>
<th>Center1 / T0(CERN)</th>
<th>T1</th>
<th>T1</th>
<th>T1</th>
<th>T1</th>
<th>T1</th>
<th>T1</th>
</tr>
</thead>
<tbody>
<tr>
<td>T0(CERN)</td>
<td>0</td>
<td>5</td>
<td>7</td>
<td>5</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>T1-FR</td>
<td>5</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>T1-DE</td>
<td>7</td>
<td>3</td>
<td>0</td>
<td>5</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>T1-IT</td>
<td>5</td>
<td>2</td>
<td>5</td>
<td>0</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>T1-ES</td>
<td>7</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>T1-UK</td>
<td>8</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>T1-US</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1. RTT values (ms)

Those RTT values are used in evaluating the efficiency of using the available bandwidth for “fhp” like transfers. Using this topology we simulated, alone and then combined, a number of Activities specific for Physics Data Production, as follows.

**RAW Data Replication.** From the experiment we assumed a mean rate of recording raw data equal to 200 MB/s. This information is stored in 2GB (normal distributed with 10% sd) data files. These files are replicated in a round robin manner to all 6 T1 regional centers. (The first file is sent to T1-FR, the second to T1-DE…).

**Production and DST distribution.** At T0 all raw data are processed and DST files are generated. The DST files are 10 times smaller in size than the RAW files. We considered again a normal distribution (sd 10%). The DST files created at T0 are sent to all T1 centers.

**Re-production and new DST distribution.** After a certain time the RAW data in each T1 center is re-processed and new DST data is created. Each T1 center will reprocesses 1/6 of the RAW data. The DST data generated at each regional center are sent to all others.

5. EXPERIMENTAL RESULTS

5.1. RAW Data Replication

In this experiment we simulated the RAW Data Replication activity. This activity involves the creation of RAW Data files at T0 (CERN) regional center with a mean rate of 200 MB/s. The produced RAW data is stored in 2GB size data files (where this size is normally distributed with 10% sd) and then each of this file is replicated in a round robin way to all the six T1 regional centers. This means that the first file is sent to T1-FR, the second file is sent to T1-DE, etc. Also the WAN links have 10Gbps available bandwidth (see Fig. 2).

We first executed a calibration experiment, involving ten rounds of simulating the production of RAW data in T0(CERN). The purpose was to evaluate the correctness of the experiment. For example, Figure 10 presents the results obtained for the number of active connections and how they vary as RAW data is sent to T1-FR, then to T1-DE, etc.

![Diagram of active connections](image)

Figure 10: The active connection in the calibration experiment.
Figure 11 shows the results obtained for the throughput in this experiment.

Figure 11: The used bandwidth in the calibration experiment.

In the next series of experiments we envisioned again only the RAW data processing activity running for one day (24 hours). Figure 12 shows the results obtained in case of the amount of data transferred on major links (CERN, T1-FR and T1-US). This parameter shows the quantity of data transferred through a given link from the beginning of the simulation until the present moment of time. As expected, in this case the same amount of data is transferred to all tiers involved in the experiment (such as T1-FR and T1-US).

Figure 12: The total amount of data transferred on the major links.

Figure 13 presents the distribution of the transfer time for the RAW data file in case of each Regional Center.

5.2. RAW Data Replication activity followed by Production and DST distribution

The second set of experiments involved the Production and DST Distribution activity. In this case, besides sending the RAW data as in the first experiments, DST files are produced at T0 (CERN) regional center from the recorded RAW data, which are then further distributed to all T1 regional centers.

Figure 14 presents the results for the total amount of data transferred on the major links in this test case. Figure 15 presents the results for the bandwidth used in the major networks.

Figure 13: The distribution of the transfer time for the RAW data file for each Regional Center.

Figure 14: The total amount of data transferred on the major links.

Figure 15: The total amount of data transferred on the major links.

Figure 16: The bandwidth used on the major links.

5.3. RAW Data Replication activity followed by Production and DST distribution followed by Reproduction and new DST distribution

After analyzing the results for the first two activities running concurrently we went further and simulated all the three activities (RAW Data Replication, Production and DST distribution and Re-production and new DST distribution) running in parallel.

The conditions are the same, as illustrated in Figure 2, for the links connecting the Regional Centers. We first executed a series of experiments for calibration. We considered a limited set of rounds to evaluate the correctness of the proposed experimental scenario. Figure 17 shows the
bandwidth usage on the major networks within the scenario. The network usage occasionally fills the available networks. These conclusions are based on the results in Figure 18.

![Figure 17: The bandwidth used in the major networks.](image1)

![Figure 18: The bandwidth used in the major links.](image2)

### 6. CONCLUSIONS

Large scale distributed systems are currently progressing from operational infrastructures towards environments providing many “modern” capabilities. As the LHC experiments are currently well underway physicists are interested in evaluating the capability of the currently deployed (networking and computational) resources to handle the large amount of data and processing requirements.

In this paper we presented simulation experiments designed to evaluate the capabilities of the current real-world distributed infrastructure to support existing physics analysis processes and the means by which the experiments bands together to meet the technical challenges posed by the storage, access and computing requirements of LHC data analysis within the CMS experiment. MONARC, with its simulation model, is capable of simulating such large-scale (in terms of the number of hosts, resources, jobs, etc) experiments. The obtained results demonstrate its capability to correctly model solutions for large scale distributed systems, and the capability of the model to pinpoint problems in the simulated environments. They demonstrate that the LHC infrastructures are well suited to support the data processes envisioned by the CMS computing model.

### ACKNOWLEDGEMENTS

The research presented in this paper is supported by national project: "TRANSYS – Models and Techniques for Traffic Optimizing in Urban Environments", Contract No. 4/28.07.2010, Project CNCSIS-PN-II-PU-PD ID: 238. The work has been co-funded by the Sectoral Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/89/1.5/S/62557.

### REFERENCES


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


### BIOGRAPHY

**CIPRIAN DOBRE** PhD, is lecturer with the Computer Science and Engineering Department of the University Politehnica of Bucharest. The main fields of expertise are Grid Computing, Monitoring and Control of Distributed Systems, Modeling and Simulation, Advanced Networking Architectures, Parallel and Distributed Algoritms. Ciprian Dobre is a member of the RoGRID (Romanian GRID) consortium and is involved in a number of national projects (CNCSIS, GridMOIS, MediGRID, PEGAF) and international projects (MonALISA, MONARC, VNIC, VNSim, EGEE, SEE-GRID, EU-NCIT). His research activities were awarded with the Innovations in Networking Award for Experimental Applications in 2008 by the Corporation for Education Network Initiatives (CENIC).
NS-3 Simulation and Management of WiMAX and LTE Networks with NSDL

Eduardo M.D. Marques, Jorge J.F. Sousa, Paulo N.M. Sampaio
Madeira Interactive Technologies Institute (M-ITI)
University of Madeira (UMa)
9000-390 Funchal - Madeira, Portugal
emails: emarques@uma.pt, jose.jorge.sousa@gmail.com, psampaio@uma.pt

KEYWORDS
Network description, WiMAX, LTE, simulation, ns-3

ABSTRACT

The Worldwide Interoperability for Microwave Access (WiMAX) and Long term Evolution (LTE) technologies promise some new possibilities to the interconnection of computers. The large bandwidth, distances supported and the use of wireless channels are important characteristics to new networks environments. The tools to the simulation and test of these technologies are still limited and, most of the times, are not simple to use. The purpose of this paper is to introduce the Network Scenario Description Language (NSDL) framework, proposed for the integration of different modeling, monitoring and simulation tools which can be applied for types of network scenarios and technologies. In particular, this paper also aims at describing all the process and steps for applying the NSDL framework in order to carry out simulations with WiMAX and LTE wireless networks.

INTRODUCTION

The design and management of communication networks are related to different activities including the definition of network requirements, network planning, deployment, operation, maintenance, and, at last, monitoring and optimization. These activities are carried out within the network life cycle, and they can be further refined into more specific tasks. Due the heterogeneity of tasks and skills required within each activity there are several tools and platforms to support network management and its life cycle. This limitation in the network management increases the redundant work and, for each people within the process, there is a different perspective for the same network scenario.

The modeling, simulation and deployment of broadband technologies for providing mobile communication using the IEEE 802.16 - Worldwide Interoperability for Microwave Access (WiMAX) (IEEE802.16 2004, Hasan 2007) and the 3GPP Third Generation Partnership Project Long term Evolution (LTE) (Motorola 2007) are examples of scenarios that can be addressed. These technologies aim at providing mobile voice, video and data services by promoting low cost deployment and service models through Internet friendly architectures and protocols. In both case, different models, approaches and tools have been proposed in order to optimize and exploit the advantages of these technologies through the simulation (Piro et al. 2010, Erbele 2011, Ball et al. 2008).

Some of the existing solutions for the design and implementation of WiMAX and LTE technologies provide an approach for some specific issues and problems, limited to some specific domains, and, most of the times, having a scarce or incomplete documentation. If all these solutions could be applied jointly in a coordinated way, they would provide a solid and helpful environment for optimizing the management of WiMAX and LTE networks. Nevertheless, the data formats used in general by these existing solutions are very distinct and, most of the times, incompatible.

In order to provide a generic solution for promoting interoperability among different WiMAX and LTE network management tools, this paper presents a framework relying on the Network Scenarios Description Language (NSDL) (Marques and Sampaio 2010), which has been proposed as a common solution that can be applied to assist network managers with the optimization of the network during its life cycle. Figure 1 illustrates the layered organization of several components of the NSDL framework. The top and bottom layers represent the existing networks tools. The top layer represents the management tools to provide the modeling, monitoring and visualization of networks (e.g., GUIs such as, topology generators, operation and failures monitoring, statistics and results, etc.). The bottom layer represents the network analysis tools, such as network simulators, analysis tools, security evaluation, etc. Actually, some of the existing management platforms support both bottom and upper layers; however, most of them are purpose-oriented and are present only in one of the layers.

The NSDL block in the framework stands as a middleware layer to connect the previous referred layers, i.e., different networks tools. The GUIs may, e.g., read and write the created network scenarios and, through specialized Application Programming Interfaces (APIs), the bottom layer tools are invoked to execute some op-
erations over the network. In this paper we illustrate the utilization of the NSDL framework to provide the network modeling and simulation of WiMAX and LTE networks.

The remainder of paper is organized as follows: NSDL language is introduced in the next section. Section ‘Describing WiMAX and LTE Networks with NSDL’ introduces the concept of profile in NSDL, called an NS-3 profile. In the section ‘Mapping NSDL to NS-3’ demonstrates how a translation from an NSDL description to an NS-3 script can be obtained. Section ‘Case Study’ addresses a brief example related to the existing WiMAX and LTE modules for NS-3. At last, section ‘Conclusions’ presents the main results of this work.

NETWORK SCENARIOS DESCRIPTION LANGUAGE

The purpose of the Network Scenario Description Language (NSDL) is to provide a vocabulary and a set of rules, both able to support the description of wired and wireless data networks and the information of the contexts, or domains, where those networks are used or evaluated.

In NSDL not only the network topology with its objects and characteristics are described, but also the different context of use or evaluation that may occur over that network. This separation is a new approach to networks description. So, NSDL allows, first, the description of the objects (and components) of the network, their relations and properties, and, second, it also allows the description of the network in different utilization contexts, such as simulation, management, and others.

The principles followed in the design of NSDL were (1) simplicity, which means the language has to be simple and clear to an application or tool and also to a human user; (2) definition of multiple abstraction levels, i.e., accommodate the specification of both simple and/or more complex descriptions of the network scenario objects and its parameters; and (3) extensibility, implying that new objects and parameters can always be incorporated in future descriptions.

The main goal of NSDL is to help network administrators throughout the network life cycle and this goal is achieved by providing a rich language to describe the network objects and, also very important, the description of the several network context of use, which we refer as scenarios. The underlying language in NSDL is the eXtensible Markup Language (XML) (XML 2011) due its richness and flexibility. Indeed, XML provides the specification of clear definitions and has a set of available tools for validation and transformation.

An NSDL representation is an XML file with two basic elements: network and scenarios (Figure 2): Network and Scenarios. The Network element contains the description of a network identifying its objects and its parameters. The Scenarios element may contain several descriptions, each one referring to a specific use, or context, to that network.

The Network element is composed of Templates, Objects and Views. Since the Objects element contains the description of the network topology, it is the main component of the language. This element is composed of nodes, links and domains. Some other important, but not mandatory elements are Templates and Views. The Templates element is important to simplify the description of similar objects. The Views element is a mechanism applied to group network objects to be used in the scenarios.

In the Scenarios element, two elements are introduced: Visualization and Simulation. The Visualization element provides additional information to enrich the network description, such as objects positioning in the GUIs and graphical data. All the parameters needed to implement a particular simulation over the network using a generic or particular simulation tool is defined in the Simulation element.

A last organization element present in the NSDL language is the profile. Briefly, an NSDL profile is a set of NSDL objects and/or scenarios with some relation among them. Profiles were defined in order to simplify the management of the several objects already defined, and the other objects still to be defined in the future, and; to make clearer the use of NSDL, providing a standard mechanism to select only the needed objects in a particular network domain scope. A last important advantage of the profiles is the strengthening of the interoperability purpose of NSDL. The existence of profiles will guide users to the use of the same objects, promoting the re-use or improvement of already defined elements, as an alternative to the definition of new network elements.

The types of profiles defined for NSDL were the Objects Profiles and the Tools Profiles. The first type of profile enables the user to create groups of network elements, of any kind. The second type of profiles, the Tools Profiles, adds to the network elements, the scenarios supported by a particular tool. Figure 3 illustrates some profiles. In the next section a case study is presented to illustrate the utilization of the NSDL framework focusing in some wireless scenarios.
DEscribing WiMAX and LTE Networks with NSDL

As introduced in the previous section, an NSDL tool profile consists of a set of objects and scenarios, with specific lexical and semantic rules, in order to accommodate the language specification for a network tool. The tool profiles already defined for NSDL are related to two simulation platforms: Network Simulator 2 (ns-2) (ns 2011) and Network Simulator 3 (NS-3) (Henderson et al. 2008). When a network scenario is described using NSDL, it is possible to translate these xml descriptions to a specific output script (e.g., C++ (Stroustrup 1986) for NS-3) and, consequently, that allow researchers to execute that scenario on the respective tool related to the defined NSDL profile. The NSDL profile for NS-3 includes the objects and the scenarios presented in Figure 3. A set of objects is illustrated as Network Objects, composed of Generic Objects, Base TCP/IP Objects and Wireless Objects. The following NSDL Object profiles were defined:

- **Generic Objects**: highest level of abstraction for the more common objects present in network scenarios;
- **Base TCP/IP Objects**: contains the TCP/IP objects, such as, IP, TCP and UDP protocols, and;
- **Wireless Objects**: contains the wireless network domains: Wireless Fidelity, WiMAX and LTE.

An NSDL tool profile to be complete needs, besides the objects profiles, one or more scenarios and, in the case of NS-3, the selected and defined scenarios were Visualization and Simulation.

A special focus was given to the wireless approach when defining the NS-3 profile for NSDL, based on the increased focus given by the NS-3 developers on these solutions (Fawooq and Tuletti 2009, Piro et al. 2010, Krill and Boyko 2010). Therefore, this case illustrates how NSDL describes specifically the WiMAX and LTE objects.

In a WiMAX environment the Base Station (BS) and Subscribed Station (SS) roles determine that a BS function is to transmit and manage exchanged flows between Ss and an SS is either a source or a destination of the flows, so, the description should reflect those characteristics. Figure 4 is the NSDL description for both BS and SS used in a WiMAX environment.

The BS and Ss in our scenarios are simple nodes. The distinction between them occurs in the interface element (<wlan0216>) and more precisely the <type> attribute. The BS <ipv4> element holds the IPv4 specifications to configure a LAN Network, using the attributes <net.address> and <net.mask>. The <interface.id> references the interface (<wlan0216>), which is optional if there is only one, but mandatory in case of having several interfaces. Thus, this ipv4 configuration will be performed by the BS, assigning IPv4 and the respective mask to the remaining nodes (i.e., the SS), as required by the NS-3 simulator. Optionally, a modulation type Quadrature Amplitude Modulation (QAM) 16/12 (<modulation.type>) for signal is set on SS's. Similarly, Figure 5 describes Enhanced NodeB (eNB) and User Equipment (UE) configurations used in LTE networks.

Figure 4: NSDL partial description of the WiMax BS (node4) and SS (node5) configuration

Figure 5: NSDL partial description of the LTE eNB (node4) and UE (node5) configuration

Herein it is possible to observe the similarity between both BS/eNB and SS/UE specifications in terms of NSDL structure. Again for eNB, IPv4 is set for LAN purposes and <lte> element only needs to know which kind of station has been dealt with (configured with parameter <type>). In the case of UE cell (node5), a reference to the associated eNB is mandatory in order to register this UE cell to its corresponding eNB.

This section presented the very basic configuration to the main nodes that will be object of study in the following section. The next section also demonstrates how an NSDL description (as the ones depicted in Figures 4 and 5) can result in an NS-3 C++ code.

Mapping NSDL to NS-3

The mapping step in NSDL is the file generation of a particular script, C++ in the case of NS-3, making use of sequential template calls by an eXtensible Stylesheet Language Transformation (XSLT) (XSLT 2011) file that helps to carry out this action. The purpose of the template calls is to generate a correct sequence for a typical NS-3 script which is given by the following structure:

1. Inclusion of main NS-3 libraries and definition of global variables (e.g., Realtime);
2. Instantiation of Helpers (e.g., WimazHelper);
3. Definition of typical applications ports;
4. Node(s) instantiation(s) (e.g., a computer);
5. Internet installation and routing (e.g., AODV);
6. Link(s) creation (e.g., P2PChannel);
7. Interface(s) definition (e.g., EnbNetDevice);
8. Node(s) Addressing (IPV4 and IPV6);
9. Application(s) and PacketSink definition and scheduled events (e.g., Start time);
10. General Simulation configuration (e.g., Simulation Start time, simulator destructor);
11. Visualization settings (e.g., area and mobility model aspects), and;
12. Output file(s) creation (e.g., Trace files).

Exceptional cases are not included in this sequence in order to keep it minimally generic. For instance, Service Flows and Radio Bearers used in WIMAX and LTE environments respectively are called after step 11, following the structure provided by NS-3 examples (can be found in folder /src/<cmodel>/examples of NS-3 installation folder).

The translation of an NSDL file, such as those presented in Figures 4 and 5, will yield as output the code presented in Figures 6 and 7, respectively. Not referred, but present, in all mapping process is the validation of the scenarios to ensure their correctness in the context of a particular tool. Some considerations needed to be highlighted in this output code are:

- IPv4 and IPv6 containers are instantiated to support both kinds of addressing;
- According to Figures 7 and 8, IPv4 is only specified on BS/eNB for LAN sub netting;
- The ‘(...)’ token between excerpts of code means the eventual omission of generated code, and;
- In the NS-3 profile, only ConstantPositionMobilityModel is foreseen to attach on scenario node(s).

The next section will provide an example of a wireless scenario created in a graphical user interface and tested over the NSDL framework.

CASE STUDY
In order to provide a clear example of the application of NSDL framework to the wireless domain, two network scenarios were defined with different types of nodes using the configurations depicted in both Figures 6 and 7 to reflect WiMAX and LTE environments, respectively. Figure 8 illustrates the adopted topology, which was the same for both scenarios. These figures also present which applications were used and their respective role (i.e. if server or client). Note that in Figure 8, the topology consists of 9 nodes and the central node plays a role of BS/eNB and the remaining nodes represent SSs/UEs to generate and receive data flows.

Specifically for WiMAX scenarios, links are implemented using Orthogonal frequency-division multiplexing (OFDM) simple data access (NS-3 class SimpleOfdmWimazChannel) with default definitions (i.e., without propagation loss model). Contrarily, links used in the LTE scenario consist of a Single Spectrum model

```cpp
// *** Node node4 /**/
Pt<Node> node4 = CreateObject<Node>();
// *** Node node5 /**/
Pt<Node> node5 = CreateObject<Node>();
(...)

// Interfase Setting /**/
Ipfv4InterfaceContainer ifwmax4;
Ipfv4InterfaceContainer ifwimaz4;
WimazHelper::SchedulerType wimazScheduler =
WimazHelper::SCHED_TYPE_SIMPLE;
NetDeviceContainer node4d = wimaz.Install(Ipfv4InterfaceContainer ifwimaz4,
WimazHelper::DEVICE_TYPE_BASE_STATION,
WimazHelper::SIMPL_PHY_TYPE_OFDM,
ifwimaz4); // NS-3 installation folder

// Network NS -> Network NS
dnode5d.Set(Ipfv4Address(Ipv4Address(255, 255, 255, 0), Ipv4Address(255, 255, 255, 0)));

// Protocol creation /**/
...
```

Figure 6: Partial C++ generated code containing the BS (node4) and SS (node5) information
Figure 8: Topology used to create WiMAX and LTE network simulation scenarios in NS-3 using the NSDL framework

using a single frequency for data transmission (using class SingleModelSpectrumChannel) with default configuration as well. Flows generated consist in 2 FTP, 1 CBR and 1 Pareto. FTP and CBR flows use Constant-Variable seed whereas Pareto flow uses Pareto Variable with ’1’ as mean value and value ‘2’ for shape. Each flow is managed either by a Service Flow (on WiMAX) or a Radio Bearer (on LTE).

Due to the NS-3 limitations presented in both WiMAX and LTE modules (Farooq and Turlleti 2009, Piro et al. 2010) there are some aspects needed to be highlighted as well:

- For WiMAX scenario, only 1 Service Flow (flow manager) peer SS is allowed;
- For LTE scenario, currently only downlink scheduler is implemented which means no uplink flow to eNB is possible with the present module.

The described network was successfully translated to an NS-3 script and the simulation scenario was correctly executed, providing us with some data to compare WiMAX and LTE technologies. Also, variations of the topology and objects parameters were tested with success, reinforcing the validating of the translation libraries. Eventually, some errors could arise from untested parameters or particular topologies, however we made sure most scenarios, or similar scenarios, found in the works listed in the references was and could be translated. Since the main focus of this paper is to introduce NSDL and the NSDL to NS-3 translation for WiMAX and LTE, the results of the simulation are not further exploited. Moreover, according to previous referred limitations, no easy or feasible comparison between both technologies can be achieved with the current NS-3 modules.

In the next, and last, section we present some conclusions about the use of NSDL to define NS-3 wireless simulation scenarios.

CONCLUSION

The purpose of this work is to illustrate the deployment of the NSDL framework in order to support the life cycle of wireless network scenarios (modeling, simulation and analysis) based on existing simulators, such as NS-3. The process of obtaining the C++ script respecting the structure of an NS-3 simulation was explained.

The simulation process used the NSDL framework and the steps for the automated generation of NS-3 simulation code was presented. A first advantage of the framework is the simpler modeling environment for the construction of wireless network scenarios. An XML network description could be achieved using a GUI and a C++ script could be automatically generated. Another important advantage of this framework is its interoperability capabilities that are enabled by the possibility of executing simulation scenarios in different tools. Some initial scenarios can be already generated for both ns-2 and NS-3.

As for future works, the NSDL framework also has been extended for different types of network scenarios, including Wireless Sensor Networks and virtualization scenarios.

REFERENCES


TRANSPORT SIMULATION
A THIRTY-TWO THOUSAND VARIABLE NONLINEAR TRANSPORTATION PROBLEM

William Conley
Austin E. Cofrin School of Business
480B Wood Hall
University of Wisconsin-Green Bay
Green Bay, Wisconsin 54311-7001
U.S.A.
Conleyw@uwgb.edu

KEYWORDS
Transportation, shipping, multi stage Monte Carlo Optimization, computer speeds, complex systems

ABSTRACT

Large businesses can have a variety of transportation problems of considerable size. Classical mathematics has helped with the linear version of delivering a product in bulk from N locations to M destinations at minimum cost as long as the costs associated with the N times M variables are linear. However, there are usually returns to scale and discounts for shipping in quantity making most of these problems nonlinear and much more difficult for theoretical mathematics. However, the power and capacity of 21st century desk top computers has advanced to the point where nonlinear transportation problems of size can be addressed with simulation techniques such as multi stage Monte Carlo optimization (MSMCO). Presented here is a thirty-two thousand variable nonlinear transportation problem solved with MSMCO in a goal programming setting on a desk top computer. This is then compared with a thirty-two variable such problem solved in the 1980s with MSMCO on a main frame computer. The speed and affordability of modern desk top PC computers make simulation applied to optimization viable as never before. Comparisons along this line will be addressed.

INTRODUCTION

Books like (Mizrahi and Sullivan 1993) and (Barnett and Ziegler, 1994) present standard explanations of the linear programming simplex technique (Dantzig 1963) developed by George Dantzig and others in the mid-20th century. This was and is a great breakthrough in linear mathematics. It can be used to solve the classical linear transportation problem when used with a computer to do the attendant large scale simplex tableaux calculations. However, Jan Tinbergen (Nobel Prize winner) used to lecture the science and business world about not linearizing essentially nonlinear relationships (Tinbergen 1978) just so one could use the well-developed linear theories in classical mathematics. Therefore, presented here will be a large nonlinear transportation problem after a section on the MSMCO statistical optimization introduces the solution technique used here. Then comparisons between 20th century and 21st century solution techniques and their viability and versatility will be mentioned.

MSMCO AND STATISTICAL OPTIMIZATION

Multi Stage Monte Carlo Optimization (MSMCO) shows its versatility on environmental problems as in (Conley, 2008) and solving nonlinear systems of equations (Wong, 1996). Additionally, MSMCO is useful in solving shortest route transportation problems producing the exact solution to well-known test problems in the mathematics literature. Two such studies are test problem 30 and test problem 32 (Lawler, Lenstra, Rinnooy-Kan, and Shmoys, 1984), (Krolak, Felts and Marble, 1970) and (Conley, 1991).

However, in the featured shipping problem here, Multi Stage Monte Carlo Optimization merely approximates the solution to an extremely difficult problem, where the true optimal is virtually impossible to find because of the size and complexity of the feasible solution space.

Statistical optimization or MSMCO treats each optimization problem as having a sampling distribution consisting of all of the feasible solutions of the optimization problem at hand. Then n dimensional rectangles of ever decreasing size (inside the feasible solution space) track and close in on the ever improving answers until the true optimal (or a useful approximate solution) is found. It is a simulation based solution based solution technique for the computer age. Generally, it is a good idea to close in slowly and draw many sample answers at each stage of the simulation. Typically somewhere in between 5,000 and 50,000 feasible solutions are drawn at each stage. An example would be to have 50 stages (with ever smaller dimensions for each subsequent “n dimensional rectangle”). Therefore, if 10,000 samples answers were drawn at each stage this MSMCO simulation run would require that 500,000 total feasible solutions be looked at as a simulation tracked and closed in on the optimal solution region.
The first stage of the MSMCO simulation usually covers the whole width of the feasible solution space. Then in stage 2, perhaps all of the width, height, and length dimensions (in n dimensional space) could be divided by 2 to reduce the search region. Stage 3 might divide the dimensions by 2 again and so on until at stage 50 the search region is reduced by a factor of 2 to the 50th power. Another commonly used reduction quotient for each stage is 1.41 or about the square route of 2. That 1.41 was used on the 50 stage simulation for our featured 32,000 variable example that is presented here.

A further illustration of the geometry and statistics ideas that can make Multi Stage Monte Carlo Optimization effective would be the following packaging example. Some countries and their national post offices or private stores sell cardboard boxes for shipping gifts to friends or other important items through the mail. Suppose someone purchased 13 of these boxes (all of different sizes) and put the smallest box inside the second smallest box. Then these two boxes were put inside the third smallest box and this arrangement of the three smallest boxes were put inside the fourth smallest box and so on until all of the boxes were put inside the largest box. Then opening the largest box would be like stage 1 of an MSMCO simulation. Then opening the second largest box would be like stage 2. Then opening the third largest box would be like stage 3 of an MSMCO simulation. By the time the thirteenth, and smallest box, was opened, the optimal solution, to this so called three dimensional simulation, would be inside this thirteenth tiny cardboard box.

The geometry is appealing in three dimensions. However, how well will this work on functions of hundreds or thousands of variables? The answer is that it will work quite well if many thousands of feasible solutions are drawn at each stage of the simulation. Many more stages are used as the problems grow in size and complexity. Let us look at a large problem where the geometry cannot be pictured, but the algebraic closing in can be programmed into the computer.

**A LARGE NONLINEAR PROBLEM**

A manufacturer has recently produced 125000, 126000, 127000, 128000, 129000, 130000, 131000, 132000, 133000, 134000, 135000, 136000, 137000, 138000, 139000, and 140000 units of its product at its sixteen factories. It wants to ship these 2,120,000 units to its 2000 best customers during this short term excess demand period at a cost of less than 75,000,000 Euros, which is about the cost it incurred in a similar supply and demand situation fairly recently.

Specifically, the valued 2000 customers would like to each receive $1506+.2j$ units for $j=1, 2, 3, \ldots 2000$ respectively for customers 1 through 2000 (or a total of 3,412,400 units).

There are substantial discounts for shipping in quantity for each factory to each customer. Let $x(i,j)$ be the amount shipped from factory $i$ to customer $j$. Then the cost equation (representing the discounts for shipping in quantity) is

$$C = \sum_{i=1}^{16} \sum_{j=1}^{2000} 0.0555(3^{i+j} - (x(i,j) - (1.0 - 0.0001(3^{i+j})))$$

(Note that the 32,000 fractional powers reflect the nonlinearities that occur in practical real world problems when there are discounts for shipping a product in quantity.)

Therefore, management sets $C$ to 70,000,000 Euros in an attempt to save at least 5 million Euros over the previous similar situation. So we use multi stage Monte Carlo optimization (MSMCO) to minimize the sum of the absolute values of the differences between the left and right hand side of the seventeen equations down to zero (or a close approximation) subject to the 2000 customers less than or equal constraints after making one slight transformation in the cost equation.

The difficulty is that the cost equation has a constant of 70,000,000 while the 16 factory equations have constants in the 125,000 to 140,000 range. Therefore, we multiply both sides of the cost equation by .0025 (note .0025 times 70 million is 175,000) to bring its constant more in line with the other 16 equation constants. Otherwise, the cost equation will be so dominant that MSMCO will not work solving all 17 equations fairly equally. We purposely leave the transformed cost equation (175,000) a little higher than the 16 factory equations in the hopes that if it is not solved exactly, its error will be on the low side, saving even more than 5 million Euros.

Specifically, our MSMCO simulation optimization here is a 50 stage Monte Carlo computer run drawing 45,000 feasible solutions (at every stage). Each stage is over an ever decreasing in size search region following the trail of better and better answers until it funnels in to the answer of

$$x_{11} = \ldots$$
$$x_{15} = \ldots$$
$$x_{16} = \ldots$$

Specifically, our MSMCO simulation optimization here is a 50 stage Monte Carlo computer run drawing 45,000 feasible solutions (at every stage). Each stage is over an ever decreasing in size search region following the trail of better and better answers until it funnels in to the answer of

$$x_{11}^{\text{min}} = \ldots$$
$$x_{15}^{\text{min}} = \ldots$$
$$x_{16}^{\text{min}} = \ldots$$
The complete answer file (all 32,000 variable values) revealed all 16 factory equations having errors of less than one unit. The largest factory equation error was .07031. The average of the 16 equation errors was .01172.

The cost equation came in lower than the artificial 175000 bound. So once it was converted to Euros by dividing by .0025 the total cost was 68,646,800 or a savings of 6,353,200 Euros when compared with the previous 75,000,000 Euro old standard.

This computer run took 253 minutes on a desk top PC computer.

TRANSPORTATION AND SCIENCE PROBLEMS THEN AND NOW

A 32 variable nonlinear transportation problem was worked on in (Conley 1987) as part of a series of chemistry problems (including a nonlinear chemical yield equation, a balancing chemical equations example, a chemical engineering problem and a linear chemical yield equation). It involved shipping a chemical in bulk from four storage tanks (locations) to eight customers (destinations) with discounts for shipping the chemicals in larger quantities. Supply exactly equaled demand (unlike our hypothetical 32,000 variable problem presented here). The goal was to drive the cost equation down as low as possible, while meeting the eight customers’ demands. Therefore, the attendant 32 variable 13 nonlinear equation system was solved using multi stage Monte Carlo optimization (MSMCO) on a mainframe computer (costing about a million dollars in today's adjusted for inflation currency). However, the 32,000 variable nonlinear transportation problem worked on here (about a quarter of a century later) was also solved with MSMCO, using an inexpensive desk top PC computer. The 32 variable nonlinear problem had a fairly large feasible solution space, but the size of the current 32,000 variable nonlinear problem (a thousand times more variables) feasible solution space stagers the imagination. Yet a large multinational company could very well have a shipping problem involving 16 locations (factories or warehouses) to 2000 destinations (customers or stores).

Additionally, it should be pointed out that it took 253 minutes of computer time to produce a useful answer. However, a large company would likely have much more powerful computers available than just a desk top PC, for its logistics department and managers who would be involved in trying to reduce shipping costs. They could then keep lowering the cost equation constant (in this goal programming setting) and rerunning the MSMCO simulation repeatedly to try to lower the cost even more as long as the factory equations were still being solved.

Note that the 32,000 variable transportation problem here had demand exceeding supply, while the 32 variable one (Conley 1987) had supply equaling demand. Additionally (Conley 2010), presents a 900 variable transportation problem where supply exceeds demand. All three of these situations are possible in the world of big business. Also, the 900 and 32,000 variable systems tried to minimize the sum of the absolute values of the differences between the right and left hand sides of the equations, where the 32 variable problem tried to minimize the maximum equation error at each stage. Both transformations (and combinations of them too) can be useful on a variety of problems that occur in science and business.

Note that our 32,000 variable problem had 2000 major less than or equal constraints. Therefore, once the inner loop of the MSMCO program calculated the 32,000 random x(i,j) variable values, all constraints were checked. If any constraint right hand side constant was exceeded, all 32,000 variables were divided by the largest excess ratio, shrinking them all back inside the feasible solution space. This is discussed on pages 228-235 of (Conley 1981), although thirty years ago it was limited to an eight variable problem. Today we can use it on a 32,000 variable problem with 2000 major constraints.

MULTI STAGE MONTE CARLO SIMULATION

Again, Monte Carlo optimization consists of a random search of the feasible solution space for the optimal solution or a useful approximate answer. Multi Stage Monte Carlo optimization views this as stage one. Then in stage two of the simulation, centered about this best answer so far, another “random” search in a slightly reduced region takes place. Its best answer is stored. This is followed by a stage three search centered about the best answer produced in stage three, in a further slightly reduced region. This is repeated for as many stages as necessary to solve the problem or obtain a useful approximate solution.

The general purpose nature of this multi stage Monte Carlo optimization (MSMCO) has allowed it to be used on a variety of applications including (Wong 1996) on a difficult nonlinear system from theoretical mathematics and (Conley 1993) on the difficult shortest route problems (that also occur in transportation). Additionally, (Conley 2008) used it on cost control and the science of pollution reduction and also (Conley 2007) with the CTSP correlation coefficient that picks up nonlinear multivariate relationships.

CONCLUSION

A 32,000 variable nonlinear transportation problem where supply temporarily does not equal demand was presented
and solved with MSMCO in a goal programming context that produced a cost reduction of over 6 million Euros.

This problem was then compared with a 32 variable nonlinear transportation problem (where supply equaled demand) which was solved with MSMCO in the 1980s on a main frame computer. The comparison was made mainly to point out that as desk top PCs (and of course main frame computers too) become cheaper, quicker, and evermore ubiquitous, MSMCO and all simulation techniques become more viable and useful in engineering, science, statistics, economics and business, among other fields.

REFERENCES


BIOGRAPHY

WILLIAM CONLEY received a B.A. in mathematics (with honors) from Albion College in 1970, an M.A. in mathematics from Western Michigan University in 1971, an M.Sc. in statistics in 1973 and a Ph.D. in mathematics - computer statistics from the University of Windsor in 1976. He has taught mathematics, statistics, and computer programming in universities for over 30 years. He is currently a professor emeritus of Business Administration and Statistics at the University of Wisconsin at Green Bay. The developer of multi stage Monte Carlo optimization and the CTSP multivariate correlation statistics, he is the author of five books and more than 200 publications world-wide. He is a member of the American Chemical Society, a fellow in the Institution of Electronic and Telecommunication Engineers and a senior member of the Society for Computer Simulation.
A SIMULATION APPROACH TO SUPPORT THE DESIGN OF FLEXIBLE TRANSPORT SYSTEMS

Ana Dias
José Telhada
Maria Sameiro Carvalho
University of Minho, Systems and Production Department, Campus Gualtar, 4710-057 Braga, Portugal
E-mail: anacecidias@hotmail.com
{sameiro, telhada}@dps.uminho.pt

KEYWORDS
Flexible transport systems, demand responsive transport, transportation, decision support system, simulation.

ABSTRACT
Rural areas are characterized by low levels of population density and complex mobility patterns. Conventional transport services have shown to be very inefficient and ineffective in these environments and the provision of traditional public transport services in these areas presents, very often, low levels of service. In this context, an alternative solution - a Demand Responsive Transport (DRT) system - has already been adopted in several countries. Although the literature reports some successful cases of DRT systems, these flexible transportation systems have still some issues to be addressed. A main question that still requires additional interest of researchers is the assessment of their overall sustainability, since its implementation, in general, requires a strong technological component and the integration of several technologies. Additionally, they are highly dependent on the correct calibration of some organizational and functional parameters. In this paper, a comprehensive framework to support decision-makers in the design and planning of flexible transportation systems is proposed. The developed approach allows the simulation of different scenarios corresponding to different design alternative solutions. Accurate estimation of their global impact can be obtained providing effective support in the design and operational stages of a DRT system implementation.

INTRODUCTION
Traditionally, public transport systems in rural areas have been based on static services: fixed routes, stops and schedules. However, the low levels of density observed in some of these areas leads to very low levels of vehicles occupancy and high levels of inefficiency. Also, vehicles used are frequently the oldest of operator fleet, frequencies are low, some schedules are omitted and, therefore, such a transportation system does not attract many potential users. In these circumstances, rural areas tend to have their mobility limited, increased social exclusion of some population sectors, such as children, elderly and/or mobility impaired people, forcing people to use private cars.

The adoption of a flexible transportation system such a Demand Responsive Transport (DRT) which can meet users’ mobility needs by adopting flexible routes and stops constitute a very interesting and attractive alternative.

Authors are unanimous in stressing that the success of DRT systems depends on the use of intelligence solutions to process trip requests, to optimize routes and schedules in order to respond in real time to users mobility needs. According to the literature, one of the gaps associated to DRT systems (already implemented) is a framework that could provide an integrated decision support system to help decision makers on devising intelligent strategic solutions at the design phase.

In this research project, a simulation approach to allow the decision support system to design and plan a DRT system in rural areas has been developed. The objective is to reproduce and test different decision-making alternatives in order to assess, in advance, the quality of alternative design scenarios or management strategies.

In fact, such a tool can provide what-if analyses required to achieve better planning decisions and will allow evaluating operating strategies prior to the implementation of such as a complex system as it is the case of a DRT system. Furthermore, the tool will ultimately assure the adoption of a sustainable DRT system, by properly allocate supply to estimated demand levels and patterns, taking into account financial, economic and social decision criteria.

The remainder of the paper is organized as follows: next section presents a brief literature review on flexible transportation systems in order to highlight the main design issues concerning DRT systems. The following section a Decision Support System based on a simulation approach to support strategic decisions concerning the design and planning of DRT systems is proposed. Then, the work implemented so far is described and performance measures are discussed. Finally, the last section presents the main conclusions and final
considerations regarding future developments.

STATE OF THE ART

Flexible transportation systems, in particular, Demand Responsive Transport (DRT) systems have been adopted over the last decades, as reported in some studies (Brake et al. 2004; Mulley and Nelson 2009). Increasing interest have been devoted to DRT as a mean to combat social exclusion. They are an attractive solution, because they offer a user-friendly answer to passengers needs and overcome some shortcomings of traditional transport services in areas with widely dispersed trip patterns. These systems provide transport on demand, using flexible schedules and routes to pick up and drop off users as required.

In general, trip requests are made by telephone dialing directly to a travel dispatch center (TDC), during a predefined time window; alternatively, there are cases where users can also use a messaging system (SMS) or they can use a web portal (Oliveira et al. 2011). Trip requests are then stored in a database system, which holds all the relevant data concerning the transportation network. The TDC coordinates a fleet of vehicles with communication technologies such as on-board integrated GPS, continuous GPRS connection to TDC. A heterogeneous fleet of vehicles is frequently available: buses, mini buses, taxis supplied by a variety of providers (taxi owners, bus operators, community transport, etc.).

A DRT system can receive trip requests either for an immediate service or as an advanced reservation. It should have the capacity to organize routes and schedules to accommodate trip requests, aiming to respond in real time to user’s mobility needs. Its implementation typically involves the use of information and communication technologies (Frosini et al. 2004) as shown in Figure 1.

The development of Intelligent Transport Systems (ITS) tools, as well as the availability of mobile communications, has allowed new public transport service options to be developed whereby the service is more responsive to customer demand in terms of time and space (Mageean and Nelson 2003; Ambrosino et al. 2004; Brake et al. 2004).

![Figure 1 Elements of a demand responsive transport system (Oliveira 2009).](image)

In practice, different levels of flexibility can be adopted and customer demand determines the route and vehicle used at varying timescales prior to travel as illustrated in

Figure 2. This figure summarizes the range of options for a flexible transportation system configuration.

![Figure 2 Different flexibility levels, adapted from (Brake et al. 2007).](image)

It is essential to realize the role of each flexible typology of transport service as part of the overall public transport system. A DRT system can be flexible in terms of route, vehicle allocation and operator, type of payment, passenger category, level of automation and level of integration with the transportation network. The flexibility of each element can vary along a continuum of demand responsiveness from services where all variables are established and fixed at a considerable time before the operation, to services where variables are determined close to the time of operation (Brake et al. 2007). Also, transport services can be operated on their own or integrated with traditional transportation systems, acting as feeder services for buses or rail services.

Different types of contributions can be found in the literature concerning flexible transportation systems and they can be grouped in three categories: general articles, case studies, and analytical models. Savelbergh (1995), Desaulniers & Villeneuve (2000) and Cordeau & Laporte (2007) have proposed interesting review articles on flexible transportation systems.

General papers address essential aspects and decisions of DTR systems (e.g. Giannopoulos 2004; Enoch et al. 2006; Mulley and Nelson 2009).

Case studies papers report experiences of DRT implementation and discuss main results achieved benefits, problems and limitations of adopted solutions (e.g. SAMPLUS 2000; Gray et al. 2001; McDonagh 2006).

Finally, modeling papers, by far, the most numerous, propose mathematical models to address some of the most complex problems faced in the operational management of DRT systems such as network routing and scheduling, demand forecasting models, simulation models, etc (e.g. Parragh et al. 2008; Quadrifoglio and Li 2009).

The DRT operation planning problem can be treated as an optimization problem where there is a number of vehicles and drivers which pick up passengers from a
point and delivery them to another within a time interval with the objective of minimizing total system costs or maximizing customers level of service, subject to several constraints. This issue is considered a vehicle routing problem with pickup and delivery (VRPPD) (Xiang et al. 2006). A specific case of VRPPD is the dial-a-real problem (DARP). The DARP have been focused on human resources and in time windows. Bolin et al. (1983) developed one of the first studies on the characterization of route planning problems of goods and people.

From some DRT systems reported in literature its implementation obeys to several conditions, encompassing high dynamic levels of both planning and coordination processes. The use of modern information and communication technologies, transport telematics/Intelligent Transport Systems or even, Flexible Transportation Services (FTS) allied to adequate strategy planning services has been pointed out as the solution to improve the costs-effective performance of DRT services (Mulley and Nelson 2009).

A conclusion that can be draw from the literature devoted to DRT systems is that these transportation systems are supposed to be used for a considerable time during which many parameters can change, in particular, demand patterns or mobility needs. It involves several decisions, some of them strategic but also tactical and operational. By nature, strategic decision should last for a substantial amount of time due to the large investments normally associated with this type of decisions. Strategic decisions are concerned with the definition of the most adequate design which involves many choices associated with the level of responsiveness to adopt (route, vehicles, operators, level of integration with PT, level of automation, etc.) These decisions have a critical impact on the level of resources to be allocated and, therefore, in its overall operational costs.

Although DRT is a very interesting solution, its use has been limited due to the relatively high costs of operation as a result of high levels of resources (vehicles and drivers) and to the relatively high complexity associated to planning and operational issues. Additionally other issues have been pointed out by several authors (Brake et al. 2004), which can have a significant impact on its implementation. Such as, the lack of legal and regulatory framework in most countries to accommodate flexible transport services; the technological issues including the selection of algorithms to provide operational solutions; the sustainability of DRT services and the lack of support to select the most appropriate service and system configuration.

The use of simulation approaches have been reported in the literature to support some design decisions whenever uncertainties are associated with some parameters. It allows to take into account different configuration designs helping the decision making process.

Wilson et al. (1970) pioneered the use of simulation and were followed by several authors using simulation to forecast and analyze different scenarios previous the implementation of a transport system. Some studies (Feuerstein and Stougie 2001; Bailey and Clark 1987) have investigated the performance impact of using different fleet sizes on a dial-a-ride system.

Fu (2002) develops a simulation model to evaluate the potential effects of the latest progress in information technologies on dial-a-ride paratransit system. Diana (2006) appraised the effectiveness of a DRT scheduling algorithm by Diana and Dessouky (2004) of dynamic parameters such as proportion of real time requests and interval between call-in time and requested pick-up time.

Quadrigfoglio and Dessouky (2007) tested the efficiency of the insertion heuristic scheduling algorithm for MAST systems; they also used simulation to perform a sensitivity analysis of the accomplishment of a MAST system varying the shape of its service area. Quadrigfoglio et al. (2008) used the simulation methods to investigate the effect of using a zoning versus a no zoning strategy and time window settings.

Dias et al. (2011) proposed a framework that will allow achieving the most adequate configuration by evaluating different design strategies prior to the implementation of such a complex system as it the case of a DRT system.

Following the approach proposed by Dias et al (2011), an integrated approach devised to address some of the main difficulties associated to the design and management of a DRT system was developed and will be described in more detail. A decision support system (DSS) incorporating mathematical models (optimization, simulation and statistical methods) and integrating some important characteristics of the real context such as the area characteristics, population demographics and legal and regulatory framework will provide an effective support in the decision making process. Additionally this DSS will incorporate a comprehensive evaluation module able to assess system performance providing valuable information on systems sustainability. Next section will discuss transportation performance indicators frameworks.

### PERFORMANCE INDICATORS

In order to evaluate the sustainability of the system it is necessary to define and use performance measures or indicators.

The performance evaluation module was intended to be as much comprehensive as possible, including a wide set of indicators from three main dimensions: social, economic and environmental. A research was carried out to identify which performance indicators are more appropriated to assess the quality of transportation systems. Based on the literature on transport performance systems (NCHRP 2006), there are a large number of measures that can be grouped in four categories: preservation of assets; mobility and accessibility; operations and maintenance and safety.
Several performance indicators can be produced in order to provide insight on systems operations, such as, total generalized costs of trip plan; medium delay of each vehicle; mean users delay time; mean waiting and travel time; vehicles utilization rate; capacity and availability rate; requests not satisfied; level of service; incident response.

According to the literature, efforts have been made in order to identify the potential benefits with the least impact on costs, but the convergence of methods for costs and benefits assessment turns difficult to measure or evaluate impacts in monetary terms. This is a complex task and additional research is needed to develop models and better evaluate the socio-economic development of ITS. Furthermore, some authors refer the absence of sufficient information to make a quantitative analysis of transport services. This information can be obtained through surveys of potential users of the service.

Particularly, in DRT projects, social assessment stands out as fundamental, addressing the efficiency for all stakeholders, including aspects like customer acceptance, impact on traffic, environmental and socio-economic impacts, and other externalities. There are also some environmental indicators, but their analysis is out of the scope of this project. But indeed, that smaller vehicles at least produce less CO2 emissions than the oldest and biggest conventional public transport. Additional research is still required to complete this evaluation module.

**THE DECISION SUPPORT SYSTEM**

The main objective of the decision support system developed is to configure a DRT system that incorporates a high level of flexibility and that responds in real-time to users demand taking into account the real context: socio-economic, demographic, legal, etc.

Approaches proposed so far focus on some particular aspects of design process, such as the fleet size or, the most numerous ones, the definition of schedules and routes allowed, as the result of explicit requests for travel (Oliveira et al., 2011). Therefore, they cannot provide additional information for the overall system evaluation.

The conceptual model proposed is illustrated in Figure 3: characteristics of the study area, including demand, and supply for transport services must be collected and stored in a database which is the basis of the main information system that constitutes the DSS. A trip demand generator will produce trip requests by applying a probabilistic model for destination choice, based on the socio-economic and demographic characteristics of the residents (from Census), and the locations and characteristics of the main attraction poles. A simulator module will determine dynamically the most appropriate (rational) set of routes and schedules to satisfy requested trips, and will reproduce all movements of users and vehicles according to predefined system operating parameters. Routing and scheduling solutions are obtained by automatically selecting the most appropriate solution method, depending on the typology and dimension of the problems/situations, the computational time performance of the algorithms and the time available to obtain the required solution. Various alternative and complementary solution methods (or algorithms) were incorporated in the routing and scheduling sub-module, ranging from an exact dynamic programming method to basic construction heuristics (ex., Clark & Wright savings method) and more advanced heuristics.

The simulator is one of the key elements of the DRT framework since it will reproduce the real system behavior (supply and demand for transport and different operation choices). The DSS, by integrating all the main design, planning and operational issues will allow analyzing and evaluating different scenarios for the DRT system integrating technical and non-technical cross-functional cooperation in order to produce better services for users and more efficient and sustainable operations for operators.

A set of discrete alternative scenarios can be produced associated with different levels of resources or different operating rules. Sensitivity analysis for small changes in parameters can be carried out to evaluate impacts on general systems performance. Examples of tests that can be performed are: spots of population concentration within counties; different routes and stops in a particular area; DRT system integration with regular transport service; flexibility of services as a function of economic efficiency, costs effectiveness and resources availability.

A main feature of the system is the routing planning module. Both advanced booking and on-line trip request are accepted and, therefore, a highly dynamic routing and scheduling approach is highly desirable. A set of exact methods and heuristics approaches have been provided and an intelligent routine will select the most appropriated algorithm according with planning context (for instance, an exact solution can provide optimal solutions but may not achieve the required solution in the time window available) or with the type of problem to solve. Information concerning routes and schedules in produced and it is available both to the drivers and to users (Oliveira 2009). Whenever a trip request cannot be satisfied due to lack of operational capacity (no vehicles available or no slack time), users are notified.

Additional analysis is then performed to assess solution viability and encompassing several dimensions such as: technical, financial and economical. Performance measurement is essential to monitor progress toward a result or goal. It is also a process of gathering information to make well-informed decisions. The result of the assessment process will provide guidelines and the required feedback to adjust system resources and operating parameters as illustrated by the feedback arrows in Figure 3.

Furthermore, beside the route optimization module, the decision support system proposed include other characteristics, such as, request management, drivers and
customers communication systems and integration with a Geographic Information Systems (GIS) tool.

**WORK DEVELOPED**

As it was already stated, our aim is to develop an intelligence solution for flexible transportation decision makers.

Figure 4 sows in greater detail the main module of the DSS: the DRT simulator. Based on information of the real area to be analyzed such as demand patterns (in particular, form this data element a survey must be carried out to devise origin destination trip matrices, for each time period) and the road network, trip requests are generated and routes and schedules are produced, that will allow to simulate different alternatives in order to assess, in advance, the performance of a wide range of scenarios or management strategies. The production of routes and schedules follow some pre-defined objectives established by the user such as: distance minimization, minimization of the number of vehicles; minimum user delay; minimum time, minimum costs (a generalized distance/time cost function or multi-objective function can also be used), etc. Constrains associated with physical resources availability and system operating parameters must be also set to configure the operational context. Additionally, associated to the production of each solution, several performance indicators are produced to allow decision maker to assess solution’s quality.

The simulation model by incorporating analytical models and using data on the transportation system will be able to produce scheduling and routing plans. The analytical tool to be used in each situation depends on the typology of the problem to address and the time available to calculate the solution.

**Figure 4 Simulation model of the DSS.**

Implemented situations were:

- “many-to-one” or “one-to-many” where there are many origins and a single destination or one origin and several destinations (e.g. school service or enterprises) – the solution method adopted is the Clark & Wright heuristic. Figure 5 illustrates the interface of a solution obtained for an instance of twenty-two requests with three vehicles available inside of a region. The green point is the pickup point and the red delivery points. The heuristic objective is to determine a good solution, that is a route or a set routes associated to a total distance as shorter as possible (but not the necessarily the shortest of them) by using a minimum of vehicles from a fleet of vehicles with different capacities. The route(s) initiate and finish at the same spot, never overcharge the vehicles maximum capacities.

**Figure 5 Many to one.**

- The vehicle only travel inside an area. It can be solved with an exact dynamic programming method because is a very restricted situation and computational time is more efficient comparing to other exact methods. Figure 6 illustrates the solution obtained for an instance of seven trip requests and with one vehicle available.
Besides the optimization routing module, the application is being enriched by the incorporating of several others modules developed or under development, such as, requests management, drivers’ communication, clients’ communication (including a web portal) and several database and GIS tools. The storage and flow of the data are fundamental to maintain the global efficiency of the system.

Figure 8 represents the proposal architecture for the DSS information system, showing the information flows among models and other elements of the system.

Figure 8 Informational architecture of a DSS system.

This information system will be able to combine data with analytical tools in order to put available information that will support top management strategic decisions. However, this tool is automatic and will be able to be also used as an operational tool to plan schedules on a daily bases, whenever system configuration is already set.

A data warehouse is used to assemble, organize and link all the relevant data: transportation network, trip requests (demand), user’s data, available resources, network, scheduling plans, etc.

Trip requests are added to the system using a trip request generator which reproduces user’s mobility needs for a given period (when using this approach in the design stage) or using a web application designed for this purpose. An additional facility was incorporated by using Google Maps allowing a visual interface both for input as for output (results) purposes. With the Google maps tool it is possible for the user to see the origin and destination points, the shorter route, the mean travel time and distance and the schedule of the trip.

There is an advantage in the use of GIS technology by integrating the Google Maps service. Google maps allows the representation of geo-referenced information on a map in a very user-friendly way and is a free service.

Google maps needs JavaScript technology that has to be supported by browsers to represent the wanted information as a route design (Figure 9).

The software application developed has the ability to choose the best methods, for each particular case, allowing planning efficiency.

Results presented before referred to a study of a case carried out in a small village in the north of Portugal, Terras de Bouro. At this stage only preliminary tests have been carried out.

A main question that requires additional research interest is to assess their overall sustainability, since its implementation, in general, requires a strong technological component and the integration of several technologies. Additionally, there are also some issues concerning the costs and benefits of DRT system that require further analyses and still need to be addressed, in order to evaluate the transport system viability and sustainability. These issues are currently being studied by the authors of this paper.

Specificities of theses transportation systems must also be taken into account: spatial aspects of public transport such as the characteristics of the local population, transport network, the patterns of commuters, and the framework within which the system works, determine the demand and operational scale of system and, as such, can affect performance and efficiency. Globally, these projects aim to improve customer satisfaction, with regard to their mobility.
CONCLUSIONS

Prior to the implementation of a DRT system there are many issues to be properly addressed at the design and planning phases. Many DRT projects have been implemented world-wide without taken adequate care of such issues, and therefore some related failures (that have led to system re-engineering or even project withdrawal) have been reported. In addition, there is currently a lack of comprehensive methodologies to address the problem of designing and planning such systems. In order to achieve a successful DRT service it is important to develop a framework that provides an integrated DSS to enable decision makers to perform systematic analysis leading to intelligent strategic solutions.

This project includes the simulation model incorporating more efficient routing and scheduling algorithms, using different solution methods (exact algorithms, heuristics, meta-heuristics), and provide the system with cleverness to adopt the most appropriate models according with the context. Furthermore, this simulation tool integrates GIS in order to enhance the graphical displaying of the solutions produced, and allow further statistical analyses of spatio-temporal indicators. A case study in a small rural area of the north of Portugal is use to validate the framework.

ACKNOWLEDGEMENTS

The authors acknowledge the financial support provided by the European Commission and the Portuguese Government through the research grant FCT/TRA/72871-2006.

REFERENCES


AUTHORS BIOGRAPHY

MARIA DO SAMEIRO CARVALHO is graduated in Computer and Systems Engineering (University of Minho – Portugal, 1986) and holds a MSC degree in Transportation Planning and Engineering (University of Leeds - UK, 1990) and a PhD degree in Transport Planning and Engineering (University of Leeds - UK, 1996). Her main research interests are in Transportation, Supply Chain Management and Operational Research.

JOSÉ TELHADA holds a PhD degree in Transport Planning and Engineering (ITS-University of Leeds – UK, 2001). His main research interests are Operational Research and Logistics applied to transportation systems.

ANA DIAS is graduated in Mathematics (University of Minho (2004) and Master in Industrial Engineering (University do Minho, 2008). At the present she works as research student at University of Minho.
MATHEMATICAL MODEL AND INDUSTRIAL APPLICATION OF
FOUR INDEX TRANSPORTATION PROBLEM

Michel Gourgand
Tuyet-Hoa Pham
Alain Tanguy
LIMOS, CNRS UMR 6158
Blaise Pascal University, Cezeaux Campus
63173 Aubière Cedex, France
E-mail: [gourgand, pham, tanguy]@isima.fr

KEYWORDS
Four index problem, solution existence condition, degeneration, potential method, mathematical model

ABSTRACT

Due to the importance of freight transport in the panel of the logistic cost, our research is conducted to optimize transportation solution on cost criterion. Its aim is to find a solution on the minimum transportation cost for the four index transportation problem (4ITP). It may be degenerated and may not verify the solution existence condition (SEC). We develop a resolution method based on the coupling of mathematical model and an algorithm. This will be implemented in the production management system used by industrial enterprise AAS which is the first rank subcontractor for Airbus.

INTRODUCTION

The study of road freight transport includes all the methods and activities which aim is to coordinate the physical flows by optimizing all intervening in each link of the chain. Since the first classical problem, called the two index transportation problem (2ITP), was developed in 1941 by Hitchcock, the research in this area has achieved satisfactory results on various extensions. There are two main directions of research: theoretical and operational research. On the theoretical point of view, the research on transportation problem is developed at the macro level with the n index problem. On the operational angle, the results are still limited because the application of transportation models to the reality is complex. Consequently, the remaining problem is to find methods to solve the particular problems.

In this paper, we are interested in the four index transportation problem, called 4ITP which four indexes: origin, destination, goods type and truck type, one of extension cases of the Hitchcock’s problem. The 4ITP includes the exchange of one or several goods types that are transported by truck types in a single link between origins to destinations. The studied problem is presented in the next section. Subsequently, we give a brief literature review and explain the relationship with our problem. The resolution algorithm is also presented. Finally, we elaborate a program based on our exact method and execute it on the artificial database. This is the foundation for the implementation of this algorithm with really large database in industry.

PROBLEM PRESENTATION

The 4ITP formalization is based on the n index transportation problem (Ninh 1979). We consider the following data:
m origin nodes O_i (i = 1...m). The offer of goods at this node is a_i (i = 1...m).
p goods types. The total quantity of goods type S_j at all m nodes is y_j (k = 1...p).
n destination nodes D_j (j = 1...n). The demand of goods at this node is b_j (j = 1...n).
q truck types. The total quantity of goods that the truck type H_l can transport is c_l (l = 1...q).
c_{ijkl} unit transportation cost for a unit of goods type S_j (k = 1...p) that is transported from node O_i (i = 1...m) to node D_j (j = 1...n) by truck type H_l (l = 1...q).
The hypothesis must be fixed by inhibition of transport in the opposite direction or from destination nodes to origin nodes. The following parameters must be known to begin to solve the 4ITP:
a_i quantity of transported goods from origin node O_i (i = 1...m).
b_j quantity of goods to be transported to destination node D_j (j = 1...n).
y_j total quantity of transported goods type S_j (k = 1...p).
d_l total quantity of goods that the truck type H_l (l = 1...q) can be transported.
c_{ijkl} unit transportation cost for a unit of goods type S_j (k = 1...p) that is transported from node O_i (i = 1...m) to node D_j (j = 1...n) by truck type H_l (l = 1...q).
\alpha_i > 0 (i = 1...m); \beta_j > 0 (j = 1...n); \gamma_k > 0 (k = 1...p); \delta_l > 0 (l = 1...q); c_{ijkl} > 0 (i = 1...m, j = 1...n, k = 1...p, l = 1...q)

The variables of 4ITP are
\x_{ijkl} quantity of goods type S_j (k = 1...p) is transported from node O_i (i = 1...m) to node D_j (j = 1...n) by the truck type H_l (l = 1...q) in the solution to establish.
\x_{ijkl} ≥ 0 (i = 1...m, j = 1...n, k = 1...p, l = 1...q).
The 4ITP becomes: Determine the variables
minL(X) = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} \sum_{l=1}^{q} c_{ijkl} \cdot \x_{ijkl}\ (1)
subject to
\sum_{j=1}^{n} \sum_{k=1}^{p} \x_{ijkl} = a_i (i = 1...m)\ (2)
\sum_{i=1}^{m} \sum_{k=1}^{p} \x_{ijkl} = b_j (j = 1...n)\ (3)
\sum_{l=1}^{q} \sum_{j=1}^{n} \sum_{k=1}^{p} \x_{ijkl} = y_j (k = 1...p)\ (4)
\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} \sum_{l=1}^{q} \x_{ijkl} = d_l (l = 1...q)\ (5)
The constraint (2) shows that all goods types that are transported from origin node \( O_i \) to all destination nodes by all truck types are equal to the offer \( a_i \).

The constraint (3) shows that all goods types that are transported from all origin nodes to destination node \( D_j \) by all truck types are equal to the demand \( b_j \).

The constraint (4) shows that the goods type \( S_k \) that is transported from all origin nodes to all destination nodes by all truck types is equal to the quantity \( q \).

The constraint (5) shows that all goods types that are transported from all origin nodes to all destination nodes by truck type \( H_l \) are equal to the quantity \( d_l \).

The 41TP was defined and a part of this problem was solved in 1979 (Ninh 1979). Therefore, which particular cases of the transportation problem were presented and which used resolution methods were presented in previous researches?

**LITERATURE REVIEW**

The 41TP is an extension case of Hitchcock’s problem. This problem, also called the two index problem, is the foundation to research the general problem. The first resolution method that was defined by Kantorovich and Gavorin, Russian researchers, is the potential method. They use a rectangle: one side represents origins. They divided this side in \( m \) segments and the other side represents destination. They also divided it in \( n \) segments and each cell of the table represents an arc (j) in which, they cost \( c_{ij} \) and goods \( x_{ij} \). Then, they calculate the potentials and test the optimality criterion. If the solution is not optimal, improve this solution to obtain the other better solution. Up to now, this method is the most commonly used method in research and teaching (Ninh 1979; Dubeau and Guèye 2008; Gourgang et al. 2011).

Another extension case of classical problem is to transform the extended two index problems into Hitchcock’s problem. There are two typical problems in this domain: continuous transportation problem (Kangbo 1998) and fuzzy integer transportation problem (Chanas and Kuchta 1998).

In fact, to solve the first problem, Kangbo transformed the continuous problem into a discrete problem by taking a sufficiently large rectangle \( A \times B \subseteq X \times Y \). He divided \( A \) in \( m \) small segments \( A_i = \{1 \ldots m\} \) and \( B \) in \( n \) small segments \( B_j = \{1 \ldots n\} \). The segment \( A_i \) was considered as the origin \( A_i \) and the segment \( B_j \) considered as the destination \( B_j \). On the small rectangle \( A_i \times B_j \), giving the value \( c_{ij} \), an approximate value of \( c(x,y) \) with \((x,y) \in (A_i \times B_j)\). Thus, Kangbo transformed his problem into Hitchcock’s problem. About the fuzzy integer problem, it was solved by Chanas and Kuchta in 1998. They transformed this problem into a different problem, and then, they proposed a method to solve it by solving several Hitchcock’s problems.

The other researchers solve the three index transportation problem (31TP) with a method based on the potential method of the Hitchcock’s problem. They utilize a parallelepiped: one edge represents origin \( O_i \) \((i=\{1 \ldots m\})\), the other one represents destination \( D_j \) \((j=\{1 \ldots n\})\), and a third edge represents the goods type \( S_k \) or the truck type \( H_l \) \((k=\{1 \ldots p\})\).

Therefore, this problem is known as “solid transportation problem” (Haley 1963). Remark that there are two 31TP according the constraints if they take the matrix sum on one or two indexes. Similarly to 21TP, the general direction to solve the extended problems is to transform them into the classical 31TP and then, they use the potential method extension (Jimenez and Verdegay 1998).

With the use of the parallelepiped, we see that the constraints of three index problem can be represented under a sequence of rectangles representing the constraints of two indexes with a variable is kept, the two other variables change and superpose on each other. Hence, if they continue to extend toward this direction, they obtain the development of \( n \) index transportation problem. A part of this general problem was solved in 1979 (Ninh 1979). P.X.Ninh stated and demonstrated the necessary condition and sufficient condition so that the general problem has the optimal solution. Being different from the precedent thinking, he did not use the \( n \) dimensional super box to solve it, he gave the resolution method on the plan. His method was inspired by the used potential method in the Hitchcock’s problem.

In general, these \( n \) index transportation problems just do value theoretically. There is only one problem contributing significantly in economic terms. The problem is presented as follows:

\[
\min L(X) = \sum_{k=1}^{m} \sum_{k=1}^{n} c_{i,k} x_{i,k} \quad \text{subject to}
\]

\[
\sum_{i=1}^{m} x_{i,k} = a_{i,k} \quad l_k = 1 \ldots n_k \quad (6)
\]

\[
\sum_{k=1}^{n} x_{k,i} = b_{i,j} \quad l_j = 1 \ldots n_j \quad (7)
\]

\[
\sum_{i=1}^{m} \sum_{k=1}^{n} x_{i,k} = a_{i,j} \quad l_j = 1 \ldots n_j \quad (8)
\]

\[
\sum_{k=1}^{n} \sum_{i=1}^{m} x_{i,k} = b_{i,j} \quad l_k = 1 \ldots n_k \quad (9)
\]

\[
a_{i,j} > 0, a_{i,j} > 0, \ldots, a_{i,j} > 0, c_{i,j} \geq 0 \quad \text{are known and deterministic.}
\]

The multi-index transportation problem is presented in another form (Queyranne and Spieskma 1997). This is a \( k \) dimensional super box \( A \) in which the \( r^k \) dimension \( A(r) \) is composed of \( n \) indexes. The segment \( A \) is a super box with \( k \) dimensions that is attached to the segment \( i \) of the edge \( A_i \). Thus, the nature of this problem is the \( n \) index problem (6,9) but it is presented under the geometrical angle. However, being different from Ninh’s exact method, these researchers used the approached method by dividing two steps. First, they fixed one index \( i \) and solved \( k-1 \) two index problems from \( A_{i} \) to \( A_{i} \) with all indexes \( r \neq i \). Thus, the solution of \( k \) index problem is induced by \( (k-1) \) solutions that have been just found. Second, they solved \( h=1 \ldots k \) problems and then chose the best solution. The observed size of problem is still small \((i=1 \ldots 3, i=1 \ldots 3, i=1 \ldots 4)\).

For the transportation problem (6)…(9), P.X.Ninh presented and demonstrated the theorem of the necessary and sufficient condition so that the problem has had the solution (solution existence condition - SEC). By replacing \( n=2 \) in this result, he found the SEC for the 21TP that existed previously. By replacing \( n=4 \), P.X.Ninh obtained the SEC for the 41TP (1…5) as follow:

**Theorem:** The necessary and sufficient condition so that the 41TP has a solution:
\[ \sum_{i=1}^{m_i} a_i = \sum_{j=1}^{n_j} \beta_j = \sum_{k=1}^{n_k} \gamma_k = \sum_{l=1}^{n_l} \delta_l \quad (10) \]

If the 4TP has solution, this problem will certainly have the optimal solution.

Thus, based on this result, P.X.Ninh solved the 4TP where the SEC was satisfied and the problem was not degenerated by proposing an extension of the potential method. Remark that there are three 4TP according to the constraints if they take the matrix sum on one, two or three indexes. The solved problem is the one with matrix sum of three indexes. Following this success, a condition of limited transportation capacity on the road was added, capacitated four index transportation problem C4ITP (Zitouni 1994). In 2010, based on a large database (771x1500) Djamel realized a comparative numerical study between three resolution methods to C4ITP: two classical methods (simplex method and inside point method) and Zitouni’s method, a method was built on Ninh’s method improvement. Based on the criterion of the smallest number of iterations to obtain the optimal solution, the obtained result demonstrated that the Zitouni’s method is the most favourable (Djamel 2010).

Hence, the general 4TP is one of transportation problem that extremely attracts the concerns of researchers. It fits the demand to the shuttle type in the companies that can use various trucks to transport goods from the manufacture (origin) to the deposit (destination). However, until now, the research results are relatively limited with only a solved case where the SEC is satisfied and the 4TP is not degenerated. In 2011, we present a resolution method under exact mathematical angle to solve the remaining cases being untreated, where the 4TP is degenerated and the SEC is not satisfied, in the first steps (Gourgand et al.2011). How to present the resolution?

**RESOLUTION METHOD**

We pose and demonstrate the sufficient condition so that the 4TP is degenerated. Based on this condition, we introduce our general algorithm, a potential method extension, to solve the 4TP where the SEC is satisfied or not and the problem is degenerated or not.

**General resolution method**

**Degeneration condition elaboration**

We elaborate the sufficient condition so that the 4TP is degenerated with a constraint where the SEC is satisfied.

1. **General form**

   **Theorem:** If there are \( m_1, n_1, p_1, q_1 \) \( (0 < m_1 < m, 0 < n_1 < n, 0 < p_1 < p, 0 < q_1 < q) \) that:
   
   \[ \sum_{i=1}^{m_i} a_i = \sum_{j=1}^{n_j} \beta_j = \sum_{k=1}^{n_k} \gamma_k = \sum_{l=1}^{n_l} \delta_l \]

   then, the 4TP will be degenerated.

   We permute \( m_1 \) equations which right side is these \( a_i \), \( n_1 \) equations which right side is these \( \beta_j \), \( p_1 \) equations which right side is these \( \gamma_k \), \( q_1 \) equations which right side is these \( \delta_l \) at the first lines in each group. Thus, in order to simplify but not to diminish the generality of the problem, we present the sufficient condition under the reduced form.

2. **Reduced form**

   **Theorem:** If there are numbers \( m_1, n_1, p_1, q_1 \) \( (0 < m_1 < m, 0 < n_1 < n, 0 < p_1 < p, 0 < q_1 < q) \) that:
   
   \[ \sum_{i=1}^{m_i} a_i = \sum_{j=1}^{n_j} \beta_j = \sum_{k=1}^{n_k} \gamma_k = \sum_{l=1}^{n_l} \delta_l \]

   then, the 4TP will be degenerated.

Consequently, the general form can be transformed into the reduced form and visa versa by the permutation of any equations in the same group.

**Resolution method elaboration**

Based on the result of the sufficient condition so that the 4TP is degenerated, we notice that the cause of degeneration of 4TP can be defined as the transportation problem divided into two independent sub problems and each sub problem responds to the SEC. Thus, to eliminate the degeneration, we have to modify the initial problem so that it cannot be divided by two independent sub problems, i.e. each sub problem does not respond to SEC. However, in fact the research of numbers \( m_1, n_1, p_1, q_1 \) is relatively difficult although if we know they exist. Therefore, we propose a practical algorithm to recognize a degenerated problem and eliminate the degeneration for two cases: the degeneration appears in the elaboration process of the first solution and the degeneration appears in the process of solution improvement to obtain the better solution.

To the first case, during the process of determining a variable \( x_{ijkl} \) \( > 0 \) of the first solution, if we lose more than one equation in the constraints, the solution is degenerated. Supposing that we lose two equations in the constraints, there will be two equations which value at right side becomes zero. Thus, to remove the degeneration, we keep the value zero on the right side of an equation and for the value zero on the right side of another equation, we add \( \varepsilon \) (\( \varepsilon > 0 \) and infinitesimal) and simultaneously we subtract \( \varepsilon \) in the right side of an equation that belongs to this group to respond to the SEC.

To the second case, supposing that during the transformation of a non optimal solution to search the other better solution, we find a solution with two variables \( x_{ijkl} \) becoming zero, this means that the number of positive variables of this solution is less than \( m+n+p+q-3 \). Thus, it is a degenerated solution. To remove the degeneration, we add \( \varepsilon \) (\( \varepsilon > 0 \) and infinitesimal) in the second variable zero. Thus, the SEC always fits but the amount of four members \( a_i, \beta_j, \gamma_k, \delta_l \) is added to \( \varepsilon \). In mathematical term, the new problem is different from the initial problem because the total amount of transported goods increases more \( \varepsilon \) but in practice, it does not influence on the actual operating result. In the case where there are more than two variables \( x_{ijkl} = 0 \), we add \( \varepsilon_1, \varepsilon_2, ... \) to these variables by the above principle to eliminate the degeneration. After obtaining the optimal solution, we replace all these \( \varepsilon \) with zero and we have the result for the initial 4TP.

On the other hand, if a transportation problem does not respond to the SEC, it is not possible to solve it. Therefore, we modify this problem to create a new problem coming up to our expectations. The modification principle is to add fictitious nodes and parameters (index i: fictitious node \( m+1 \); index j: fictitious node \( n+1 \); index k: fictitious parameter...
$p+1$; index $l$: fictitious parameter $q+1$) to create equality between the four amounts. The transportation cost is considered as null: from a fictitious node to all destination nodes, from all origin nodes to a fictitious destination node, to transport the fictitious goods or using by a fictitious truck.

**Resolution algorithm**

Based on the resolution method, the steps of our general algorithm to solve the 4ITP in all particular cases are presented as follow:

**Step 1**

We test the SEC (the relationship 10)
If yes then go to the step 3, else go to the step 2.

**Step 2**

Search $\max (\sum_{i=1}^{n} \alpha_i, \sum_{j=1}^{n} \beta_j, \sum_{k=1}^{p} \gamma_k, \sum_{l=1}^{q} \delta_l)$

Supposing that

$$\max (\sum_{i=1}^{n} \alpha_i, \sum_{j=1}^{n} \beta_j, \sum_{k=1}^{p} \gamma_k, \sum_{l=1}^{q} \delta_l) = \sum_{l=1}^{q} \delta_l = A$$

Add a fictitious node $(m+1)$ with the quantity $\alpha_{m+1} = A - \sum_{i=1}^{n} \alpha_i$. Therefore, $\sum_{i=1}^{n+1} \alpha_i = A - \sum_{l=1}^{q} \delta_l$

Give $c_{m+1,l} = 0, \forall j, k, l, q$.

Similarly, execute for any amount that is less than A to obtain the relationship $\sum \alpha_i = \sum \beta_j = \sum \gamma_k = \sum \delta_l$

Go to the step 3.

**Step 3**

Search the first solution.

1. Loop 1

Determine $x_{ijlk}\alpha_{ijlk} = \min (\alpha_{ij}, \beta_{ij}, \gamma_{ij}, \delta_{ij})$

- If $\min (\alpha_{ij}, \beta_{ij}, \gamma_{ij}, \delta_{ij}) = \alpha_{ij}$

Give $x_{ijlk} = \alpha_{ij}$

Thus, $x_{ijlk} = \alpha_{ij}$ with $j = 1 ... n, k = 1 ... p, l = 1 ... q, (ijkl) \neq (ljk0)$

The right part in the constraints becomes

$$\alpha_{ij}^{(2)} = \alpha_i$$

$$\beta_{ij}^{(2)} = \beta_j$$

$$\gamma_{ij}^{(2)} = \gamma_k$$

$$\delta_{ij}^{(2)} = \delta_l$$

- If $\min (\alpha_{ij}, \beta_{ij}, \gamma_{ij}, \delta_{ij}) = \beta_{ij}$

Give $x_{ijlk} = \beta_{ij}$

Thus, $x_{ijlk} = \beta_{ij}$ with $j = 1 ... n, i \neq j, k = 1 ... p, l = 1 ... q, (ijkl) \neq (ljk0)$

We similarly execute to the other cases.

2. Loop 2

We determine $x_{ijlk}\alpha_{ijlk} = \min (\alpha_{ij}, \beta_{ij}, \gamma_{ij}, \delta_{ij})$

While the number of the first solution is less than $(m+n+p+q-3)$ positive variables do to execute the loop 1.

We remark that the variables $\{x_{m+1,lk} = \psi(ijkl), x_{ij,lk,m+1} = \psi(ijk), x_{ij,lk,m+1} = \psi(ijk)\}$ will be determined in the final sequence.

Go to the step 4.

**Step 4**

Control the optimality of the solution $X$.

1. Determine the potentials $u_{ij}, v_{ij}, w_{ij}^{(i)}, u_{ij}^{(i)} (i = 1 ... m, j = 1 ... n, k = 1 ... p, l = 1 ... q, i, j, k, l = 1 ... p, l = 1 ... q)$ for $u_{ij} + u_{ij}^{(i)} + u_{ij}^{(i)} + u_{ij}^{(i)} + u_{ij}^{(i)} + u_{ij}^{(i)} = c_{ijkl}$ with $x_{ijkl} > 0$

2. Compute the elements

$$\Delta_{ijkl} = c_{ijkl} - (u_{ij} + u_{ij}^{(i)} + u_{ij}^{(i)} + u_{ij}^{(i)} + u_{ij}^{(i)} + u_{ij}^{(i)}) \forall x_{ijkl} = 0$$

Test $\forall \Delta_{ijkl} \geq 0$ (12)

- If the condition (12) is satisfied, go to the step 6.
- else the solution $X$ is not optimal go to the step 5.

**Step 5**

Modify the solution $X$ to find the other better solution.

1. Determine the elements

Determine the element $\Delta_{ijkl} = \min \Delta_{ijkl}$

If there are several $\Delta_{ijkl}$ get the minimum value, give any $\Delta_{ijkl}$ of these $\Delta_{ijkl}$.

Determine the element $t_{ijkl}$ for

$$\sum_{ijkl} t_{ijkl} = 0$$

The right part of this constraint is the vector 0.

$p_{ijkl}$ is the coefficient vector of variables in the constraint. It is the column vector which has four digits 1 at the line $i, m+i, m+n+k, m+n+p+l$, the others are zero.

2. Elaborate a new solution

Determine $\theta$ for $\theta = \min \frac{\Delta_{ijkl}}{t_{ijkl}}$ with $t_{ijkl} < 0$

$x_{ij}^{(i)} = x_{ij} + \theta, t_{ijkl} = 0$ with $x_{ijkl} > 0$

$x_{ij}^{(i)} = \theta, t_{ijkl} = 0$ with $x_{ijkl} = 0$ and $(ijkl) \neq (i^{(i)}j^{(i)}k^{(i)}l^{(i)})$

We test, among these found variables $x_{ijkl}$, there is just one being zero.

- If yes then the solution is not degenerated.
- else keep the value of a variable and suppose that the value of the others are respectively $x_{ij}, x_{il}, x_{lk}, x_{kl}$.

Return to the step 4.

**Step 6**

- Replace $\epsilon$ with zero in the found variables.
- Remove the variables that correspond to the added fictitious nodes and parameters in the step 2.

Finally, we obtain the optimal solution of the initial problem.

**APPLICATION**

Based on the algorithm, we give numerical examples on small and medium database. The aim of this section is to test the method before an implementation in industry. Firstly, by using the previous algorithm for the 4ITP and the FORTRAN 95 program, we present the obtained optimal
goods distribution planning on the data of a very simple model with 4 origins, 2 destinations, 6 goods types, 4 truck types and 144 unit costs.

<table>
<thead>
<tr>
<th>Planning</th>
<th>Unit: ton</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>Destination</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

Transportation cost: 147,949€
Iteration number: 23
Execution time: 0.00652s

The table 2 presents some performances of the FORTRAN95 program on an AMD Sempron 2600+ at 1.6GHz.

<table>
<thead>
<tr>
<th>Name</th>
<th>m</th>
<th>n</th>
<th>p</th>
<th>q</th>
<th>Variable number</th>
<th>Iteration Number</th>
<th>Execution time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>16</td>
<td>4</td>
<td>0.0015</td>
</tr>
<tr>
<td>H10</td>
<td>4</td>
<td>2</td>
<td>10</td>
<td>4</td>
<td>520</td>
<td>35</td>
<td>0.2185</td>
</tr>
<tr>
<td>H20</td>
<td>4</td>
<td>2</td>
<td>20</td>
<td>4</td>
<td>640</td>
<td>67</td>
<td>0.5781</td>
</tr>
<tr>
<td>H30</td>
<td>4</td>
<td>2</td>
<td>30</td>
<td>4</td>
<td>960</td>
<td>92</td>
<td>1.1250</td>
</tr>
<tr>
<td>H40</td>
<td>4</td>
<td>2</td>
<td>40</td>
<td>4</td>
<td>1280</td>
<td>126</td>
<td>1.9687</td>
</tr>
<tr>
<td>H50</td>
<td>2</td>
<td>4</td>
<td>50</td>
<td>4</td>
<td>2000</td>
<td>138</td>
<td>2.8000</td>
</tr>
<tr>
<td>H60</td>
<td>4</td>
<td>4</td>
<td>60</td>
<td>6</td>
<td>5760</td>
<td>243</td>
<td>8.4218</td>
</tr>
<tr>
<td>H55</td>
<td>4</td>
<td>5</td>
<td>55</td>
<td>7</td>
<td>7700</td>
<td>269</td>
<td>9.7343</td>
</tr>
<tr>
<td>H70</td>
<td>4</td>
<td>8</td>
<td>70</td>
<td>5</td>
<td>11200</td>
<td>348</td>
<td>17.2968</td>
</tr>
<tr>
<td>H80</td>
<td>5</td>
<td>6</td>
<td>80</td>
<td>5</td>
<td>12000</td>
<td>373</td>
<td>21.4375</td>
</tr>
<tr>
<td>H100</td>
<td>4</td>
<td>100</td>
<td>100</td>
<td>4</td>
<td>6400</td>
<td>328</td>
<td>25.1562</td>
</tr>
<tr>
<td>H100bs</td>
<td>10</td>
<td>10</td>
<td>100</td>
<td>10</td>
<td>100000</td>
<td>958</td>
<td>165.9218</td>
</tr>
</tbody>
</table>

This program is treated in the case where the SEC is verified. When the SEC is not confirmed, our work is in progress. Based on the solver to obtain, we foresee to implement an industrial case study for Aerospace Auvergne Silicon (AAS) which is the leading supplier of elementary parts and Airbus logistic service.

Figures 1: AAS transportation diagram

We model the transportation problem by setting the database consisting of 212 parameters, 9600 unit costs and 9600 variables as follow:

- Origin: the group of two AAS sites: Aulnat and Cournon
- Destination: the group of four Airbus sites: St Eloi, Nantes, Méaulte and St Nazaire

- Goods types: 200 references including weight, size, and transportation constraints
- Truck types: one 1.5-ton truck; one 3-ton truck, two 5.5 ton trucks; one 10-ton truck and one 0.65-ton car

Thus, before making the delivery of parts, the logistics management will interrogate this management software that edits a delivery plan after optimization. This plan includes the number of used trucks for a list of parts from origins to destinations and expected cost.

CONCLUSION

Our work has resulted from the resolution of all previously untreated cases of 4TP under exact mathematical angle. Thus, we have proposed a mathematical tool for solving optimization problem of weight transportation. For implementing it in the management system used by an industrial enterprise, we realized a programme in the FORTRAN 95 language with the Gfortran compiler running on a computer equipped with a CPU at 1.6GHz. The execution time is fast enough for middle sized problems and larger ones. In terms of modelling, we foresee to develop a four index model with interval parameters. The research method that we anticipate to use fully takes advantage of the result of general 4TP and that of the two index problem with intervals for source, destination and cost parameters.

REFERENCES


VEHICLE TRAFFIC SIMULATION
AUTONOMOUS VEHICLES SIMULATION: A COMPREHENSIVE REVIEW

José L. F. Pereira, Rosaldo J. F. Rossetti
Laboratório de Inteligência Artificial e Ciência de Computadores
Departamento de Engenharia Informática
Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
email: {jlpereira,rossetti}@fe.up.pt

KEYWORDS
Autonomous Driving, Autonomous Vehicles Simulation, Robotics Simulation, Microscopic Traffic Simulation, Agent-based Simulation

ABSTRACT

Research on autonomous vehicles has come a long way since first findings, and its software tools are increasingly acclaimed by the research community. Particularly with robotics simulators, autonomous vehicles have been provided with a suitable test-bed for experimentation of new methodologies such as long-term navigation algorithms, map building and intelligent reasoning. However, when it concerns the deployment and validation of such vehicles in a larger urban traffic scenario, robotics simulators do not seem to provide the required functionality for road traffic analysis, or inter-vehicular communication infrastructure as they seem present in today’s traffic simulators. The improvement of such features is the key for the successful practical deployment of such a critical system.

In this paper, we present a study on the currently available robotics and traffic simulators focused in its applicability on autonomous vehicles research. The simulator integration approach was introduced by Figueiredo et al. (2009), although the selected software architecture and frameworks did not provide satisfactory results. In the former paper, the traffic and robotics simulation softwares chosen did not have the sufficient maturity level, which led to various implementation difficulties and, consequently, to an inefficient platform. However, its findings have provided some lights on the usefulness of newer possible integrations and difficulties to be overcome.

In this paper we present the current features and issues on some of the most known robotics and traffic simulators, and propose a taxonomy for selecting these simulators, in order to foster further developments on the topic.

INTRODUCTION

Urban traffic represents one of the most problematic products of our contemporary society. Traffic accidents caused by driver error are commonly associated with alcohol and drugs consumption, stress and, more recently, distraction due to cell phone use. Traffic congestions are found very often on road networks, causing high levels of pollution as well as increased delays and money spent in a journey. Ultimately, society’s quality of life is substantially degraded with these effects. With the introduction of Intelligent Transportation Systems (ITS) and accounting for Future Urban Transport Systems (FUT) requirements, traffic flow in overpopulated areas was vastly improved along with overall road safety and security. This effort has been taking a wider focus with research of autonomous urban vehicles, as recent advances in technology processes supply the needed software and hardware requirements for such critical and complex systems.

There are already present in literature various approaches to this concept since the 70s, albeit only recently the industry market started investing on the needed technology. Also, some autonomous vehicles competitions were held past millennium, such as the Defense Advanced Research Projects Agency (DARPA) Grand Challenge, to encourage researchers onto the development of driverless cars (Leader et al. 2007).

Some authors, (Dickmanns et al. (1994) or Pomerleau and Jochem (1996)), point to the autonomous vehicle paradigm as a software problem. Not surprisingly, the most difficult challenges about autonomous systems are on the algorithmic side. Particularly in autonomous vehicles research, the highly non-linear behavior of traffic networks’ heterogeneous interacting entities is a rather complex task to handle.

Robotics simulators have been the standard test-bed to experiment with new vehicles methodologies of autonomous navigation, obstacle avoidance and human-machine interaction. However, when developing new approaches such as cooperative driving behavior in realistic multi-modal traffic scenarios, these simulators are still very unfeasible.

Following the idea proposed in Figueiredo et al. (2009),
and facing this emerging interest regarding the application of robotics on regular-sized vehicles, both traffic and robotics simulators must be integrated with each other in order to allow engineers and researchers to evaluate and deploy new methodologies towards FUT. This paper proposes a taxonomy for the evaluation of both a traffic and a robotics simulator towards autonomous vehicles simulation and reviews some of the most well-known simulators to assess their applicability in the integration.

**MICROSCOPIC TRAFFIC SIMULATION**

The use of simulation methodologies in the field of Transport Systems has been widely acclaimed for decades. If one takes at current transportation states in urban scenarios, high traffic saturation levels due to the increasing demand and unoptimized transportation planning is evident (Yang et al. 2010). Traffic simulation tools intend not only to cope with undesired events as noticed above, but also to generate scenarios, optimize control, and predict network behavior at the operational level. This allows a specialist to virtually modify the network topology or the traffic control strategies in order to validate the reliability of new models without any disruption to traffic in a real network. With the emergence of ITS and FUT (Passos et al. 2011), traffic simulators are as well being constantly upgraded to support new features such as intervehicular communications (IVC), multi-modal simulation, or environment and emissions reporting (Sullivan et al. 2004).

When simulating autonomous vehicles in an urban traffic environment, each vehicle must be individually modeled to allow singular data extraction. Furthermore, each vehicle sensors’ input is simulated taking into account the current and past surrounding states, such as neighbor vehicles kinematics, traffic lights state or the presence of pedestrians. Also, according to Punzo and Ciuffo (2010), when the aim of the study is to design and evaluate systems that affect driver behavior and choices, detailed microscopic models should be used.

Having this perspective in mind, the traffic simulators to be reviewed may be based on a microscopic level of detail. In fact, regarding the framework capability to be extended to external driver agents, a nanoscopic simulation level is achieved, as the Driver-Vehicle-Environment system states (Ni 2006).

The microscopic traffic simulation consists of a set of models representing the individual vehicle behavior in road traffic that should be calibrated to follow the macroscopic traffic flow patterns. Despite its complex configuration, once a good calibration is set up, the model follows the macroscopic traffic flow patterns allowing for a wider analysis on the vehicle behavior, showing itself to be suitable for individual intersection optimization, e.g. traffic light planning.

The general approach when simulating traffic in a microscopic level of detail is to treat the driver and vehicle as one unit. Hence, there are several dynamic rules also called behavioral models, each one supporting a specific interaction. The most important behavioral model is one that permits handling the longitudinal interaction between two preceding vehicles, and is commonly known by the car-following model. However, there are other sub-models used depending on the simulators’ capabilities and type of road to be simulated (e.g. lane-changing, gap-acceptance, overtaking, ramp merging, speed adaptation) (Olstam 2005).

The aforementioned rules should be reliable to simulate a realistic traffic environment, however, they still suffer from the following drawbacks:

- They are usually decoupled from each other, i.e. on a lane changing, no longitudinal acceleration is affected.
- The vehicle lateral position is discrete (per-lane).
- There is no cooperation capabilities among the various entities of the road.

More recently though, some alternatives were presented to address some of the aforementioned issues using agent-based methodologies (Chen 2010). However, such implementations did not have the required maturity level and popularization to be evaluated on this paper. Other recent improvements on microscopic traffic simulation are the integration of the agent computing paradigm into many aspects of transportation systems, such as in modeling and simulation, dynamic routing and congestion management and intelligent traffic control (Chen 2010). Also the use of game engines technology in simulators is proposed by Miao et al. (2011) with promising potential.

**Criteria for Evaluating Traffic Simulators**

Bearing the above perspective in mind, some microscopic traffic simulators are briefly analyzed towards is applicability in autonomous driving. One must note that only fully validated and widely used simulators are described. We assume these simulators perform the minimal aspects of a reliable microscopic traffic simulation to simplify the study. In Passos et al. (2011), a comparison taxonomy for microscopic simulators applied to FUT is proposed, thereby offering a good start point into the criteria selection for integration with autonomous vehicle simulators. Furthermore, the selected criteria is described below:

**Software License** Open-source simulators are generally inferior in features as commercial ones. Nonetheless, when they are well documented they
Figure 1: A screenshot of the open-source Simulation of Urban Mobility (SUMO) traffic simulator

are more flexible and are rapidly extended as a result of community support.

**Extensibility** When using a closed-source software, its extensibility should be analyzed in order to study if it suits the integration of other tools, i.e. the level of accessibility of the simulation core.

**External Agent Support** The ability to use agent methodology, not only in driver behavior modeling, but in simulation initiation, control or deployment.

**Parallelism/Distribution** To support a large traffic scenario, simulators must feature distributed processing over several cores or a computer network.

**Inter-vehicular communications (IVC)** Virtual communication infrastructure support for V2V or V2I and physical restrictions simulation must be present as well.

**Interactivity** What features are controllable from simulation in run-time, and general graphical aspect.

**Level of Maturity** Whether the simulator is widely used and validated by the scientific community.

A taxonomy to compare microscopic traffic simulators and their application to autonomous vehicles simulation was proposed. Finally, some results are described in the following section.

**A Brief Analysis on Traffic Simulators**

Following the comprehensive analysis in Passos et al. (2011), a general overview of the referred simulators against the selected criteria is presented, namely VISSIM (Chen et al. 2007), PARAMICS (Bertiini and Lindgren 2002), AIMSUN (Barcelo et al. 1998), MITSIM (Chen et al. 2010), SUMO (Krajzewicz et al. 2002) and MAS-T²er Lab (Ferreira 2008). Figure 1 depicts the GUI aspect of SUMO microscopic traffic simulator.

Table 1: Feature comparison of microscopic traffic simulators for agent-based autonomous vehicle simulation.

<table>
<thead>
<tr>
<th>Simulator</th>
<th>License</th>
<th>Extensibility</th>
<th>Agent Oriented</th>
<th>Parallelism/Distribution</th>
<th>IVC</th>
<th>Interactivity</th>
<th>Maturity Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>VISSIM</td>
<td>Cmrcial</td>
<td>Y</td>
<td>N</td>
<td>Y³</td>
<td>N</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>PARAMICS</td>
<td>Cmrcial</td>
<td>Y</td>
<td>N</td>
<td>Y³¹</td>
<td>N</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>AIMSUN</td>
<td>Cmrcial</td>
<td>Y</td>
<td>N</td>
<td>Y³</td>
<td>N</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>MITSIM</td>
<td>Both</td>
<td>Y²</td>
<td>N²</td>
<td>Y³</td>
<td>N</td>
<td>L</td>
<td>M</td>
</tr>
<tr>
<td>SUMO</td>
<td>GPL</td>
<td>Y²</td>
<td>N²</td>
<td>Y³</td>
<td>N</td>
<td>M</td>
<td>H</td>
</tr>
<tr>
<td>MAS-T²er Lab</td>
<td>Free</td>
<td>Y¹</td>
<td>Y¹</td>
<td>Y³</td>
<td>N</td>
<td>M</td>
<td>L</td>
</tr>
</tbody>
</table>

Regarding to the extensibility of these simulators, all of them offer some type of modularization. MAS-T²er Lab simulator only provides an UDP connection to control semaphoric intersections and statistical data. SUMO simulator provides an extension named TraCI, which provides statistical data and direct access to some core elements. Nonetheless, it has been extended by the TrasMAPI project to support the implementation of agents in Java (Timóteo et al. 2010). All commercial simulators seem to fulfill this requirement, but a tougher analysis should be needed to evaluate the possibility of integrating external vehicles in real-time into the simulations.

VISSIM, PARAMICS and AIMSUN are full closed-source packages, MITSIM has both the closed- and the open-source variants and SUMO and MAS-T²er Lab are open-source, with being SUMO the most featured and referenced open-source project with over hundred papers. The simulator licensing is a very important factor, as in the case of modification needed on the core, only open-source ones would allow it. Only MAS-T²er Lab seems to support agent-based driver behavior simulations, with multi-connection and local information.

PARAMICS is the only simulator supporting distributed computing over a network off-the-shelf. The remaining simulators support parallel processing over all CPUs.

Inter-vehicular communications are not supported by any of the commercial variants, as they do not seem to

---

1 Only for external traffic light control.
2 Being an open-source software, it can be extended through source code modification.
3 Supports Parallel simulation processing.
4 Supports Distributed simulation processing.
5 Successful implementation reported in Sommer et al. (2007).
be targeted at research purposes. Only SUMO was already modified to support this kind of simulation (Sommert et al. 2007).

Regarding the interactivity criterion, the most 3D realistic simulators are the commercial ones, followed by MAS-T²er Lab. Only SUMO and MITSIM do not have 3D visualization. In-simulation parameter modification is only widely supported by commercial applications and SUMO.

With regard to the last criterion, commercial simulators have an expected high maturity, and on the open-source side, only SUMO is being actively developed.

There is an obvious disparity when looking at microscopic traffic simulators. Although commercial packages are very expensive, they provide the most feasible results. However, their source-code is not accessible, meaning that if an important core modification was needed, a dead end would be reached.

On the other hand, open-source simulators are at in inferior maturity level, albeit they provide full custom control of the application which seems to be most suitable when in a research context.

A brief summary of the aforementioned observations is presented in table 1, using the following notation: L - Low, M - Medium, H - High, Y - Yes and N - No.

ROBOTICS SIMULATION

The autonomous vehicle simulation, on the other hand, is much more recent, and also suffers from some handicaps regarding its ability to be integrated into a traffic simulation environment.

A viable simulation of an autonomous vehicle entity presupposes the modeling of all electronic sensors and actuators in a near realistic environment in order to study the following paradigms (Brunet et al. 1995):

- Complex algorithms for sensory acquisition and fusion.
- Navigational planning and reasoning.
- Control of steering and traction systems.

One of the important aspects to take into consideration while selecting a robotics simulation framework will be the types of sensors supported and their implemented mathematical model.

Robotic Sensors

The following list contains the most used sensor types in autonomous robots, with a particular focus in ground vehicles. A comprehensive research was also made of the technical papers from DARPA’s Urban Challenge 2007 teams.

GPS Provides absolute location and velocity using orbiting satellites, however it only works in open spaces. High quality GPS systems can have a precision on the centimeter scale when in optimal conditions.

Optical Camera Gives vision to the robot. In controlled conditions, vision-based algorithms recognize anything in the surroundings.

Infra-red Camera Similarly to the former optical camera, it gives vision to robots in the infra-red spectrum. It is commonly used along with an infra-red light to provide night vision.

Laser scanner Also known as LIDAR (Light Detection And Ranging), laser scanners are the most popular and expensive sensors used on autonomous vehicles. It returns information of surroundings as a 3D point cloud typically at a 12.5Hz sampling rate.

Ultrasound Measures short distances recurring to the sound spectrum.

Radar Similarly to ultrasound, but in the electromagnetic spectrum. It can achieve even more precise results at a long distance.

Odometry Estimates changes in velocity, acceleration and position from moving sensors, e.g. legged joints or wheels.

Inertial Measurement This sensor type returns a measure from the relative movement of the robot, in a linear or angular context, be it an accelerometer or a gyroscope.

Compass Determines the robot direction relative to the Earth’s magnetic poles.

Actuator types are not going to be thoroughly evaluated as despite their being present in autonomous vehicles, there is a wide support in any up-to-date simulator.

Criteria for Evaluating Robotics Simulators

Nowadays, robotics simulators are indispensable for reliable intelligent robot deployment, responding to the complexity of overall system complexity and expenses. Moreover, as autonomous vehicle research is a branch of robotics, typical solutions seem to adopt these as a base framework for autonomous vehicles simulation. Also, when using a Hardware Abstraction Layer (HAL) between the hardware level (actuators/sensors) and the implementation code, more flexibility would be achieved in the two parts as if they were virtually decoupled.

Bearing this in mind, a taxonomy for comparing autonomous vehicles simulators is proposed. It is already assumed that these simulators perform both reliable 3D
simulation and visualization of the scenario, and also are widely used by the research community. At least one robotics HAL framework must be supported.

The following criteria depict the most important aspects to take in account while selecting a robotics platform in an autonomous vehicle approach:

---

**3D rendering** The visual robustness of the simulation.

**License** Whether the simulator is open-source, free of charge, or commercially licensed.

**External Agent Support** In order to control a vehicle using an agent-based methodology the simulator should feature a distributed architecture at the control level.

**Parallelism/Distribution** In order to distribute processing power over processor cores or networks.

**Level of Maturity** If the simulator is already widely used and validated.

**Fault-tolerance** When a hardware module fails, higher level modules should rapidly make decisions whether to stop or adapt the control system. When developing a final product such behavior should be strictly tested.

**Realistic Scenario Simulation** The level of realism to simulate difficult context scenarios e.g. snow, day and night, wind not only interactively but also physically, i.e. affecting the sensorial input.

---

### Robotics Simulator Analysis

After careful research, we have limited the review to two up-to-date open-source and two closed-source simulators: Microsoft Robotics Developer Studio (Jackson 2007), Player/Stage/Gazebo (Gerkey et al. 2003), USARsim (Carpin et al. 2007) and Webots (Michel 1998). From researching DARPA challenge technical papers, we acknowledged that most of the teams developed their own custom simulators or used/adapted one of the above. Figure 2 illustrates the Webots robotics simulator.

---

Table 2: Feature comparison of robotics simulators for agent-based autonomous vehicle simulation.

<table>
<thead>
<tr>
<th>Simulator</th>
<th>License</th>
<th>External Agent Support</th>
<th>Parallelism/Distribution</th>
<th>Level of Maturity</th>
<th>Fault-Tolerance</th>
<th>Realistic Scenario Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ms Robotics</td>
<td>Free</td>
<td>N</td>
<td>Y1</td>
<td>M</td>
<td>N</td>
<td>H</td>
</tr>
<tr>
<td>Gazebo</td>
<td>GPL1</td>
<td>Y</td>
<td>Y</td>
<td>M</td>
<td>N</td>
<td>M</td>
</tr>
<tr>
<td>USARsim</td>
<td>GPL</td>
<td>Y</td>
<td>Y</td>
<td>M</td>
<td>N</td>
<td>H</td>
</tr>
<tr>
<td>Webots</td>
<td>Commercial</td>
<td>Y</td>
<td>Y</td>
<td>H</td>
<td>N</td>
<td>M</td>
</tr>
</tbody>
</table>

Microsoft Robotics Developer Studio (MRDS) is a robotics platform from Microsoft Company using .NET-based technology. It features visual programming, web- and windows-based interfaces, 3D simulation with advanced physics, as well as easy access to robot’s sensors and actuators from a number of languages. The Player/Stage project is an open-source distributed robotics platform. It features a distributed POSIX-compatible HAL on a minimal design. Stage and Gazebo are 2D and 3D simulators respectively, Gazebo is build on top of the rendering engine Ogre3D to provide more realistic environments. USARSim is a high-fidelity open-source simulator based on the Unreal game engine. It is the official simulator platform of the Robocup virtual robot competition supporting a wide range of sensors with noise input. The USARSim simulator is compatible with the MOAST (Scrapper 2006) and Player frameworks. One curious characteristic of Mobility Open Architecture Simulation and Tools (MOAST) is that it implements the Real-time Control System (RCS) reference model architecture (Albus 1994). This model was developed by National Institute of Standards and Technology (NIST) to be used as a standard. It is commonly applied in many kinds of robot control such as autonomous vehicle control.

---

1A license for Unreal Engine must be purchased. Future versions will use Unreal Development Kit, which is free of charge.

2Supported on robot domain. Latest development version also features distributed simulation processing, i.e. sensors can be simulated on different machines.

3Only supported on robot domain.

---

221
Webots is a commercial robot simulator developed by Cyberbotics used in more than 800 universities and research centers worldwide. It has reached a fairly stable state and supports a wide range of hardware. It is also supported by Universal Robot Body Interface (Baillie 2004), a client/server based framework targeted for humanoid devices.

This study uses Figueiredo et al. (2009) and Petry et al. (2011) as a reference works. Table 2 depicts these simulators capabilities in the earlier proposed criteria, using the following notation: L - Low, M - Medium, H - High, Y - Yes and N - No.

Support for autonomous vehicles in robotics simulators lacks some key characteristics in order to realistically simulate from the sensor to the environment level. However, its current state already allows practitioners to perform fairly complex testing within its deployment.

CURRENT ISSUES ON THE INTEGRATION OF SIMULATORS

If we bear in mind the distributed ability of the integration of both a traffic and a robotics simulator, we must admit a networked access to its core level, in order to provide data interconnection between them. Furthermore, to integrate two simulators in time and space, a bidirectional communication should take place, with the autonomous vehicles providing kinematic variables to the traffic simulator. The traffic simulator then calculates its surroundings and return its data back to the simulator. All of these transactions should occur in the same time step.

In order to implement such an infrastructure between the two simulators, a set of parameters should be properly handled on the data exchange methodology, as proposed below:

Network topology consistency between simulators
Each traffic and robotics simulator requires a specific scenario information. While the traffic simulator may need consistent data about road characteristics and positions, such as number of lanes, road segments or traffic lights information, the autonomous vehicle simulator demands for a highly realistic and calibrated 3D scenario with accurate road geometry for instance, usually not present in most traffic simulators. In order to frame this issue, each file format for the two simulators should inherit from the same network description file (e.g. a shape file) to maintain their coherence.

Synchronization of Simulators: Although traffic simulators are not implemented with hard real-time constraints, the processing power of today’s computers allow us to consider this is quite acceptable. As most traffic simulators supports more than a thousand individual vehicles in real-time, a 30 Hz frame rate may be achievable on an optimized data exchanging between simulators. If we take into account this simulation approach should support more than one autonomous vehicle, a large data flow should be expected between the two simulators. Moreover, all step calculations need to be inferior to the overall frame rate of the simulation for a correct user experience.

From the former synchronization critical issue, an obvious choice for solving such a problem is the minimization of data flow between traffic and autonomous vehicle simulators. Below, three methods are presented aiming to minimize them:

Simulation of only the surrounding elements
When simulating an autonomous vehicle entity, only its surrounding elements (vehicles, traffic lights, etc.) information is needed to process the next sensors’ state. Moreover, to save network resources between simulators, a 2D envelop is defined around the autonomous vehicle which will be synchronized with the same envelop on the traffic simulator. Using this approach, the internal congruence can be maintained between the simulators without having a large amount of redundant data among them.

Serialized binary data: Former distributed approaches make use of the XML file format for data exchange. However, newer formats such as serialized data report 20 to 100 times more efficiency in data transactions. This technology approach is well acclaimed in today’s large-scale distributed systems (e.g. Google data centers).

Asynchronous data exchange: To minimize data flow and to account for its content, instead of a specific number of transactions between simulators be exchanged in each time step, it should be reduced to when a change in the variable state happens. For instance, considering an autonomous vehicle X is in a state $S_r$ in step $\tau$, only when $S_r \neq S_{r+1}$ the simulator will report the traffic simulator about its changes.

One must note that, although the cited issues are the most obvious problems for integration of a traffic simulator with an autonomous vehicle simulator, implementation issues may also arise depending on the chosen software and the architecture of the whole system.

CONCLUSIONS AND FUTURE WORKS

Research in autonomous vehicles has proven that it is quite possible to integrate fully operator-independent
robotic systems with an urban traffic environment. Indeed, in June 2011 the state of Nevada has announced they will be the first place in the world to allow self-driving cars on its urban roads (Nevada 2011). Facing the current traffic situation in most developed countries, it is now imperative to foster new transportation methods using state-of-the-art technologies towards the implementation of the so-called Future Urban Transport (FUT). With the attention on sustainability, autonomous vehicles will also gain a particular focus as its paradigm envisions an equal and sustainable ground transport which should ensure its users improved productivity and better mobility.

With new performance measures stirring up from FUT requirements as well as the likely perspective of our urban areas being populated with fully autonomous vehicles, it is evident there will be an increasing demand for appropriate tools to test with alternative and innovative transport solutions, management policies and control strategies. On the one hand, interactions within complex traffic systems involving unmanned vehicles and traditional vehicles steered by normal people must be thoroughly assessed for the sake of safety and efficiency of the system. On the other hand, the full potential of driverless cars is still questionable as much scepticism still remains within more conservative practitioners. To address the aforementioned issues, some traditional microscopic simulators have been adapted or extended to partially support many of these new performance measures. This has given rise to more advanced traffic models. However, more complex behaviours of autonomous vehicles demand appropriate representations that are not contemplated in such new generation simulation tools. The community then has recurrently opted for the use of robust robotics simulators to overcome such limitations, which are limited to some extent to the number of interacting entities thus constraining the complexity of scenarios to be simulated.

In this paper we contribute with a comprehensive review of the current state-of-the-art traffic and robotics simulators as a means to fully understand their characteristics, allowing us to select those that can effectively foster the appropriate assessment of the full potential of new ITS technologies such as autonomous driving. From the discussion herein presented, we can perceive that these software tools are not yet adapted to support the analysis of complex environments such as those encompassing autonomous vehicles systems. Bearing this in mind, when selecting the appropriate tools, practitioners should have to account for the openness of the frameworks in order to be able to modify them in all their essence, if necessary, and therefore have a highly customizable work-bench. As a practical result of this study, we conclude that the combination of characteristics of both microscopic and robotics simulators is imperative. Highly populated areas, as in urban scenarios, are very likely to encompass a high number of interacting entities, which can be reasonably represented by traditional microscopic models. More complex details of internal works of autonomous vehicles, on the other hand, are appropriately represented within robotics simulators. The combination of both is our approach to implement an appropriate environment to test with FUT requirements and new performance measures imposed by highly complex transport and mobility systems.

Based on the study carried out and presented in this paper, we have selected the two simulators to be integrated, namely the microscopic traffic model and the robotics simulator. The integration architecture, issues related to data exchange and performance assessment are out of the scope of this paper though. Nonetheless, a proof-of-concept prototype is fully operational and demonstrates the full potential of such an approach. The very next steps in our research work include the improvement of that prototype and the simulation of real complex traffic networks. We are well aware of the need for a sound and robust methodology for validating and calibrating such models. Indeed, combining autonomous vehicles with human-steered cars in urban scenarios is a hard and complex task to perform, as complex interactions are to be accounted for, as well as issues such as privacy, safety, accessibility and equality are also very likely to arise then. In summary, this area represents a very prominent and emergent field of research, deserving a great attention from the scientific community, the Industry, governments and practitioners in general with a vast potential to be explored in many different ways.

ACKNOWLEDGEMENTS

Authors gratefully acknowledge the financial support from the Master’s Programme in Electrical and Computers Engineering, MIEEC/FEUP.

REFERENCES


IEEE Transactions on Intelligent Transportation Systems, 1–11.


UTILISATION OF COMPUTER SIMULATION FOR TESTING ADDITIONAL SUPPORT FOR DISPATCHING RAIL TRAFFIC

Jan Fikejz and Antonín Kavička
Department of Software Technologies
Faculty of Electrical Engineering and Informatics
University of Pardubice
Studentská 95, CZ-532 10 Pardubice
Czech Republic
E-mail: Jan.Fikejz@upce.cz, Antonin.Kavicka@upce.cz

KEYWORDS
Dispatching control, rail traffic, decision making support, global navigation satellite system, railway infrastructure, rolling stock position visualisation.

ABSTRACT
The attention is paid to the design of a demonstration software tool InfraRAIL (including relevant testing with the help of computer simulation) determined for an additional support related to the dispatching control of the rail traffic namely within the frame of regional railway lines in the Czech Republic. The utilisation of satellite navigation for rolling stock positioning is discussed. Consequently the design of a railway infrastructure model is introduced (incl. relevant data structures). The model reflects the needs to store rolling stock positions and to make relevant visualisations. The simulations of rolling stock movements are carried out in order to test the functionality of InfraRAIL tool. Within this context the real historical (time-stamped) data items are utilised. In addition, generated data reflecting potential collision train movements (typically within the frame of single-track line section) are exploited.

INTRODUCTION
Availability of current rolling stock positions (within railway infrastructure) plays an important role in the area of dispatching rail traffic. There are two main problems in this field: (i) how to get information about rolling stock position and (ii) how to represent/model and visualise the railway infrastructure (incl. depiction of rolling stock positions) on the computer.

Rolling stock positioning can be based on the utilisation of GNSS (Global Navigation Satellite System). Disadvantages of that system (e.g. limited precision, no usability in tunnels etc.) are not so important in this case because GNSS is supposed to be utilised “only” within an additional/auxiliary support related to the dispatching control (i.e. another GNSS-unrelated dispatching support system is available).

The problem of modelling and visualising railway infrastructure is supposed to be completely solved because any unified railway regulation does not pay attention to that area (Fikejz 2010; Fikejz 2011).

UTILISATION OF GNSS TECHNOLOGY
For the needs of positioning only the system GPS (Global Positioning System) can be currently applied – European system Galileo is supposed to start in 2017. Utilisation of satellite navigation for rolling stock positioning is frequently criticised mainly because:

- GPS is not sufficiently precise,
- GPS is not applicable within the tunnels,
- GPS is not reliable enough for the needs of train movement control,
- GPS signal is not available along all railway lines,
- GNSS systems can be eliminated any time in case of some crisis or war conflict.

In spite of the above mentioned facts the additional support (of rail traffic control) can exploit the GPS system. GPS system reaches the precision about ±10m – the concrete real precision depends on the number of available satellites and the character of the landscape. For example in the open landscape the precision can be ± 3m (Filip 2007). In addition, there are always more and more sensitive GPS chips (e.g. SIRFstar IV) available. Those chips enable to get more precise position information even within problematic landscape sections.

During the last years selected trains in the Czech Republic contain specific communication terminals including GPS modules. These terminals periodically transmit the data (using GPRS - General Packet Radio Service) involving information about the current train position. The mentioned terminals are as follows:

- Telarail TLR-ZJ (made by Unicontrols, a.s.) and
- Radiostanice VS67 (made by T-CZ, a.s.).

The terminals exploit for the communication purposes the specialised mobile phone network GSM-R (Global System for Mobile Communications - Railway) constructed along
the selected railway lines. In case that GSM-R signal is not available, the terminals try to communicate through the public/general mobile phone network (GSM). The scheme of communication model is illustrated in the figure 1.

**DESCRIPTION OF RAILWAY INFRASTRUCTURE**

As mentioned above no unified methodology focused on the railway infrastructure description currently exists in the Czech Republic. The mentioned disadvantage can be eliminated in the future using unified European regulation INSPIRE - INfrastructure for SPatial InfoRmation in the European Community (Fikejz 2010; Evropa 2007). This regulation contains (among others) formal descriptions related to transportation networks of different kinds (the relevant data items are stored in the specific format based on the XML standard).

In order to describe the Czech railway infrastructure the older regulation M12 of the Czech Railways can be partially utilised. The regulation M12 (focused on the unified denotation of railway lines and railway yards) mainly pays attention to the methodologies of:

- a basic description of the railway network, railway lines and yards and their partial components for the needs of railway information systems,
- mutual conversions among different description schemes applied within various railway information systems,
- creating and updating codebook files related to railway lines descriptions.

For the needs of rolling stock positioning the following facts involved within M12 regulation are supposed to be emphasised:

- the unique identifier specifies every railway line,
- the railway line can be decomposed to so called line definition supra-sections (hereafter supra-sections),
- every supra-section is supposed to be divided into individual elementary definition sections (hereafter elementary sections), the length of which is usually several kilometres – every elementary section is reflected within a special data table (called as track section table)
- elementary sections can be further sectionised by means of the mileposts (distances between the mileposts are 100 metres – some exceptions may occur).
- substantial transport position defines the track embranchment within railway infrastructure,
- supra-sections are potentially further integrated into higher hierarchical units called super-routes.

The mileposts are interconnected with the physical infrastructure (they are placed directly within the set of tracks) and represent traditional important localisation points on the railway network. For example, the positions of the mileposts are also utilised within the information system employed by the General Headquarters of Fire Department (Čihal and Procházka 2006). The total number of the mileposts exceeds 100 000 and 99.95% of them are visualised. The following database items characterise every milepost:
Table 1: Table of mileposts and their characteristics

<table>
<thead>
<tr>
<th>Milepost identifier</th>
<th>Track section table identifier</th>
<th>Kilometrical milepost position</th>
<th>Type of track embranchment</th>
<th>X-coordinate</th>
<th>Y-coordinate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>10112</td>
<td>18.8</td>
<td>T</td>
<td>15.1781377781762</td>
<td>50.3735413890415</td>
</tr>
<tr>
<td>2.</td>
<td>10112</td>
<td>18.9</td>
<td>T</td>
<td>15.2392483335071</td>
<td>50.3317077774472</td>
</tr>
<tr>
<td>3.</td>
<td>10112</td>
<td>19</td>
<td>S</td>
<td>15.1349277777142</td>
<td>50.4403944444656</td>
</tr>
</tbody>
</table>

- milepost identifier,
- relevant track section table identifier,
- kilometrical milepost position,
- type of track embranchment,
- X-coordinate,
- Y-coordinate,
- number and name of the railway line according to the public timetable (tab. 1).

The connections between the regulation M12 and the mentioned table of mileposts (tab.1) are represented by the values contained within the second column “track section table identifier”.

The columns “X-coordinate” and “Y-coordinate” contain the values specifying the geographic mileposts positions. The values were computed by the Railway Infrastructure Administration (the Czech national manager of the railway network). It was used the national coordinate system called JTSK, whereas the relevant coordinates can be transformed into the system WGS-84 (World Geodetic System 1984) exploited by GPS system.

The designed model of railway network infrastructure (incl. relevant data structures) takes into account the connection between railway regulation M12 (namely with regard to substantial transport positions, i.e. embranchments of tracks) and the table of mileposts (tab.1)

**RAILWAY NETWORK REPRESENTATION**

The model of a railway network infrastructure can be intuitively designed as an undirected graph (defined in graph theory). The graph vertices reflect the selected mileposts and the edges are determined on the basis of a further specific elaboration.

Analysis of information provided by the Railway Infrastructure Administration leads to the conclusion that the following four partial tables of data are needed for the design of a railway network representation:
- table of mileposts (tab.1),
- table of super-routes,
- table of railway stations and
- table of supra-sections.

The key link among the added tables represents the track section table (reflecting relevant elementary track definition section). All the tracks within railway station belong to one track section table and therefore it is necessary to make certain reduction of the set of mileposts. It means the railway station is represented by just one running track belonging to one track section table. The second modification of the mileposts table is connected with track embranchments on the railway network. All cases of track branching are indicated by the flag value 'S' (the switch) in the table of mileposts (tab.1). The flag value 'T' (the track segment) indicates the place without the track embranchment.

The following generalised algorithm carries out the above mentioned reductions – the data obtained from that algorithm are consequently stored with the specialised table called VERTICES.

**The algorithm counting the set of graph vertices**

1. They are searched out all the records from the mileposts table with the flag value “S” (reflecting the switches).
2. All found records are consequently divided into the groups according to the relevant track section table identifier. Then, the records are sorted according to the values of items „kilometrical mileposts positions” within the frame of each group.
3. The established groups of mileposts (with the same values of items „kilometrical mileposts positions”) represent new members of the set of graph vertices – it is necessary to calculate new GPS coordinates for these points. Every newly calculated geographical coordinate corresponds to the arithmetical mean with regard to all relevant coordinates from the considered group of mileposts.
4. Finally, all the records from the mileposts table with the flag value “T” (reflecting the track segments) are inserted into the set of graph vertices.

The data items obtained from the algorithm (and stored within VERTICES table) represent information about graph vertices needed for a relevant data structure reflecting the undirected graph (i.e. railway network model).

The edges of the considered graph are supposed to be acquired from the tables of super-routes and VERTICES by means of another algorithm, which results are stored in a table called EGDES.
The algorithm counting the set of graph edges

1. It is selected such a super-route, which was not processed yet.
2. They are selected all supra-sections from the current super-route.
3. For every track section table belonging to the supra-section the corresponding mileposts are sorted according to the values of relevant items „kilometrical mileposts positions”.
4. Then, the edge is established between two successive mileposts.
5. In case of a changeover from one track section table to another one, the relevant edge is defined between the last milepost from a previously processed track section table and the first milepost from the current track section table.

Thus, two above specified algorithms count the tables of vertices and edges mirroring the railway network infrastructure.

In addition, it is required to create another railway network model (using higher level of abstraction) for the needs of monitoring rolling stock positions within the higher railway network units. The mentioned level is composed of super-edges, which correspond to such sections of the railway lines, where no track branching occur. A derived table (called SUPERVERTICES) is counted by a quite simple algorithm. The algorithm takes such elements of the VERTICES table, which correspond to such graph vertices that are adjacent to more than two graph edges. The derived table SUPEREDGES is constructed with the help of processing records from SUPERVERTICES table (using an analogical way of getting table of EDGES from the table of VERTICES). Thus, the tables of edges and vertices reflecting railway network on the higher level of abstraction are established.

ROLLING STOCK POSITIONING

The data describing the particular position on the land surface can be classified as multidimensional. The minimal requirement of positioning expects the utilisation of two coordinates. The multidimensional data can be stored:

- within suitable data structure directly in the main computer memory or in the external storage (Samet 2006),
- in databases supporting storage of multidimensional data (e.g. ORACLE Spatial or MS SQL 2008).

Both mentioned database products provide a support for processing multidimensional data, however each of the products implements different technical solution – the presented software tool InfraRAIL stores the relevant data in the system ORACLE Spatial (Fikejz 2010; Murray 2003).

Having data (obtained from the relevant GPS module) reflecting the current position of rolling stock on the land surface, it is possible to find the nearest point/record in the table of VERTICES (SDO_NN operator searching the nearest neighbour can be utilised). It means in fact that rolling stock position can be linked with the railway network (e.g. its kilometrical position within the relevant elementary track section).

VISUALISATION TOOLS

Considering railway network visualisation it is also necessary to concern how to store the multidimensional data. If the database ORACLE Spatial is utilised the visualisation Java-based tool MapViewer (www.oracle.com 2008) can be exploited. MapViewer represents J2EE service for displaying maps using the spatial data (e.g. using object data type SDO_GEOMETRY) managed by ORACLE Spatial. The basic conception of MapViewer illustrates the figure 2.

![Figure 2: The basic conception of the MapViewer service](image)

Nowadays three following kinds of servers are applicable for supporting J2EE service:

- WebLogic Server version 10 or higher,
- ORACLE Fusion Middleware or
- application server OC4J.

Application server OC4J is available together with the service MapViewer as the standalone kit, which is possible to be downloaded directly from the ORACLE web portal. If only the MapViewer service is supposed to be utilised then the application server OC4F (with the standalone kit) represents the easiest way how to start using that service. WebLogic Server and ORACLE Fusion Middleware provide more robust palette of services, which are excessive for the MapViewer service.

The individual map layers for visualisations are composed of metadata. For metadata processing the solitary Java-based application ORACLE Map Builder is exploitable (this tool supports building the extensive map layers). The styles represent the wall stones of the map layers (there are defined for example the points, lines, polygons etc. associated with various shapes and colours). The data visualisations use so called themes. They define, the data of which column (from the type SDO_GEOMETRY) are supposed to be visualised. For each visualised column the required style is selected – it means the relevant colours and shapes are assigned.
The last step before visualisation requires creating a relevant map layer. The layer is composed of individual themes, whereas the values called MinScale and MaxScale are set for each theme. These values define the zoom levels, application of which determines if the relevant theme is depicted or not. The designed map layer can be then displayed within the ORACLE Map Builder tool.

MapViewer service provides much more functions supporting the work with map layers. There are three ways of communication how to call the required functions:

- XML API,
- Java-Base API,
- JSP Tag Library.

![Communication concept of the MapViewer service](image)

Figure 3: The communication concept of the MapViewer service

The software application InfraRAIL utilises Java-Base API (the conception of that interface illustrates the figure 3). The selected API is usable for desktop applications as well as and for applets and servlets. Within this context the Java library mvcclient.jar (www.oracle.com 2008) is exploited. This library contains the following packages:

- oracle.lbs.mapclient,
- oracle.mapviewer.share,
- oracle.mapviewer.share.ext,
- oracle.mapviewer.share.mapcache,
- oracle.mapviewer.share.style,
- oracle.mapviewer.share.stylex.

One of the most important packages is oracle.lbs.mapclient containing the key classes for the MapViewer client. One class called “MapViewer” is parameterised by an URL address of MapViewer service. Let us illustrate creating an instance of MapViewer class:

```java
import oracle.lbs.mapclient.MapViewer;
MapViewer mv = new MapViewer("http://localhost:8888/mapviewer/mserver");
```

The visualisations are carried out with the help of bitmaps. MapViewer service generates a new bitmap after attending each relevant query. The bitmap is stored on the server and the client application obtains only an address of the generated bitmap. This technique is exploitable for all three kinds of API. In addition, in case of the desktop Java application another technique is usable: the MapViewer service can send the entire compressed bitmap directly to the desktop application.

**VISUALISATIONS OF THE RAILWAY NETWORK AND ROLLING STOCK POSITIONS**

The visualisation uses the data described above in connection with the memory representation of railway network infrastructure. The data items are stored within ORACLE Spatial database. Next, the relevant map is built (with the help of the tool ORACLE Map Builder), which is composed of three disjoint layers:

- micro-layer,
- mezzo-layer,
- macro-layer.

The visibilities of individual layers do not overlap each other and are always determined by the applied zooming level.

**Micro-layer** represents the highest level of details. Its memory representation is based on the table of VERTICES and EDGES. They are displayed the following graphical elements/symbols on this level:

- mileposts (i.e. graph vertices),
- arcs between the mileposts (i.e. graph edges) and
- railway stations (determined by GPS coordinates corresponding to a specified station centre).

**Mezzo-layer** utilises the same base of data as the micro-layer and displays:

- arcs between the mileposts (the mileposts themselves are not depicted) and
- railway stations.

**Macro-layer** constitutes the layer on the highest level of abstraction. The memory representation exploits the data from the tables of SUPERVERTICES and SUPEREDGES. They are visualised only the super-edges (corresponding to the sections of railway lines, where no track branching occur). The figure 4 depicts a selected segment of the Czech railway network from the viewpoint of macro-layer. Next, the figure 5 demonstrates a cut of the mentioned segment, which corresponds to the mezzo-layer.

![Selected visualised segment of the railway network using macroscopic layer](image)

Figure 4: Selected visualised segment of the railway network using macroscopic layer
Table 2: Recorded time-stamped data about train movements

<table>
<thead>
<tr>
<th>Train number</th>
<th>Latitude</th>
<th>Longitude</th>
<th>Speed</th>
<th>Azimuth</th>
<th>Train vehicle identifier</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>48701</td>
<td>50.02274</td>
<td>15.33554</td>
<td>78</td>
<td>57</td>
<td>91547123022</td>
<td>14.02.11 04:34:25</td>
</tr>
<tr>
<td>48701</td>
<td>50.02654</td>
<td>15.34246</td>
<td>80</td>
<td>43</td>
<td>91547123022</td>
<td>14.02.11 04:34:55</td>
</tr>
<tr>
<td>48701</td>
<td>50.03077</td>
<td>15.34873</td>
<td>68</td>
<td>43</td>
<td>91547123022</td>
<td>14.02.11 04:35:25</td>
</tr>
<tr>
<td>48701</td>
<td>50.03495</td>
<td>15.35505</td>
<td>61</td>
<td>45</td>
<td>91547123022</td>
<td>14.02.11 04:35:55</td>
</tr>
</tbody>
</table>

TRAFFIC SIMULATION ON THE RAILWAY NETWORK

It was needed to carry out simulations of railway traffic within the frame of the tool InfraRAIL (determined for an additional support related to the dispatching control of the rail operation). As mentioned above the selected train locomotives carry communication terminals transmitting packets of data containing current GPS coordinates of the relevant rolling stock. In case the train moves the communication terminal sends position information every 30 seconds. The table 2 contains a sample of recorded data.

Testing application specialised on the simulations (reflecting the rail traffic on the selected segment of the railway network) utilises the following software:

- NetBeans - Java 1.6,
- ORACLE Spatial,
- MapViewer and
- ORACLE Map Builder.

The application exploits the library mclient.jar enabling communication with the MapViewer service (by means of Java API). Using appropriate methods the required functions can be called and rolling stock positions are visualised directly within the relevant map layer.

The recorded historical data about rolling stock positions (on the railway network) contains the time stamps (tab.2). The interpretation of these data items enables to carry out simulation runs (incl. visualisation/animation of the changing train positions). The figure 6 demonstrates the

![Image of a railway map showing train positions](image1)

Figure 5: Selected visualised segment of the railway network using mezzooscopic layer

![Image of a railway map showing train positions](image2)

Figure 6: Running application showing the movements of rolling stock
running application showing the movements of rolling stock on the railway network.

Another set of simulation experiments is based on the generated data corresponding to non-standard/collision situations, which can occur on the railway network. For example, two trains moving against each other on the single-track line represents a typical conflicting situation. The model of a railway network is based on the conception of undirected graph (incl. relevant data structures). Thus, it is possible to couple the current train position with a corresponding graph edge mirroring the track section on which the train occurs. Considering macro-layer of the infrastructure model (being composed of super-edges, i.e. line sections without track branching) the occurrence of two trains on one edge/section indicates a potential collision situation (such an information could be very valuable for the need of dispatching control). The figures 7 and 8 illustrate an example of a potential traffic conflict.

**Figure 7:** Detection of traffic collisions on the single-track line

**Figure 8:** The detailed depiction of a collision situation

**CONCLUSION**

The attention was paid to the possible utilisation of GNSS technology within the frame of an additional software support for dispatching the rail traffic. A multi-layered model (based on undirected graph) of the railway network infrastructure was designed. A selected segment of the Czech railway network was visualised using the tools ORACLE Spatial and the MapViewer service.

The software demonstrator InfraRAIL was built in order to test the railway network model. Rolling stock positioning (incl. relevant visualisations) on the railway network was investigated with the help of computer simulation. Finally, the detections of selected collision situations on the single-track lines were demonstrated.

With regard to the fact that an undirected graph model was utilised (mirroring railway network infrastructure) the next research steps are supposed to be focused on potential applications of selected graph algorithms potentially exploitable within the support related to dispatching the rail traffic.

**ACKNOWLEDGEMENTS**

This work has been supported by the National Research Program of Czech Republic under project “MSM 0021627505 Theory of Transportation Systems" and the project “SGF/E04/2011 Modelling railway network infrastructure” (financed by the University of Pardubice).

**REFERENCES**


Čihal, R; Procházková, P. Možnosti využití popisu železničních tratí ČR v IS ČD pro potřeby IS krizového řízení v dopravě. Nová železniční technika. 2006, 2, pp. 20. (in Czech)


GPS BASED SIMULATION
GPS-BASED REAL-TIME TRANSPORT CONTROL
FOR PRODUCTION NETWORK SCHEDULING SIMULATION

Matthias Klumpp
Christof Kandel
Institute for Logistics and Service Management
FOM University of Applied Sciences
Leimkugelstraße 6
D-45141 Essen, Germany
E-mail: matthias.klumpp@fom-ild.de
www.fom-ild.de

KEYWORDS
Event-oriented, Network, Production, Real-time, Scheduling.

ABSTRACT
Nowadays, supply chains or production alliances are characterized by several different partners with different functions within these associations. Important for reaching the common goal of added value is the connection of these networks. Therefore it is important to secure an efficient flow of materials as well as an efficient flow of information. Innovative logistics service providers can tackle these tasks by providing new technical solutions, for example offering the benefits of a real-time GPS track and trace system based on GPS. This paper shows an approach how to use GPS tracking data to identify discrepancies of standards in transport processes and to support flow of information reducing synchronization problems between different partners within a production environment characterized by low stock levels, short cycle times and the challenge of on-time delivery under these requirements. A piloting example using the Ild GPS.LAB in the logistics cluster of Duisburg/Germany highlights the possible uses and profits by GPS-based real-time transport control information and dynamic production scheduling.

PROBLEM DESCRIPTION
A production network is characterized by different players and their connections. Such associations are often described with nodes and links how it is displayed in figure 1 (Kumar and van Dissel 1996): Nodes standing for companies (e.g. production plants); links representing connections, generally as flow of material and flow of information.

The production control within such a node is often self-organized by the individual company. Therefore the goals of primary production planning can be controlled by use of a real-time control system, e.g. by analyzing performance indicators. The Association of German Engineers (VDI) provides several guidelines with performance indicators for logistics systems as VDI-Guideline 2525 which provides logistics performance indicators for small and medium enterprises or guidelines 4400 with logistics performance indicators for procurement, production and distribution (Weigert et al. 2010).

But how to control the flow of material in such a network? These logistics processes are often bottlenecks and important for the smooth and fluent run of production because delayed or interrupted supplies may cause shutdown of production if the worst comes to the worst. But it is a challenge to control these processes because they were often executed by sub-contractors and the control of bottleneck steps have always been important for the whole process (Hopp and Spearman 2001). To control these processes and to have a possibility to react to discrepancies, the implementation of a forwarder independent real-time GPS track and trace system within a production network is suggested.

CARGO TRACKING IN PRODUCTION NETWORKS

Stock-keeping in the shop floor has to be minimized to cut capital lockup in the form of material, which also supports a reduction of cycle time. Therefore it is necessary to safeguard just-in-time supply of material. In times of globalization, global supply chains with local separation of production plants and long transport routes in between are usual. An important problem is that a delay involves troubleshooting in production scheduling to avoid an interruption of the production process. A reliable flow of information is a basic requirement for today’s dynamic and flexible production and logistics activities. In fact this should be included in any software for supporting the production process (Meers et al. 2010). Telematics or cargo tracking in production alliances support the flow of information and improve the production system by reducing cycle-time, especially if tracking data is generated automatically and is transmitted in real-time to analyse data and to perform necessary actions in time (Brewer et al. 1999). Tracking data has to be continuously available at any
time independent of transport mode and company boundaries, reliable and secure as well as easy to transmit and cost-efficient for the producer (Stopka 2009). The problem of using a tracking system within a supply chain is the holistic integration of one system, especially if actors within the supply chain vary from time to time. So for these short-term multi-company networks an independent forwarder tracking solution is needed. By this idea the main difficulties of software integration could possibly be solved (Kärkkäinen et al. 2004). This could be a battery powered GPS tracking system. By tracking shipments or vehicles many information and features are offered or can be estimated to control and schedule production processes in real-time (figure 2). Similar research has been put forward in transportation research regarding personal transport and urban travel systems (O’Connor 1997; Wolf 2006; Tsui and Shalaby 2006; Wolf et al. 2003; Wolf et al. 2001; Wolf 2000).

![Figure 2: Benefits of cargo tracking](image)

To summarize the benefits production planners get a better support to synchronise the delivery of material and preliminary products with the production scheduling and get a longer time period to reschedule the production if needed.

**REAL-TIME TRANSPORT CONTROL**

The flow of material within a network can be displayed in a network plan. Such a plan offers planning and controlling possibilities of projects by dividing whole projects into small and single activities. Every activity is described with a defined start, an operating time and a defined ending. Because of upstream, downstream and parallel activities buffer times can be estimated (Domschke and Drexl 2005).

![Figure 3: Section of a network plan](image)

Figure 3 shows a section of a network plan: Operation B displays a transport process. Company I passes shipments at a defined finishing time to the forwarder. As a result the forwarder gets a time window for pick up the shipments and another time window for delivering the shipments to company II, which results from the predefined starting time of operations C. According to both, time windows and the operating time, the forwarder’s buffer time can be estimated.

The planned operating time of a transport process can easily be estimated with the help of route planning software to set up the transport processes within the network plan. By running the activities the planned times have to be controlled to analyze if the production process is in time and the planned due date can be reached. With the help of a real-time track and trace system the operating time of a transport process can be controlled by executing following algorithm:

```plaintext
1 For every transport process do
2 Request of actual position of shipment/vehicle
3 Route planning from actual position to destination
4 Comparism of planned arrival time (starting time downstream process) and estimated time of arrival (result route planning step 2)
5 If estimated arrival time is equal or less than planned arrival time then
6 Next request
7 Else
8 Production Re-Scheduling
9 End If
10 End For
```

This algorithm is based on the idea of supply chain event management and has to be executed automatically and continuously to safeguard delivery times and because of that production process.

**INTEGRATION OF REAL-TIME TRANSPORT CONTROL IN PRODUCTION PLANNING**

Since this year the FOM ild - Institute for Logistics and Service Management, Essen/Germany, operates a GPS based track and trace system by AIS Advanced InfoData Systems GmbH, Ulm, partly funded by the Ministry of Innovation, Science, Research and Technology of the German State of North Rhine-Westphalia. To use the system and to analyse the results, a laboratory has been installed.
under the name of GPS.LAB. One GPS tracking device has three major parts and is shown in figure 4:

- A GPS black box as central part of the system. The module is equipped with a GPRS SIM card for transmitting tracking data in real-time to a server. Transmission intervals can be set individually.
- A high performance GPS antenna for strong receiving power to calculate the run-time of signals even with low signal strength or inside of trailers and containers. This was already tested and is working very stable inside steel boxes and trucks.
- A high powered rechargeable battery for an independent electrical powering of the GPS module. With a fully loaded battery the GPS module can be operated at least 72 hours.

![Figure 4: GPS tracking device components](image)

By reason of the above mentioned specifications the GPS.LAB makes it possible to track goods down to the level of pallets, cases, cartons or ideally items during the whole transport to retrace the logistic process in detail and to analyse the performance of supply chains. By the use of the software map & guide by PTV Planung Transport Verkehr AG, Karlsruhe, the GPS performance data can be mapped as a basis for further analysis and calculations. In fact, with the help of such a system logistic processes can be analysed well-founded as the data is verified by real transport and not by simulation. Figure 5 shows the system setup.

![Figure 5: GPS.LAB system components](image)

The GPS.LAB will be used to test the idea of real-time transport control within production networks. Therefore transport processes in real production environments will be tracked to check whether the results are useful to integrate them into production planning systems as shown in figure 6.

![Figure 6: Holistic integration of a GPS track and trace system within a supply chain](image)

**TESTING EXAMPLE LOGISTICS NETWORK**

A business example in the logistic services industry was tested with DB Schenker in Germany (Kandel et al. 2011): In fixed route scheduled transport networks in groupage freight real-time information about delays and disruptions in transport schedules are of high value (Friedrich et al. 2001). Because subsequent transports as for example last mile transports after the longer main haul transportation usually during the night shift depends on on-time arrival at the inbound depot (Polyvyanaya 2011) as depicted in figure 7.

![Figure 7: Last mile tour planning problem as path-dependent problem from main haul in logistics](image)

Therefore the local depot (here: Duisburg in Germany) has the important scheduling problem of planning local transport tours without the real-time information about transport processes (main haul) destined towards the depot from other depots in Germany as for example Munich or Kassel (Lefebvre et al. 2007; Ramming 2002). This problem up to today is solved by simply allocating specific routing areas to fixed routes as shown in the following picture as standard routes A and B starting from the depot in Duisburg (Balsys et al. 2007; Daganzo and Sheffi 1977; Prato 2009).

In the here described testing case the inbound transport from the Kassel depot was delayed 55 minutes with several shipments for tour A and five shipments for tour B. in the traditional setup both trucks for boths tours A and B would have waited one more hour for the truck from Kassel to arrive in Duisburg (figure 8).
order to support new value added services based in this new and more exact service in the future.

Further applications as for example a suggeste mystery shipping concept (Klumpp et al. 2011) or with dangerous goods transport (Batarliiené 2007) or within other fields in the logistics industry (Forster 2005) as well as in mobile service applications (Batarliiené and Baublys 2007) could lead to additional benefits for production networks such as recognition and mitigation of typical transport disruption points in order to avoid delays in transport and production flows in such networks and to identify shortest and therefore most efficient routes (Azvedo et al. 1993; Jaraščienė 2007) for cargo traffic in a dynamic setting.

Also further steps in the expected integration of GIS, GPS, GSM and RFID technologies and services (Derekenaris et al. 2001) will bring important development and application options for the production and logistics sector, for example the integration of transport planning, warehouse management and production control based on ‘smart materials’, also known as ‘intelligent cargo’, in the supply chain.

REFERENCES


Kandel, C., Klumpp, M., Keusgen, T. 2011. „GPS based Track and Trace for Transparent and Sustainable Global Supply


BIOGRAPHIES

MATTHIAS KLUMPP studied economics and business administration after a vocational degree in logistics (Speditionskaufmann IHK, 1996) at University Leipzig and the IECs Strasbourg from 1995 to 1998 (Diplom-Kaufmann, Diplom-Volkswirt). Parallel to his professional consultant career in strategy management, logistics and education (zeb, conosco) he also obtained a PhD at University Leipzig in 2007 and started as a professor at FOM University of Applied Sciences.

He founded the Institute for Logistics and Service Management (ild) at FOM in 2009. He is member of the scientific committee for the German national excellence cluster “LogistikRuh” funded by the German national ministry for research (BMBF) and has done research or EU ERASMUS teaching stays in Australia, China, Finland, France, Netherlands, Norway, Spain, Turkey and the USA.

CHRISTOF KANDEL studied industrial engineering at University Duisburg-Essen since 2005. After achieving the bachelor degree he started the master course in 2008 with major courses in ‘product engineering’ and ‘technology and operations management’. He finished in 2010 with the degree ‘Master of Science’. During his study he worked at VOITH paper in Krefeld in the department of project management and afterwards at SIEMENS Energy in Duisburg in the department “procurement & logistics”.

After his studies he began to work at institute for logistics & service management at FOM in December 2010 as research assistant. His research interests are real-time production control, green logistics in particular green manufacturing and green routing as well as the calculation of CO₂ emissions and tour planning of e-vehicles. In addition, he is responsible for the GPS.LAB at ild.
SHAPLEY VALUE SIMULATION FOR ALLOCATING GHG EMISSION SAVINGS DUE TO LOGISTICS POOLING WITHIN ECR COOPERATIONS

Christian Wick
Matthias Klumpp
Christof Kandel
Institute for Logistics and Service Management (ild)
FOM University of Applied Sciences
Leinkugelstraße 6
D-45141 Essen, Germany
E-Mail: matthias.klumpp@fom-ild.de
www.fom-ild.de

KEYWORDS
GHG emission allocation, SHAPLEY value, logistics pooling, Efficient Consumer Response (ECR), cooperation

ABSTRACT

For a couple of years companies – and especially logistics transportation networks – are compelled by a number of new developments: Next to cost reduction efforts also considering ‘green thinking’ within logistics processes is of higher interest – constituting a critical success factor. Although companies are already legally forced to report sustainable – especially environmental – performance in certain extend voluntary reporting is important to several stakeholder groups. Currently, a couple of organisations work on standards for how to calculate and report greenhouse gas (GHG) emissions. If logistics pool their transport capacities – as e.g. within Efficient Consumer Response (ECR) cooperation networks – not only costs but also GHG emissions can be saved (‘eco-efficiency’). But up to the present the question of how to allocate such savings has widely been neglected. Thus, this paper provides a first problem solution on how to allocate GHG emission savings by the help of cooperative game-theory’s SHAPLEY value concept. Therefore, operations research and computer aided simulation seem helpful.

INTRODUCTION

As within recent years companies are affected by rising competitive pressure new ways of differentiation are needed in order to secure sustainable company growth (Winkler et al. 2006). Especially, logisticians are addressed by such developments: Customers become aware of green delivery strategies next to a minimal level of costs (Delay 2007; Fiksel 2009). In order to fulfil customer’s needs green strategies within logistics transportation networks are needed. Saving greenhouse gas (GHG) emissions is just one – but a major – issue that needs to be addressed in scientific research (cp. e.g. Beamon 1999; United Nations Framework Convention on Climate Change 1998; Wick and Klumpp 2010). Calculating carbon footprints is actually of highest interest, but a very complex subject (BSI British Standards 2008; Schmidt 2009; Wackernagel and Rees 1996). As no standard has been established yet a number of companies, governments and non-governmental institutions work on problem solutions (e.g. BSI British Standards 2008; WBCSD and WRI 2004). One topic that has absolutely not been mentioned within this context is the way of how to calculate and allocate GHG emission savings within logistics transportation networks. This paper provides a first outlook on future developments as logistics pooling continuously gains more importance within global competition. Next to cost-savings this strategy especially effects GHG emission savings as physical transportation processes decrease (Pan et al. 2010).

But one question that remains unanswered in this field is the question of how to report these GHG emission savings towards customers. If several companies cooperate, which of them is responsible for GHG emission savings and therefore allowed to report them (Young 2010)? Thus, this paper provides a first problem solution of how to allocate GHG emission savings within logistics transportation networks operating logistics pooling. Therefore, operations research modelling is used (Maloni and Benton 1997): The mathematical and formal concept of SHAPLEY value – part of cooperative game theory – seems to be appropriate. As the whole subject is affected by high complexity computer aided simulation is consequently needed.

LOGISTICS POOLING, EFFICIENT CONSUMER RESPONSE AND ECO-EFFICIENCY

As due to globalisation merging markets are more and more common both international and national companies are threatened by new market conditions – e.g. division of labour and cost pressure (Middendorf 2008). Next to fierce competition also rising customer expectations – e.g. acting sustainably – have to be addressed in order to ensure continuous company growth. Thus, companies are often no longer able to work independently: cooperative strategies within long-term partnerships are necessary.

One strategy that recently has become of high interest for logisticians is logistics pooling. Here, a number of partners cooperatively plan and optimise own and shared transport capacities. Not only transport route and capacity optimisation are addressed, but also cooperative use of warehouses and transshipment centres. The aim is to operate vehicles at full capacities and reduce empty trips to a minimum. Finally, cost reductions are the overall aim. Despite a number of possible benefits in cooperative transport and unit load optimisation very little research deals with corresponding
effects (cp. e.g. Corsten 2004; Klumpp and Jasper 2007; Seifert 2006).

Logistics pooling is a sub-strategy of the Efficient Consumer Response (ECR) concept. ECR is a broad management philosophy – predominantly applied to the interface of producers and traders – that consists of two scopes: ECR Supply Side and ECR Demand Side. Supply Side strategies – as e.g. logistics pooling – cover those activities associated with the concept of Supply Chain Management (SCM) especially aiming at cost reduction potentials. Demand Side strategies on the other hand are part of Category Management (CM) – i.e. marketing activities. In extended relationships companies cooperate putting the focus on customers’ needs (cp. e.g. Corsten 2004; ECR Europe 2006; Fernie 2004; Klumpp and Jasper 2007; Seifert 2006).

Nevertheless, for a couple of years trends exceeding cost- and revenue-thinking have become of higher interest. Due to the fact that a limited and short-term economic point of view does not meet the interests of all stakeholders, further considerations have to be taken into account. Recently, environmental – i.e. green – needs gained more attention both in literature and practice. As e.g. climate changes cause natural disasters, companies (will) have to develop common strategies by improving essential sustainable developments of economic activities (Lamsali 2006; Winkler et al. 2006).

Because companies are often afraid of rising costs and other disadvantages due to an implementation of environmentally friendly strategies it can be hypothesised that “... early or first movers, following the idea of Schumpeter’s pioneer profit, can achieve financial gains by introducing new goods or methods of production as innovative action results in monetary benefits” (Wittneben and Kiyar 2009). In other words: a company’s economical success – e.g. rising shareholder value (SHV) – can be influenced by the (early) adoption these sustainable – particularly green – issues (Franck 2008; Hutchins and Sutherland 2008; Lamming and Hampson 2008; Sen 2009). However, not only financial subjects are of interest. The continual improvement of intangible assets as e.g. company image has to be kept in mind as well.

One concept that broadly addresses just mentioned considerations is ‘eco-efficiency’ (Fiksel 2009; Sen 2009). In reference to the World Business Council for Sustainable Development (WBCSD) (2000) “Eco-efficiency is achieved by the delivery of competitively-priced goods and services that satisfy human needs and bring quality of life, while progressively reducing ecological impacts and resource intensity throughout the life-cycle to a level at least in line with the earth’s estimated carrying capacity”. If this verbal definition is expressed mathematically, one might shape the ratio as follows (WBCSD 2000):

\[
\text{Eco - Efficiency} = \frac{\text{product or service value}}{\text{environmental influence}}
\]  

As a product’s or service’s economic value (‘efficiency’) is nothing else than the ratio of revenues and costs, then eco-efficiency can also be formulated as:

\[
\text{Eco - Efficiency} = \frac{\text{revenues} \uparrow}{\text{costs} \downarrow} \div \text{environmental influence}
\]

ECR Supply Side strategies aim at a reduction of costs, while ECR Demand Side strategies aim at an increase in revenues. Thus, it is demonstrated that ECR Supply Side – here: logistics pooling – and ECR Demand Side strategies may positively correlate with environmental needs – here: GHG emission saving activities – resulting in higher eco-efficiency (Pan et al. 2010). Consequently, in a green context the concept of eco-efficiency becomes global part of the ECR concept.

This paper puts the focus on how to allocate GHG emission savings within cooperative logistics transportation networks using Shapley value. As this operations research’s issue is of high complexity computer aided simulation is necessary (Maloni and Benton 1997).

**PROBLEM STATEMENT**

Up to the present, mathematical methods operationalising the impact of business processes on the environment – especially on global warming – are very rare. A major question that remains unanswered in this context is the way of how to realistic calculate the environmental impact of certain companies respectively cooperative logistics transportation networks in terms of absolute GHG emissions. Moreover, the existence of inconsistent calculation approaches makes it almost impossible to benchmark carbon footprints of different companies (Olson 2010). As especially CO₂ emissions are likely to become a ‘new currency’ within the near future, a quick problem solution is necessary. Thereby, GHG emission calculation can be divided into two sections: (a) GHG emission measurement or estimation and (b) GHG emission allocation.

(a) An often used approach to measure or estimate GHG emissions is process analysis. It is characterised by a bottom-up approach (Wiedmann and Minx 2007). Therefore, it is necessary to identify all relevant processes that cause GHG emissions both within a company and even more important within the entire logistics transportation network. As there are lots of interdependencies among inter-company activities it has to be ensured that system boundaries are explicitly defined (Young 2010). Otherwise, the problems of under- and double-accounting could occur (BSI British Standards 2008; Schmidt 2009; Wiedmann and Minx 2007). After all processes have been mapped, their total impact on global warming has to be measured. If this is not possible in all cases – as e.g. due to the possibility that a certain product may be used in completely different ways or over varying time periods – reference values, i.e. generic values, are needed (Schmidt 2009). The final step is to sum up all GHG emissions set free due to all logistics network processes.

As this subject is a very young discipline just a few organisations as e.g. the British Standards Institution (BSI) (UK), the Carbon Trust (UK), the Department for Environment, Food and Rural Affairs (DEFRA) (UK), the DIN Deutsches Institut für Normung e.V. (D), the International Organization for Standardization (ISO) (CH), the World Business Council for Sustainable Development (WBCSD) (CH) and the World Resources Institute (WRI) (USA) have already discussed possible ways of standardisation. As process analysis is based on micro-economical data it can be regarded to be very
complex demanding for computer based simulation. In order to reveal accurate GHG emission results this method seems to be most appropriate. Considerations made by above-mentioned organisations underline this assumption.

(b) After GHG emissions have been measured or estimated, allocating them to single GHG emission objects – e.g. companies – is necessary. Thereby, especially the question of how to accurately respect system boundaries within logistics transportation networks occurs. As due to the application of cooperative strategies like ECR’s logistics pooling another problem needs to be solved.

The probably most known global GHG accounting and reporting standard is the ‘Greenhouse Gas Protocol’ provided by the World Business Council for Sustainable Development (WBCSD) and the World Resources Institute (WRI). In order to accurately define system boundaries their ‘concept of scope’ can be adopted. It consists of three scope categories standardised to achieve transparency and ensuring avoidance of double counting problems by different companies. Scope 1 represents all direct GHG emissions caused by “… sources that are owned or controlled by the [regarded] company …” (WBCSD and WRI 2004). Scope 2 then contains indirect GHG emissions generated by purchased electricity. Physically, these GHG emissions are set free outside the defined company’s boundaries. Finally, scope 3 refers to all other indirect GHG emissions caused by sources that are not under control of the company, e.g. GHG emissions emitted by third party logistics providers (WBCSD and WRI 2004).

While scope 1 and 2 GHG emission boundaries are a minor problem within logistics transportation networks, scope 3 GHG emission separation is a major one. Up to the present, there is no discussion of how to realistic allocate GHG emissions if transportation capacities – as e.g. vehicles – are cooperatively used. If there is no cooperation each company is able to separately measure and publish its scope 1 GHG emissions. If on the hand transportation capacities get pooled a shift of several input (resources as e.g. vehicles and staff) and output parameters (cost and GHG emission debits) is likely (mixture of scope 1 and 3 GHG emissions). This is an old problem always discussed when dealing with cooperative concepts like ECR (Seifert 2006; Wick and Klumpp 2009). In a two partner relationship it possibly might be that one partner is burdened with higher costs and GHG emissions while the other one is able to save costs and GHG emissions – compared to their recently non-cooperative work (Seifert 2006). Consequently, cost and GHG emission compensations are inevitable as otherwise there is no basis for cooperation (Dudek 2004). Thus, the question of how to border (Young 2010) and allocate the overall GHG emission savings within logistics transportation networks needs to be answered as companies are forced to publish their environmental commitments more than ever (Olson 2010).

One mathematical concept that has already been used to fairly allocate cost savings in n-person partnerships is **SHAPLEY value** (Maloni and Benton 1997; Thun 2005; Wick and Klumpp 2009). As each partner’s contribution to all possible cooperation formations is respected this concept seems to be appropriate allocating GHG emission savings in logistics transportation networks. But with a rising number of partners calculation becomes quickly difficult demanding for computer aided simulation.

Finally, the problem of missing GHG emission calculation standards – generally accepted both in theory and practice – was demonstrated. Consequently, there is need to solve this problem by developing global guidelines on this subject. Otherwise, there won’t be a feasible possibility to publish and benchmark fair and realistic carbon footprints of different companies – possibly implying distortions of competition (Olson 2010).

**SHAPLEY VALUE**

Introduced in 1953 by LLOYD S. SHAPLEY the **SHAPLEY value** is part of cooperative game theory. Compared to other cooperative game theory concepts – as e.g. the NASH solution – it is generally applicable in n-person games. It allocates a definite solution to each player. SHAPLEY demonstrated that his concept is the only one that satisfies the following four axioms (Holler and Illing 2009; Shapley 1953; Thun 2005):

1. **Symmetry**: All players are regarded as equal. I.e. if some players provide the same input, they also obtain the same output.

2. **Pareto-Optimality**: Due to an arrangement made all players will be awarded with an output that is superior to that they actually achieve (‘status quo’). Further, no player may achieve a better output without downgrading another player. The sum of all individual payoffs is equal to the overall payoff that is allocated.

3. **Dummy-Player**: If a player achieves the same output within all coalition formations possible – compared to the output gained by working independently – she or he will also just be awarded with the original output. I.e. marginal benefits achieved by player i in all possible coalition formations are constant and are equal to those achieved working in isolation.

4. **Additivity**: If a game is divided into sub-games, the sum of a player’s outputs within these sub-games is equal to that within the overall game. I.e. if there is a special coalition formation each player achieves the quota equal to the output within several sub-coalition formations.

The **SHAPLEY value** is mathematically formulated within equation (3):

\[
\phi_i(v) = \sum_{K \subseteq N, K \neq \emptyset} \frac{(k-1)!(n-k)!}{n!} [v(K) - v(K \setminus \{i\})],
\]

whereby:

- \( \phi_i(v) \) **SHAPLEY value** for player i,
- K Sub-coalition,
- k Number of players in K,
- N Coalition,
- n Number of players in N.
Moreover, possible sub-coalitions in N, i.e. K is proper subset of N (k < n). The term K, i expresses that for any player i all sub-coalitions K are included in calculation if they are a proper subset of N and if i is even part of it. [v(K) − v(K − {i})] represents the marginal contribution of player i to an existing coalition K. Furthermore, there are n! different combinations of how all players n may join coalition N. I.e., the probability of all possible orders joining an existing coalition is always the same. Thereby, exact (k − 1)(n − k)! combinations exist in which i joins the coalition lastly. Finally, Σ means that all marginal contributions of player i weighted by the term \( \frac{(n-k)!}{n(n-1)(n-2)\ldots(n-k)} \) have to be summed up, if i joins sub-coalition K (Holler and Illing 2009; Shapley 1953; Thun 2005).

CASE STUDY SIMULATION

In order to become more practical a brief fictitious case study approach is provided demonstrating high need for computer aided calculation in the broad field of operations research (Maloni and Benton 1997). Therefore, it is assumed that three logisticsian (L_i, L_j, and L_k) – providing road transportation services – cooperate. Each vehicle fleet causes an already measured amount of GHG emissions if there is no partnership. After a logistics pooling cooperation between L_i, L_j, and L_k has been initiated, the amount of GHG emissions decreases due to an optimisation in vehicle capacity use (Pan et al. 2010). Several values of absolute GHG emissions are needed: For all possible cooperation formations absolute GHG emissions in tons (t) per year are given as follows (sample numbers):

\[
v(\{L_i\}) = 60.00, \quad v(\{L_j\}) = 15.00, \\
v(\{L_k\}) = 25.00, \quad v(\{L_i, L_j\}) = 52.50, \\
v(\{L_i, L_k\}) = 68.00, \quad v(\{L_j, L_k\}) = 35.00, \\
v(\{L_i, L_j, L_k\}) = 82.50.
\]

Table 1: Marginal GHG emission savings/SHAPLEY values.

<table>
<thead>
<tr>
<th>permutations</th>
<th>marginal GHG emissions savings (in t/year)</th>
<th>L_i</th>
<th>L_j</th>
<th>L_k</th>
</tr>
</thead>
<tbody>
<tr>
<td>L_i/L_j/L_k</td>
<td>00.00</td>
<td>22.50</td>
<td>00.00</td>
<td>-05.00</td>
</tr>
<tr>
<td>L_i/L_j/L_k</td>
<td>00.00</td>
<td>00.50</td>
<td>17.00</td>
<td></td>
</tr>
<tr>
<td>L_i/L_j/L_k</td>
<td>22.50</td>
<td>00.00</td>
<td>-05.00</td>
<td></td>
</tr>
<tr>
<td>L_i/L_j/L_k</td>
<td>12.50</td>
<td>00.00</td>
<td>05.00</td>
<td></td>
</tr>
<tr>
<td>L_i/L_j/L_k</td>
<td>17.00</td>
<td>00.50</td>
<td>00.00</td>
<td></td>
</tr>
<tr>
<td>L_i/L_j/L_k</td>
<td>05.00</td>
<td>12.50</td>
<td>00.00</td>
<td></td>
</tr>
<tr>
<td>total (Σ)</td>
<td>57.00</td>
<td>36.00</td>
<td>12.00</td>
<td></td>
</tr>
</tbody>
</table>

SHAPLEY Value: 09.50 06.00 02.00

The first line in table 1 (L_i/L_j/L_k) illustrates that logisticsian L_i initially works independently. Consequently, no GHG emissions can be saved as there is no cooperation \([v(\{L_i\}) = 60.00] − [v(\{L_i\}) = 60.00] = 0.00\). Then, logisticsian L_j joins logisticsian L_i. Thus, an annual GHG emission reduction of 22.50 tons per year is achieved \([v(\{L_i, L_j\}) = 52.50] − [v(\{L_i\}) = 60.00] + [v(\{L_j\}) = 15.00] = -25.00\). As this reduction is only possible because logisticsian L_j gets in cooperation with logisticsian L_i the reduction amount is completely dedicated to logisticsian L_i. If finally logisticsian L_k joins the cooperation of the logisticsians L_i and L_j the new partner (L_iL_jL_k) gets charged with 5.00 annual tons of GHG emissions as L_j does not effect a marginal benefit in this constellation \([v(\{L_i, L_j\}) = 82.50] − [v(\{L_i, L_j\}) = 52.50] + [v(\{L_iL_j\}) = 25.00] = +5.00\). This does not mean that the overall three player coalition is inefficient. If all marginal savings (or debts) for all theoretical joining combinations are calculated and the sum for each player i is divided by the number of total combination possibilities (here: 3! = 6) positive savings (in tons per year) for all players are generated (SHAPLEY value L_i = 9.50, SHAPLEY value L_j = 6.00 and SHAPLEY value L_k = 2.00). Thus, an overall GHG emission saving of 17.50 tons per year is achieved (cp. sum of values in each line provided in table 1). Finally, it has to be outlined that the concept of SHAPLEY value is fair and realistic – as already demonstrated by the description of its associated axioms. Nevertheless, there are a number of barriers that need to be addressed. For instance, only GHG emissions are taken into account. Thus, cause (input) and effect (output) relations are neglected. If e.g. just one player invests in new technologies – implying less GHG emissions – it is the only player that has to be awarded. As faultless measurement of GHG emission can currently not be guaranteed, values that should be allocated are imprecise probably effecting calculation bias. Only if correct input parameters are given a correct allocation is possible. Moreover, calculating SHAPLEY value is very complex as all theoretical marginal GHG emission savings for all companies are needed. Depending on the number of players this leads to a high workload. Thus, cost-benefit analysis is necessary. Otherwise, considering marginal GHG emission savings is fair as different players contribute to different extents – SHAPLEY value provides a unique problem solution for each player.

CONCLUSIONS

This paper highlighted a number of aspects that logistics transportation networks are confronted with:

- Today in logistics operations not only economic (e.g. cost and revenue considerations), but also ecologic (e.g. green thinking) aspects become critical success factors.
In order to secure existence and sustainable growth, companies are often forced to implement new strategies: Cooperation concepts as e.g. ECR are one opportunity.

Green thinking has to be understood as chance of differentiation and not only as negative impact: The concept of eco-efficiency shows that economic and ecologic needs can be linked positively.

ECR’s cooperative strategy of logistics pooling is just one example for successful eco-efficiency actions.

Up to the present no global (reporting) standard for measuring and allocating GHG emissions exists. A number of organisations work on problem solutions.

In order to maintain cooperations a fair and realistic solution on how to allocate cost and GHG emission savings is needed. One concept that seems appropriate to solve allocation problems is SHAPLEY value.

As the application of SHAPLEY value within logistics transportation networks is very complex operations research and computer based simulation are necessary.

REFERENCES


DIN Deutsches Institut für Normung e. V. 2011. Methodology for calculation and declaration on energy consumptions and GHG emissions in transport services (good and passengers transport); German version prEN 16258:2011.


LOGISTICS PROBLEMS SIMULATION
A COMPUTER ASSISTED STUDY
APPLIED TO THE ANALYSIS
OF THE PRODUCTION SET EFFICIENCY

Magdalena Hajdasz
Dept of Construction Engineering and Management
Poznań University of Technology,
Piotrowo 5,
60-965 Poznań,
Poland
E-mail: magdalena.hajdasz@put.poznan.pl

Adam Marlewski
Institute of Mathematics
Poznań University of Technology,
Piotrowo 3a,
60-696 Poznań,
Poland
E-mail: adam.marlewski@put.poznan.pl

KEYWORDS
construction processes, modeling, simulation,
visualization, decision support system

ABSTRACT
With the increasing complexity of construction processes, there is a need to advance tools supporting decision making in construction management and scheduling. We developed an intelligent decision support system for construction planning and site management for monolithic technologies; we called it MoCCAS (as monolithic construction computer aided system). In this paper we present the work of a simulation module mag_811, which is the part of MoCCAS, by examining selected strategies where the correlation between the crew and equipment has been changed. The solutions generated by the mag_811 module demonstrate the effectiveness of visual simulation as a key assistant tool in handling the construction management problems.

INTRODUCTION
The construction process is a multi-faceted enterprise that involves various, interdependent stages, for instance feasibility and strategy, preconstruction planning and design, construction, implementation and maintenance. The multi-disciplinary and on-time nature of construction projects poses specific requirements for construction process planning and management. The complexity of the construction process is related to the fact that such an undertaking takes place within a diverse physical environment, and dynamic conditions, such as changing participants, interacting resources, interruptions, unforeseen weather conditions, or technical, material and financial constraints. Another challenge is concerned with the simultaneity of the construction activities, where any disturbance can have an effect on the performance of other parties, which may impact time and cost. Thus, the risk rate for the effective planning and management, as well as a successful execution of a construction project is very high. A variety of methods and tools, such as virtual reality approach, advanced simulation and visualization techniques, have been offered by researchers for problem solving across all project stages. Techniques of Virtual Reality (VR) and Virtual Environments (VEs) are useful tools that facilitate construction progress monitoring, coordination of work, deployment of construction resources, knowledge communication or decision making (e.g. Haiyan et al. 2010, Teizer et al. 2007, Sampaio et al. 2007, Kano 2006, Dawood et al. 2003). Recently researchers focused on the Augmented Reality (AR) animations of engineering processes, which provide a significant potential in visualizing and communicating construction operations (Behzadan and Kamat 2010). The significance of the simulation and visualization techniques, as efficient operational procedures for construction management, became widely adopted in advancing construction process planning and execution (e.g. Behzadan and Kamat 2011, Hajdasz 2008, Kamat and Martinez 2008, Hartmann and Fischer 2007, Chau et al. 2005, Hong et al. 2002, Marlewski and Hajdasz 2000). The benefits of knowledge management (KM) to construction industry and an integrated knowledge management system for construction projects have been widely discussed by Kanapeckiène (2010) and colleagues. Currently attention is devoted to furthering research in simulation systems with a long term vision to strive for a highly integrated and automated construction execution environment (Abou Rizk and Hague 2009).
In our study we focused on the issues of construction planning and construction site management. In this area we designed an intelligent decision support system and we implemented it as a software MoCCAS (still under development). We employed techniques applied in the expert system, which allow for gathering and drawing from the expert knowledge, which is of a great significance given the specific nature of construction processes. In our mag_811 module a decision maker reveals his/her requirements in a dialogue session. The software, drawing from the knowledge base and using the inference mechanism imitating the reasoning of the expert, generates variant solutions. Our mag_811 is concerned with monolithic processes that have to be in conformity with strict technological procedures and require advanced organizational skills of the construction project.
manager in dynamically changing conditions within time-limits. This paper is concerned with the selected aspects of the construction process modeling, simulation, and the analysis of the exemplary solutions generated by mag 811, which are followed by some concluding remarks.

**SYSTEM SCHEME AND CONSTRUCTION PROCESS MODEL**

The construction process, given its multi-faceted and multi-stage nature, is subject to research from diverse perspectives and approaches at a various level of detail. Our research concept is based on the synergistic combinations: the expert system, simulation and visualization techniques, as well as a computer algebra system. Fig. 1 presents a schematic overview of our research framework. The key objective is a combination of the selected elements of the production set (concrete mixers, buckets, pump, forms, cranes, crews), which need to be integrated and synchronized within the system according to the requirements revealed by the decision maker. His/her decisions give an impetus for the activity of the simulation module. The mag 811 module allows for experimentations and provides a range of scenarios that enable the decision maker to undertake actions adequate to the assumptions within a dynamic construction execution environment. For the purpose of our system a number of models had to be devised. Fig. 1 illustrates a generic model presenting processes under consideration: a cyclic nature of the monolithic process; correlations between elements; cycle time changes as the work progresses.

**EXAMPLE SIMULATION STRATEGIES**

Construction work can be executed in a number of ways, which have been described as schemes (sequences of activities named here as strategies). They state the aims, e.g. to keep a steady daily performance and rhythm, to meet the deadline for task completion, to efficiently use determined means of production. Here we present the analysis of the two strategies, which have the same aim: to execute the construction within a required time frame for task completion. This aim can be achieved by employing various organizational strategies. Fig. 2 displays solutions generated by the mag 811 module according to the strategy U, i.e., ‘the task completion according to the deadline (30 days)’, realized in two variants: U3 and U4. In both variants an object of the identical dimensions has been erected. Fig. 2 a) concerns the U3 strategy, where each concreters’ crew has been assigned to a crane. Fig. 2b) deals with the U4 strategy, where the crew is working constantly, which requires crew’s cooperation with more than one crane. As the height of the object increases, the crane needs more time to transport the bucket, which results in the efficiency decrease. Furthermore, due to the
criterion for a task completion by the deadline (30 days), a greater number of cranes is necessary. Thus, more cranes are required in order to meet the execution deadline and secure a steady work pace of the concrete crew. Fig. 2 illustrates graphs of the 24-hour efficiency, production set efficiency, concrete layer thickness, and curing time. Our mag_811 software generates files where the results of the analyzed scenarios are stored. It has to be pointed out that the presentation, in a format of the three related diagrams (see Fig. 2), vividly shows the dynamics of the process and captures the changes within the site conditions.

**GENERATED STRATEGIES: CASE STUDY**

Let us analyze the three variants of one strategy, which differ in the bucket capacity (see Fig. 3). There are significant differences displayed in the graphs of the 24-hour efficiencies for each variant. This reveals to what degree the daily work at the construction site is affected by slight changes in the production system (all the elements remain identical except for the bucket capacity). The changes in the 24-hour efficiency have an impact on the use of resources and financial expenditures. The efficiency shifts are rather unexpected and would be difficult to predict (see the interface of shapes in Fig. 3a and 3b). It is worth highlighting that our mag_811 graphical presentations are particularly useful in providing a more comprehensive insight into the complex nature of the construction process, as well as, in the identification and measurement of what, when, where and to what degree is subject to change. The complexity of the space within which the construction site manager operates is, to some extent, illustrated by Fig. 4. Let us compare the variants presented in Figures 4a, 4b, 4c which show surfaces of the 24-hour efficiency: Fig. 4a corresponds to the case where there is only one crane at work, while Figures 4b and 4c deal with the case where more cranes are in operation. The surface presented on Fig. 4a is smooth and easy to be described by a mathematical formula,

$$E_c = \frac{60 Q_b}{M + T + C + t_{cnc}}$$

where $E_c$ stands for the crane efficiency [m$^3$/h]; $Q_b$ denotes the bucket capacity [m$^3$]; $M, T, C$ are the operation times of cycles; 

$$M := \frac{Q_b}{q_m}(t_{fil} + t_{mix} + t_{emp}),$$

$$T := (\frac{2h}{v_{ver}} + \frac{b}{v_{hor}})k,$$

$$C := t_{fix} + t_{cnc} + t_{unf};$$

$q_m$ – the mixer capacity [m$^3$]; $t_{fil}, t_{mix}, t_{emp}$ – the time at which the mixer is filled up and is emptied, respectively, [min]; $t_{mix}$ – the time at which the concrete is produced in the mixer [min]; $t_{fix}, t_{unf}$ – the time at which the bucket is fastened and unfastened, resp., [min]; $h$ – the height to which the bucket is transported [m]; $b$ – the width of the object [m]; $v_{ver}, v_{hor}$ – the vertical and horizontal velocity, resp., [m/min]; $k$ – the coordinating coefficient $[-]; t_{cnc}$ – the time of the cooperation between the crane and the concreters’ crew [min]. The surfaces shown in Figures 4b, 4c are completely different from the surface visualized in Fig. 4a and both display discontinuities. The interruptions occur when, in order to secure the harmonization and work continuity, a new crane is introduced due to technological and organiza-
Fig. 3. Visualization of the day progress in considered models (three variants in strategies U3 and U4), in the coordination system: height (h), time (day), 24 hour efficiency (E)

Fig. 4. Visualization of surfaces presenting the efficiency of the production sets, 4a) when only one crane is operating; the surface is covered by the equation (1); 4b) and 4c) when there are more than one crane can be used, the strategies U3 and U4 are applied, respectively, in the coordinate system: height (OSW), strategy variant (determined by the type of bucket), 24 hour efficiency (E)

Tional requirements. The discussed surfaces are difficult to be described in an exact mathematical formula. These diagrams allow for a detailed examination of the strategy and precise estimation of the results when the values of parameters change. It has to be pointed out that the above analysis takes into account only a few parameters, while the mag_811 software can also deal with more complex cases. The mag_811 module is implemented in Pascal language, we also used the computer algebra system Derive 5 for Windows (Texas Instruments, Inc.) in order to produce algebraic formulas, to conduct symbolic transformations and to enrich the analysis of the results generated by this system.
CONCLUSIONS

This paper demonstrates the utility of the mag_811 software for a comprehensive construction planning and efficient site management in monolithic constructions. This software offers a wide-ranging insight into the subject matter, allows for fast generating variant solutions, and provides a wide spectrum of the information about the construction process. This software enables a comprehensive decision support by covering macro- (e.g. the chosen strategy) and micro- (e.g. the crew composition) issues. We would like to emphasize the originality of the mag_811 software marked by its unique structure, where the entry data (i.e. the preferences of the decision maker) revealed during the dialogue session in a verbal format, are transformed by the mag_811 with the use of the inference mechanism that imitates the expert reasoning. We have also offered an innovative perspective for the schedules comparison of various solution variants. Fig. 3 presents schedules visualized within a three-dimensional space (its coordinates are the day efficiency, time, height of the object under construction). Moreover, the visual simulation presented in this paper allows for a precise examination of the interdependencies between selected elements of the production set and operations while controlling various parameters. This simulation provides many rich data solutions and widens the decision space of the construction project manager.

REFERENCES


BIOGRAPHIES

MAGDALENA HAJDASZ. Associate Professor at Poznan University of Technology, Ph.D. (1998). Research interests: organisation and modeling construction processes, planning and scheduling, multicriterias analysis, expert systems, simulation and visualization techniques, knowledge management

ADAM MARLEWSKI. Associate Professor at Poznan University of Technology, Ph.D. (1973). Research interests: approximation theory, numerical methods, computer algebra systems, modeling and simulation of technological processes.
An Empirical Evaluation of Martins’ Algorithm for the Multi-Objective Shortest Path Problem

Gerrit K. Janssens
Hasselt University, Diepenbeek, Belgium

gerrit.janssens@uhasselt.be

and

Jose Maria Pangilinan
Saint Louis University, Baguio City, Philippines

joey.pangilinan@slu.edu.ph

KEYWORDS
Multi-objective optimization, Martins’ algorithm, shortest path problem

ABSTRACT
The Shortest Path Problem is a popular optimization problem in operations research due to its wide range of practical applications. In most cases a single objective is considered, while also the multi-objective case has useful applications. The algorithms by Martins is considered very efficient. This study evaluates this algorithm by comparing it to a brute force algorithm as a first step to develop evolutionary algorithms for the multi-objective case. Experiments confirm the strength of the Martins’ algorithm.

1. INTRODUCTION
The Multi-Objective Shortest Path Problem (MSPP) is an extension of the Shortest Path Problem that aims to find efficient (non-dominated or Pareto-optimal) paths from a source vertex to a target vertex with multiple objectives in a single execution. MSPP have applications in many different industries such as telecommunications, traffic routing, resource allocation and rapid response for different purposes like cost cutting and time management. Most studies on the evaluation of the performance of Martins’ algorithm for the MSPP are theoretical and mathematical in nature. A few empirical studies show the algorithm’s performance in terms of the number of solutions it generates on different test networks. However, recent empirical studies lack necessary comparative illustrations to show its performance against another search algorithm.

This paper presents a comparative performance evaluation of Martins’ algorithm against a brute-force algorithm in terms of the number of efficient solutions generated and execution time. The paper first describes a brief background on the MSPP, then gives a summary of several types of algorithms that compute efficient paths for the multi-objective shortest path problem. Based on these algorithms, the performance of Martins algorithm is evaluated against a brute-force algorithm using several test networks. Finally, the results of the experiments are interpreted in terms of solution sets and execution time.

2. BACKGROUND

Multi-objective Shortest Path Problem

Given a directed graph $G = (V, E)$, where $V$ is the set of vertices (nodes) and $E$ the set of edges (arcs) with cardinality $|V| = n$ and $|E| = m$ and a $d$-dimensional function vector $c: E \rightarrow [\mathbb{R}]^d$. Each $e$ belonging to $E$ is associated with a cost vector $c(e)$. A source vertex $s$ and a target vertex $t$ are identified. A path $p$ is a sequence of vertices and arcs from $s$ to $t$. The cost vector $C(p)$ for linear functions of path $p$ is the sum of the cost vectors of its edges, that is $C(p) = \sum_{e \in p} c(e)$ while $C(p) = \min_{e \in p} c(e)$ for maximin functions. Given two vertices $s$ and $t$, let $P(s, t)$ denote the set of all $s$-$t$ paths in $G$. If all objectives are to be minimized, a path $p \in P(s, t)$ dominates a path $q \in P(s, t)$ iff $C_i(p) \leq C_i(q)$, $i = 1, \ldots, d$ and we write $p \preceq q$. A path $p$ is Pareto-optimal if it is not dominated by any other path and the set of non-dominated solutions (paths) is called the Pareto-optimal set. The objective of the MSPP is to compute the set of non-dominated solutions that is the Pareto-optimal set $P(s, t)$ with respect to $c$.

The problem of the single-source $s$, single-target $t$ multi-objective shortest path is to find the set of all paths from $s$ to $t$ in $G$.

Algorithms for the MSPP

A variety of algorithms and methods such as dynamic programming, label selecting, label correcting, interactive methods, and approximation algorithms to name a few have been implemented and investigated with respect to the MSPP (Ehrgott and Gandibleux, 2000). The problem is known to be NP-complete (Garey and Johnson, 1979). It has been shown that a set of problems exist wherein the number of Pareto-optimal solutions is exponential which implies that any deterministic algorithm that attempts to solve it is also exponential in terms of runtime complexity at least in the worst-case. But some labeling algorithm studies (Gandibleux et al., 2006; Müller-Hannemann and Weihe, 2001) dispute this exponential behavior. They show that the number of efficient paths is not exponential in practice. Other authors avoid the complexity problem by developing new methods that run in polynomial time. For instance, Hansen and Warburton (Müller-Hannemann, 2001) separately developed fully polynomial time approximation schemes (FPTAS) for finding paths that are approximately Pareto-optimal. Interactive procedures (Coutinho-Rodriguez, 1999; Granat, 2003) similarly avoid the problem.
of generating the whole set of efficient paths by providing a user-interface that assists the decision-maker to focus only on promising paths and identify better solutions according to preference.

Martins’ algorithm (Martins, 1984) is a label setting algorithm that assigns for every vertex in its path permanent labels and temporary labels. The algorithm selects the minimum lexicographic label from all the sets of temporary labels and converts it to a permanent label, and propagates the information contained in this label to all the temporary labels of its successors. The process stops when there are no more temporary labels. Each permanent label corresponds to a unique efficient path. Martins algorithm ensures the computation of the maximal complete set of efficient paths from one vertex to all the other vertices of a network.

Gandibleux et al. (2006) extended the capability of Martins algorithm by modifying its dominance test to ensure the computation of the maximal complete set of efficient paths for min-max problems associated with the multi-objective networks.

Pangilinan and Janssens (2007) explored a Multi-Objective Evolutionary Algorithm as applied to the MSPP and described its behavior in terms of variety of solutions, computational complexity, and optimality of solutions. Results showed that the evolutionary algorithm is capable of finding diverse solutions to the MSPP in polynomial time.

Granat and Guerriero (2003) proposed an interactive procedure for the MSPP based on a reference point labeling algorithm. Their algorithm converts the multi-objective problem into a parametric single-objective problem whereby the efficient paths are found. The algorithm was tested on different grid and random networks and performance was measured based on execution time. They conclude that an interactive method, from their experimental results, is encouraging and does not require the generation of the whole Pareto-optimal set (which avoids the intractability problem). Likewise, Coutinho-Rodrigues et al. (1999) proposed an interactive method that incorporates an efficient k-shortest path algorithm in identifying Pareto-optimal paths in a bi-objective shortest path problem. The algorithm was tested against other MSPP algorithms on 39 network instances. They conclude that their k-shortest path algorithm performs better than other MSPP algorithms in terms of execution time.

Tsagouris and Zaroliagis (2006) provided a Fully Polynomial-Time Approximation Scheme (FPTAS) for the determination of an approximate Pareto curve for multi-objective shortest paths that significantly improves especially in the case of more than two objective functions. The study shows that it can be used to provide better approximate solutions to multi-objective constrained efficient paths, multi-objective constrained paths, and non-additive shortest paths.

Paixao and Santos (2007) proposed a ranking algorithm that solves MSPP. This ranking algorithm finds all non-dominated path solution between a source node to destination node based on ranking path procedure by applying a stop ranking condition which allows to determine the entire set of non-dominated paths at the very early stage of the ranking procedure.

Pinto and Pascoal (2010) developed a labeling algorithm that computes multi-objective shortest paths by restricting the set of arcs according to the bottleneck values in order to find the minimal complete set of Pareto-optimal solutions.

3. EXPERIMENTS

The experiments intend to compare the performance between Martins’ algorithm and a brute-force search algorithm. The algorithms are evaluated in terms of the number of efficient solutions and computation time for a single source-target multi-objective shortest path problem. The single source-target problem is different from the single source MSPP as it requires the computation of all efficient paths from a single source to a single target only and not from a single source to all other vertices.

Program Implementation

The implementation of Martins algorithm is based on Gandibleux et al. (2006). The algorithms are implemented using the C++ programming language, a desktop computer equipped with an Intel Core 2 Duo 2.56 GHz processor and two Gigabytes of RAM on a 32-bit operating system.

ALGORITHM 1: MARTINS’ ALGORITHM

Requires: G=(V, A) and C, the cost matrix for all arcs (ij) ∈ A

Ensures: All efficient paths from s to all vertices i ∈ V \{s}

li : is the label of vertex i
lti : is the entire list of temporary labels of vertex i
lpb : is the entire list of permanent labels of vertex i
zp,q,h : is the pth performance of a permanent label of vertex q in position h
Δ : is the dominance relation (if zΔ z’ then z is dominated by z’)
perf() : performance operator

Initialization

ltv, lpv ← \0, ∀ v ∈ V
ltv ← \{0, ... , 0, 1, 1, 1, 1\} (the latter two have no meaning in the start vertex)

Iteration

while (Uev ⊆ \0) do

Find the minimum lexicographic label in ltv, ∀ v ∈ V
ltv ← lex(Uev, ltv)

Move the selected label from the ‘temporary’ to the ‘permanent’ list ltp ← ltp \{lpv\}; ltp ← ltp U\{ltv\}

Store the position of label ltp from list of vertex j
hj ← location(lpj)

Label all successors of q

for all j ∈ \{q\} (q,j) ∈ A do

Compute lj, the current label of vertex of j
lj ← \{zp,q,h + c(q,j),..., zp,q,h + c(q,h), q, h\}

Verify that there is no performance of vertex j labels dominating perf(lj)

if (∃ lj′ ∈ \{ltv U lpj\} | perf(lj′) Δ perf(lj)) then

Store the label lj of vertex j as ‘temporary’
ltp ← ltp U \{lj\}

Delete all temporary labels of vertex j dominated by lj
ltv ← ltv \{lj′ ∈ ltv | perf(lj′) Δ perf(lj)\}

end if

end for

end while
The brute-force search algorithm used in the study is implemented as a depth-first search algorithm. The depth-first search starts at the root vertex and explores all possible successors before backtracking. The brute-force search algorithm lists all possible paths that lead to the target vertex. After the listing process, all dominated solutions are eliminated by dominance tests.

**Algorithm 2: Brute-Force Search Algorithm**

BruteForceSearchAlgorithm(G,v) (v is the source vertex)
Stack S : \(\leftarrow\) \(\emptyset\); (start with an empty stack)
for each vertex u, set visited[u] := false;
push S, v;
while (S is not empty) do
  u \leftarrow pop S;
  if (not visited[u]) then
    visited[u] := true;
    for each unvisited neighbor w of u
      push S, w;
  end if
end while
end for
for path \(p\) in S
  if (\(\text{perf}(l_i) \triangle \text{perf}(l_j)\)) then
    if \(\text{perf}(l_i)\) is dominated by \(\text{perf}(l_j)\) then
      Store the label \(l_i\) as an optimal solution
      \(l_i \leftarrow l_i \cup \{l_j\}\)
      Delete all temporary labels of vertex \(j\) dominated by \(l_i\)
      \(l_i \leftarrow l_i \setminus \{l_j \mid \text{perf}(l_j) \triangle \text{perf}(l_i)\}\)
    else
      if \(\text{perf}(l_i)\) and \(\text{perf}(l_j)\) is non-dominated then
        Store the label \(l_j\)
        \(l_j \leftarrow l_j \cup \{l_i\}\)
      end if
    end if
  end if
END BruteForceSearchAlgorithm()

**Test Networks**

Forty-five network configurations were randomly chosen from Gandibleux et al. (2006). The configuration is as follows: 15 networks with 50 vertices (5 networks of 5% density, 5 networks of 10% density and 5 networks of 20% density); 15 networks with 100 vertices (5 networks of 5% density, 5 networks of 10% density and 5 networks of 20% density); and 15 networks 200 vertices (5 networks of 5% density, 5 networks of 10% density and 5 networks of 20% density). Each edge of network has two cost values in the range \([1, 1000]\).

For each network configuration, the maximal complete set of the shortest paths from vertex \(v = 1\) to a randomly selected vertex was computed. The objective function is a minimize a (2-Sum) problem, i.e. min \(\Sigma_{v \geq r} c(e)\). The algorithms were run and given a 24 hours duration limit to complete. All network traversals that are still in-process after 24 hours were marked as an incomplete execution.

**4. Results and Findings**

Figure 1 and Figure 2 show the efficient paths found for a single source-target problem on five different 50-vertex networks using Martins’ algorithm and the brute-force search algorithm respectively. Martins’ algorithm was able to compute the maximal complete set of efficient solutions to the 50-vertex networks of different densities whereas the brute-force search algorithm can only compute the maximal complete set of efficient solutions to the 50-vertex networks with 5% density.

Figure 1 shows that the number of efficient solutions generally increases as the density of the test network increases. This is not the case for Network 1 wherein the number of solutions for the 20% test network is lower than the 5% and 10% networks.

![Figure 1 Efficient Solutions using Martins’ Algorithm for 50-Vertex Networks](image1)

Figure 2 shows the cardinality of the solutions sets of the test networks and it is evident that the brute-force algorithm failed to compute efficient paths when the densities of the networks are 10% and 20%. This means that there were no solutions found in a span of 24 hours.

![Figure 2. Efficient Solutions of the Brute-force algorithm for 50-Vertex networks](image2)
Looking at Figure 1 and 2, the number of efficient solutions and the efficient solutions to the 50-vertex test networks with 5% density found by Martins’ algorithm and by the brute-force algorithm are the same. Since the brute-force algorithm in this case computed for the complete set of efficient solutions, means that Martins’ algorithm also computed the complete set of efficient solutions for 50-vertex, 5% density test networks.

Figures 3 and 4 show solution sets to the 100-vertex and 200-vertex test networks obtained by Martins’ algorithm. No solutions were generated by the brute-force algorithm for these types of networks. Figure 3 presents the cardinality of the solution sets of efficient paths for the 100-vertex networks with 5%, 10%, 20% densities. It is observed that the number of efficient solutions increases as the density of each network increases.

![Figure 3 Efficient Solutions using Martins Algorithm for 100-Vertex networks](image)

Figure 3 Efficient Solutions using Martins Algorithm for 100-Vertex networks

Figure 4 shows the cardinality of the solution sets of efficient paths for 200-vertex networks with 5%, 10%, 20% densities. Again, it is observed that the number of efficient solutions increases as the density of each network increases.

![Figure 4 Efficient Solutions using Martins’ Algorithm for 200-Vertex networks](image)

Figure 4 Efficient Solutions using Martins’ Algorithm for 200-Vertex networks

All efficient paths in less than 4 seconds for all density configurations and execution time ranges from 0.02 to 3.31 seconds whereas the brute-force algorithm computes for efficient paths in the range of 16 to 5,823 seconds. With respect to Network 5 at 5% density, Martins algorithm computes for the solution set in 0.02 seconds while the brute-force algorithm requires 1.6 hours. The variability in the execution time of Martins algorithm is small while the variability in the brute-force algorithm is very large.

![Figure 5 Execution Time in Seconds using Martins’ Algorithm for 50-Vertex Networks](image)

Figure 5 Execution Time in Seconds using Martins’ Algorithm for 50-Vertex Networks

The execution times of the brute-force algorithm for the 10% and 20% density configurations cannot be shown since no solutions were found within a 24-hour execution time span. The zeroes shown in Figure 6 are equivalent to undetermined execution times.

![Figure 6 Execution Time of Brute-force algorithm for 50-Vertex Networks](image)

Figure 6 Execution Time of Brute-force algorithm for 50-Vertex Networks

Figure 7 and Figure 8 show the execution times of Martins’ algorithm for the 100-vertex and 200-vertex test networks respectively. It can be observed from Figure 7 and Figure 8 that the execution times increase as the network density increases. With respect to the 100-vertex test networks and considering all density configurations, Martins’ algorithm computes all efficient paths in 2.4 to 140.8 seconds. With respect to the 200-vertex test networks and all density configurations, Martins’ algorithm computes all efficient paths in 1.0 to 3.5 seconds. It is evident that the execution times of the 100-vertex test networks are higher than the execution times of the 200-vertex test networks. This may be due to the test problem i.e. efficient paths are computed
for only one pair of vertices instead of one source vertex to all other vertices in the network.

![Figure 7 Execution Time using Martins' Algorithm for 100-Vertex networks](image)

![Figure 8 Execution Time of Martins' Algorithm for 200-Vertex networks](image)

CONCLUSIONS

This study presents an empirical evaluation of Martins’ algorithm against a brute-force search algorithm for the single source-target multi-objective shortest path problem. The efficient solutions and execution times are the primary concern of the evaluation process. Several network configurations with varying sizes and densities were used to compare the performance of Martins’ algorithm and the brute-force search algorithm.

The results of the experiments show that Martins algorithm generates a complete set of efficient solutions to all network configurations whereas the brute-force algorithm only generates solutions for small-sized and low-density networks. Hence comparison of solutions between algorithms is defined by the limitations of the brute-force algorithm. With respect to Martins’ algorithm, the size of the solution set increases as size and density increases. In terms of execution times Martins algorithm is fast but its execution time is dependent on the density of the network i.e. it requires longer execution times for higher density networks.

Acknowledgment The authors would like to thank the group of Kevin Sumagit, CS students in 2010 at Saint Louis University, in Baguio, The Philippines for coding the algorithms and running the experiments.

REFERENCES


AN EXACT ALGORITHM FOR THE FULL TRUCKLOAD PICK-UP AND DELIVERY PROBLEM WITH TIME WINDOWS: CONCEPT AND IMPLEMENTATION DETAILS

Gerrit K. Janssens
Kris Braekers
Transportation Research Institute (IMOB) – Hasselt University
Universiteit Hasselt – campus Diepenbeek
Wetenschapspark gebouw 5, 3590 Diepenbeek, Belgium
E-mail: {gerrit.janssens,kris.braekers}@uhasselt.be

KEYWORDS
Exact algorithm, full truckload, pick-up and delivery, set partitioning

ABSTRACT
Intermodal goods transport is characterized by a main transport by rail, barge or seaborne vessel, preceded and followed by short in time but expensive road transport. Many times trucks pick-up or deliver a single container which leads to a full truckload vehicle routing problem in terms of economically efficient transport services. Time windows induced by the customer or due to external situations increase the complexity of an efficient planning by a logistics provider. An exact algorithm for this type of pick-up and delivery problem is developed and the details of its implementation are explained. A set partitioning problem is automatically generated and solved by means of the Lingo software.

1. INTRODUCTION AND LITERATURE REVIEW
Road transport is subject to several problems like congestion, environmental concerns and traffic safety. Intermodal transport is often put forward as a solution to these problems. It can be defined as the combination of at least two modes of transport in a single transport chain, without a change of container for the goods, with most of the route travelled by rail, barge or seaborne vessel and with the shortest possible initial and final journeys by road (Macharis and Bontekoning 2004). An important aspect of intermodal transport is the efficient planning of the pre- and end-haulage activities since these activities constitute a large part of the total costs. Pre- and end-haulage activities involve the transport of containers by road between a container terminal and customer locations. Some containers need to be picked up at a customer location and transported to the terminal, while others are located at the terminal and need to be delivered to a customer location. Time windows may be imposed on these transports. The problem is to find efficient vehicle routes performing all transportation tasks within their time window at minimum cost. Since it is generally assumed that trucks can only transport a single container at a time, routing problems for pre- and end-haulage can be classified as full truckload pick-up and delivery problems (Erera and Smilowitz 2008). When time windows are involved, the problem becomes a Full Truckload Pick-up and Delivery Problem with Time Windows (FT-PDPTW). In this paper, an exact algorithm for solving a FT-PDPTW is presented, which is solved by means of the Lingo software (Schrage, 2007). A review of the general Pick-up and Delivery Problem (PDP) can be found in Savelsbergh and Sol (1995). For an overview of research on less-than-truckload pick-up and delivery problems, the reader is referred to Parragh et al. (2008a, 2008b). In this section, research on full truckload pick-up and delivery problems is discussed. Gronalt et al. (2003) present four savings-based heuristics for a FT-PDPTW where full truckloads are transported between distribution centers. A tabu search heuristic for a FT-PDPTW with heterogeneous products and vehicles is proposed by Currie and Salhi (2004). Wang and Regan (2002) study a FT-PDPTW in the context of an intermodal container terminal. A time window partitioning method is used to solve the problem. Another full truckload pick-up and delivery problem in the context of an intermodal container terminal is introduced by Imai et al. (2007). A heuristic based on Lagrangian relaxation is presented. In a first phase, delivery customers are merged with pick-up customers, which may lead to substantial cost and time savings. In a second phase, trucks are assigned to these merged trips. Caris and Janssens (2009) extend the problem of Imai et al. (2007) to a FT-PDPTW by introducing time window constraints at the customer locations. A two-phase insertion heuristic and a local search procedure are proposed. In a subsequent work, a deterministic annealing algorithm is developed to solve the problem (Caris and Janssens 2010). In Section 2, related literature is reviewed. The problem formulation is presented in Section 3. In Section 4 the proposed exact algorithm is discussed. Finally, a small numerical example is presented in Section 5 and conclusions are drawn in Section 6.

2. SET PARTITIONING APPROACH
Exact algorithms, which guarantee that the best solution found is optimal, are very limited in their applicability due to the NP-complete nature of vehicle routing and scheduling problems. Heuristic methods explore only a small part of the solution space, require less solution time
but do not guarantee that the solution, if found, is a (near) optimal one. A class in between can be referred to as heuristics based on exact methods, as they are often variants of exact methods. The set partitioning approach falls in this class.

The set partitioning approach consists of two phases. The first phase of the approach is a construction phase. Let $C$ be a set with $|C|$ elements and let $S_j$ be subset of $C$ with cost value $c_j$. In the subset construction phase a number of subsets $S_j$ with cost $c_j$ are generated. The second phase is an optimization problem which selects a subset $CS$ of all subsets $S_j$ : (1) which are mutually disjunct, (2) with a union that is equal to the set $C$, and (3) with a minimal sum of cost values $c_j$ of the selected subsets in $CS$.

Algorithms for the first phase are specific for vehicle routing and scheduling problems (full truckload, time windows, ...), while algorithms for the second phase are general in nature. The set partitioning approach is a special structured integer programming formulation onto which a vehicle routing problem is mapped. The formulation was first introduced in vehicle routing by Balinski and Quandt (1964). Magnanti (1981) has shown that the vehicle flow formulation of the classical one depot vehicle routing problem may be rewritten into a formulation similar to the set partitioning problem.

In the classical VRP a feasible solution should satisfy a number of constraints like: (1) a vehicle is used at most once; (2) if a vehicle arrives at a customer, it must also depart from there; (3) the load of the vehicle should not exceed its capacity; (4) subtour elimination constraints and (5) integrality of the decision variables. Let $FT_v$ be the set of all feasible solutions for a particular vehicle $v$ with respect to the those constraints and let $r$ be an index to a feasible solution in this set. Such a solution $r$ for a vehicle $v$ is either a feasible route visiting at least one customer, starting and ending at the depot and satisfying the capacity constraint, or an empty route without customers and not departing from the depot.

For each vehicle $v$, exactly one route of $FT_v$ must be chosen. Let $x_r$ be a 0-1 variable indicating whether route $r$ is chosen or not. Then

$$\sum_{r \in FT_v} x_r = 1, \forall v \in V$$

Since each customer $c$ has to be visited exactly once, only one of the routes that contains $c$ in all sets $FT_v$ may be chosen. Let $a_{cr}$ be a 0-1 coefficient that indicates whether customer $c$ is visited in route $r$ or not. Then

$$\sum_{v \in V} \sum_{r \in FT_v} a_{cr} x_r = 1, \quad \forall c \in C$$

To each route $r$ a cost $c_r$ is assigned. Therefore the objective is the minimization of the total cost of the chosen routes and can be written as

$$\sum_{v \in V} \sum_{r \in FT_v} c_r x_r$$

The objective function with both types of constraints exactly forms the set partitioning problem.

In this exact formulation it is assumed that $FT_v$ is the set of all feasible routes for vehicle $v$. In many situations it is impossible to construct the entire set of feasible routes and only a subset of ‘attractive’ routes are considered. The construction of this subset is called the route generation phase. It turns the exact algorithm into a heuristic.

In terms of performance, it is worth considering the type of solution algorithm for the set partitioning problem. A general purpose integer programming code can be used, but as it does not exploit the special structure of set partitioning, it can be applied only to rather small problem instances. A number of special algorithms have been developed for solving the set partitioning problem so that larger instances can be solved. Sometimes information from the specific application, viz. the VRP, might be used in ordering rows in the tableau to speed up the solution process. An extensive description of the approaches can be found in Balas and Padberg (1976). Later some other algorithms have been developed by Albers (1980), Fisher and Kedia (1986), and Christofides and Paixão (1993).

The problem under study in this paper is a variant of the classical VRP. Any variant or extension of the classical VRP implies some changes in either the route generation phase or the optimization phase. The variant under study differs from the classical VRP in the following aspects: (1) time windows are imposed; (2) loading is both of the pick-up and the delivery types; and (3) a vehicle can be used multiple times during the planning period.

3. EXACT ALGORITHM

3.1 Algorithm logic

The full truckload routing problem is defined on a network $G=(V,A)$ where the customers are located at the nodes (set $V$). A special node $\{0\}$ is added and should be interpreted as the single depot. The set $V$ consists of the union of two mutual exclusive sets representing both the ‘delivery customers’ (set $V^D$) and the ‘pick-up customers’ (set $V^P$), i.e. $V = V^D \cup V^P$ and $V^D \cap V^P = \emptyset$. The members of $V^D$ are designated by $d_i$ $(i=1..n_d)$ and those of $V^P$ by $p_j$ $(j=1..n_p)$.

A route is defined as a finite sequence, starting and ending at $\{0\}$, of customers. A route is called feasible if it satisfies two types of constraints, i.e. logical constraints and temporal constraints. The logical constraints refer to the physical load as a truck is able to carry only one full load (container). A route contains either a single customer or multiple customers.

A feasible route can be grammatically described as:

- feasible route ::= {0} <feasible sequence> {0}
- feasible sequence ::= <single customer> | <multiple customer sequence>
- single customer ::= <delivery customer> | <pick-up customer>
- multiple customer sequence ::= <ordered customer pair> | <ordered customer sequence> | <feasible sequence> | ordered customer pair ::= {0} <feasible sequence> ordered customer sequence ::= <delivery customer> <pick-up customer>
Let \( C_2 \) be a set of customers \( \{c_1, c_2, ..., c_l\} \) of cardinality \( l \). The set \( C_2 \) may be projected into a finite set of sequences (routes). A procedure needs to be developed to generate the set of feasible sequences related to set \( C_2 \). At least one logically feasible sequence can be obtained from the set \( C_2 \) (i.e. the route consists only of single customer sequences). The logically feasible sequences, which have been generated, need to satisfy the temporal constraints too. The temporal constraints represent time windows within which the service at the customer’s site should start, and a depot time window (opening hours of the depot).

The output of the procedure is the required input for a set partitioning problem, which is formulated as follows:

\[
\min \sum_{i=1}^{z} c_i x_i \\
\text{subject to} \\
\sum_{j=1}^{n} a_{ij} x_i = 1 (i = 1, ..., z) \\
x_i \in \{0, 1\}
\]

The components of the optimization model should be interpreted as follows:

- \( i \) represents an index of a customer set, from which at least one logically-and-feasible sequence can be generated;
- \( c_i \) represents the minimum cost among the logically-and-feasible sequences generated from set \( i \);
- \( a_{ij} \) takes value 1 if customer \( j \) is visited in a sequence generated from set \( i \) and 0 otherwise;
- \( x_i \) is the decision variable which takes value 1 if the minimum-cost sequence from set \( i \) is included in the dispatching plan and 0 otherwise.

The algorithm solves the full-truckload vehicle routing problem with time windows to the optimal value. The procedure can also be used as a heuristic through reduction of the computational effort. This reduction may be realized in two ways as the procedure consists of two main phases: (1) a preprocessing phase in which all logically-and-temporal feasible routes are generated; and (2) the solution of the set partitioning problem. The procedure in the preprocessing phase works in an iterative way. An iteration \( l \) includes all operations related to the cardinality of the customer sets equal to \( l \), starting from \( l = 1 \) and increasing it in each iteration by 1. The procedure has a stopping criterion which is described below. By specifying, as a user parameter, a maximal cardinality, the number of generated routes is limited and makes the optimal algorithm a heuristic. Solving the set partitioning problem by means of a heuristic instead of the 0-1 programming optimal algorithm also makes the optimal algorithm a heuristic.

A customer set of size \( l \) is called live at size \( l \) if at least one logically-and-temporal feasible sequence can be generated from the set. The iterative procedure stops after step \( l' \) if no customer sets live at size \( l' \) can be found. During step \( l \) of the procedure, customer sets of size \( l \) are generated by the union of a customer set, live at size \( l-1 \), and an element (customer) not included in the set of size \( l-1 \). For this operation only information on the customer sets, live at size \( l-1 \), is required.

In order to generate logically feasible sequences from a set \( C_2 \), the set needs to be partitioned into singletons (single customers) and pairs (ordered customer pairs), and an ordering amongst them. The length of a sequence (number of trips in the route) is indicated by \( l_{ab} \) \((b = 1, ..., l')\), with \( |C_2|/2 \leq l_{ab} \leq |C_2| \). Let the set of logically feasible sequences, generated from \( C_2 \) be denoted by \( \text{Seq}(C_2) \). The elements of the set, which satisfy the temporal constraints, build up the set of logically-and-temporal feasible sequences, denoted by \( \text{TSeq}(C_2) \), for which holds \( \text{TSeq}(C_2) \subseteq \text{Seq}(C_2) \).

Let us call the newly generated customer set of size \( l \) consisting of the base set (live at \( l-1 \)) and the additional customer. The logical-and-temporal feasible sequences of size \( l-1 \) have been stored. The sequences to be tested for the set, consisting of the union of the base set and the additional customer, are generated by insertion of the additional customer at the head of the sequence, at the tail of the sequence, and at all places within the sequence. Two actions have to be taken: (1) a logically feasible sequence at size \( l \) needs to be generated; and (2) the temporal constraints have to be tested.

The logically feasible sequences to be generated depend on (1) the type of additional customer and (2) on its neighbour(s) in the existing sequences (or non-existing in case the route is initialized by a first customer). The type of additional customer is either a delivery customer (\( dnew \)) or a pick-up customer (\( pnew \)). The customer(s) served immediately before (resp. after) the additional customer is (are) indicated by \( pi \) or \( di \) (resp. \( pj \) or \( dj \)) in case of single customer and by \( (di,pj) \) (resp. \( (dj,pj) \)) in case of ordered pairs. Other customers, either before or after the immediate neighbours of the additional customer are not explicitly mentioned in the following rules:

- If the additional customer is the first customer to initialize a route

**Case 1:** delivery customer

\( \{0\} \ dnew \ {0} \)

**Case 2:** pick-up customer

\( \{0\} \ pnew \ {0} \)

- If the additional customer is at the head of the sequence (string left of additional customer is empty)

**Case 1:** delivery customer, followed by a singleton (remaining customers to the right are not explicitly indicated, only by …)

\( \{0\} \ dnew \ {0} \ dj \ {0} \ ...
\)

\( \{0\} \ dnew \ {0} \ pj \ {0} \ ...
\)

\( \{0\} \ dnew,pj) \ {0} \ ...
\)

**Case 2:** delivery customer, followed by an ordered pair

\( \{0\} \ dnew \ {0} \ (dj,pj) \ {0} \ ...
\)

**Case 3:** pick-up customer, followed by a singleton
Case 4: pick-up customer, followed by an ordered pair
{0} \(p_{new}\) {0} \(d_{j}\) {0} ...
{0} \(p_{new}\) {0} \(p_{i}\) {0} ...

- If the additional customer is at the tail of the sequence (string right of additional customer is empty)

Case 1: delivery customer, preceded by a singleton
... {0} \(d_{i}\) {0} \(d_{new}\) {0} ...
... {0} \(p_{i}\) {0} \(d_{new}\) {0} ...
Case 2: delivery customer, preceded by an ordered pair
... {0} \((d_{i},p_{i})\) {0} \(d_{new}\) {0} ...
Case 3: pick-up customer, preceded by a singleton
... {0} \(d_{i}\) {0} \(p_{new}\) {0} ...
... {0} \((d_{i},p_{new})\) {0} ...
... {0} \(p_{i}\) {0} \(p_{new}\) {0} ...
Case 4: pick-up customer, preceded by an ordered pair
... {0} \((d_{i},p_{i})\) {0} \(p_{new}\) {0} ...

- If the additional customer is within the sequence (strings right and left of the additional customer are non-empty)

Case 1: delivery customer, left and right are singletons
... {0} \(p_{i}\) {0} \(d_{new}\) {0} \(p_{j}\) {0} ...
... {0} \((d_{new},p_{j})\) {0} ...
... {0} \(d_{i}\) {0} \(d_{new}\) {0} \(d_{j}\) {0} ...
... {0} \((d_{i},d_{new})\) {0} \(p_{j}\) {0} ...
... {0} \((d_{i},d_{new})\) {0} \((d_{j},p_{j})\) {0} ...
... {0} \((d_{i},d_{new})\) {0} \(d_{j}\) {0} ...
Case 2: delivery customer, left is ordered pair, right is singleton
... {0} \((d_{i},p_{i})\) {0} \(d_{new}\) {0} \((d_{j},p_{j})\) {0} ...
... {0} \((d_{i},p_{i})\) {0} \((d_{j},p_{j})\) {0} ...
... {0} \((d_{i},p_{i})\) {0} \(d_{new}\) {0} \(d_{j}\) {0} ...
Case 3: delivery customer, left is singleton, right is ordered pair
... {0} \(p_{i}\) {0} \(d_{new}\) {0} \((d_{i},p_{i})\) {0} ...
... {0} \(d_{i}\) {0} \(d_{new}\) {0} \((d_{j},p_{j})\) {0} ...
Case 4: delivery customer, left and right is ordered pair
... {0} \((d_{i},p_{i})\) {0} \(d_{new}\) {0} \((d_{j},p_{j})\) {0} ...

Case 5: pick-up customer, left and right are singletons
... {0} \(p_{i}\) {0} \(p_{new}\) {0} \(p_{j}\) {0} ...
... {0} \(p_{i}\) {0} \(p_{new}\) {0} \(d_{j}\) {0} ...
... {0} \(d_{i}\) {0} \(p_{new}\) {0} \(d_{j}\) {0} ...
... {0} \((d_{i},p_{new})\) {0} \(d_{j}\) {0} ...
... {0} \((d_{i},p_{new})\) {0} \(p_{j}\) {0} ...
... {0} \((d_{i},p_{new})\) {0} \(p_{j}\) {0} ...
Case 6: pick-up customer, left is ordered pair, right is singleton
... {0} \((d_{i},p_{i})\) {0} \(p_{new}\) {0} \(p_{j}\) {0} ...
... {0} \((d_{i},p_{i})\) {0} \(p_{new}\) {0} \(d_{j}\) {0} ...
Case 7: pick-up customer, left is singleton, right is ordered pair
... {0} \(d_{i}\) {0} \(p_{new}\) {0} \((d_{j},p_{j})\) {0} ...
... {0} \((d_{i},p_{new})\) {0} \((d_{j},p_{j})\) {0} ...
... {0} \(p_{i}\) {0} \(p_{new}\) {0} \((d_{j},p_{j})\) {0} ...
Case 8: pick-up customer, left and right is ordered pair

3.2 File management

Input data
The input data related to the customers are read from two text files. A first file contains a matrix of distances (expressed as integers) from every delivery point to every pick-up point. A second file contains the distances to the depot both from delivery customers as from pick-up customers, as well as the earliest and latest start-of-service times (time windows).

Input data also come as parameters. These parameters are hard coded, so they are not read from an input file. They include:
- the name of the first input file
- the name of the second input file
- the number of pick-up customers
- the number of delivery customers
- the service time at the customer’s site
- the depot closing time.

The first and the second parameters are the names of the text files (including .txt). The third and fourth parameters are required because of the use of static data structures (vectors and matrices). A vector of length equal to the number of delivery (resp. pick-up) customers is used for (1) distances from the customer to the depot, and (2) earliest and latest start-of-service times at the customer’s site. A two-dimensional matrix is defined of size ‘number of delivery customers’ by ‘number of pick-up customers’ used for the distances between delivery and pick-up customers.

The fifth parameter assumes that the service time is equal at all customer’s sites. The sixth parameter refers to the time epoch at which any truck should return at the latest (assuming earliest time of departure from the depot is equal to 0).

Work file data
A customer set may have multiple sequences but a sequence belongs to only one customer set. This 1:N relationship might be used in a database programming environment but not in an environment using only sequential (text) files. Therefore the link between, customer sets and sequences must be explicitly coupled in a single data file. The program uses a data file type in which customer sets are defined and in which, after each customer set definition, the logically-and-temporal feasible sequences are defined. A customer set without logically-and-temporal feasible sequences is not included in the file.

As the procedure works in an iterative way, increasing the size of the customer set by 1 in each iteration, two work files of the same type are in use. A first file (called the old work file) contains all customers sets and their related sequences at level I-1 while a second file (called the new work file) contains all customers sets and their related sequences at level I.

Archive file
A customer set which contains at least one logically-and-temporal feasible sequence is a candidate for inclusion into
the set partitioning problem solution. During the iterative process these customer sets are stored. While creating the work file called the new work file it can be checked which of the feasible sequences lead to the lowest cost sequence. The value of this lowest cost is required for the formulation of the set partitioning problem. Also the sequence, related to this minimal cost value, is stored. This is not a requirement for solving the set partitioning problem, but the information is needed to publish the list of customers in a specific route which has been selected in the optimal solution of the problem.

Set partitioning problem file

Once the archive file has been completed and the procedure has satisfied its stopping criterion, a program translates the information from the archive file into Lingo-code for the formulation of the 0-1 integer linear programming problem which, as a text file, is ready to be solved by the Lingo software.

The algorithms have a bad worst-case complexity. In case the time windows are hardly restrictive a huge amount of subsets could be eligible as routes, making the set partitioning problem hard to solve. With relatively narrow time windows the complexity is much lower and the problem much easier to handle.

4. NUMERICAL EXAMPLE

In this section, a small numerical example with tree pick-up customers \( \{ V^P = \{ p_1, p_2, p_3 \} \} \) and three delivery customers \( \{ V^D = \{ d_1, d_2, d_3 \} \} \) is presented. The time windows of the depot \( 0 \) and customers are shown in Table 1. Distances between the customers and the depot and between delivery and pick-up customers are shown in Table 2. Customer service time is assumed to be 10.

Using the procedure presented in the previous section, eighteen feasible routes or sequences can be found. Six of these contain only a single customer, while there are nine sequences of two customers and three sequences of three customers. Solving the set partitioning problem results in the optimal solution which is shown in Table 3. Three vehicles are used and the total distance travelled is 416.

Table 3: Results

<table>
<thead>
<tr>
<th>Vehicle</th>
<th>Distance</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>99</td>
<td>0 - p_3 - p_4 - 0</td>
</tr>
<tr>
<td>2</td>
<td>220</td>
<td>0 - p_2 - p_3 - 0</td>
</tr>
<tr>
<td>3</td>
<td>97</td>
<td>0 - d_1 - p_2 - 0</td>
</tr>
</tbody>
</table>

5. CONCLUSIONS

The full-truck load pick-up and delivery problem is a relevant problem in an intermodal transport context to make the complete transport chain economically efficient. As most of the pre- and end- haulage in intermodal transport is realized by road transport and is relatively expensive, efficient vehicle routing is an economical benefit. The pick-up and delivery problem is NP-complete making optimal solution very difficult, especially when on top time window constraints are added. In this case researchers turn mostly into heuristics, but it might be also reasonable to investigate how exact algorithms behave computationally. The introduction of time windows probably limits the high computational complexity nature of the problem. In the case that the time windows are relatively hard constraining, an exact algorithm might be solved until optimality in a reasonable computing time. This paper shows how such an implementation can be realized in a computationally efficient way.

REFERENCES


BIographies

Gerrit K. Janssens holds a Ph.D. in Computer Science from the Free University of Brussels (VUB). Currently he is Professor of Operations Management and Logistics at the Hasselt University, Belgium within the Faculty of Business Administration. He also holds the CPIM certificate of the American Production and Inventory Control Society (APICS). During the last eighteen years he has been several times visiting faculty in universities in South-East Asia and in Southern Africa. His main research interests include the development and application of operations research models in production and distribution logistics.

Kris Braeckers graduated as Master in Business Economics with a major in Operations Management and Logistics at Hasselt University in 2008. Currently he is preparing a PhD in Applied Economic Sciences at Hasselt University. He is a member of the research group Logistics at the Transportation Research Institute (IMOB) of Hasselt University. His main research interests include modeling empty container management and vehicle routing issues in intermodal freight transport using Operations Research techniques."
Green Bullwhip Effect Simulation Concept

Matthias Klumpp
Institute for Logistics and Service Management (ILD)
FOM University of Applied Sciences
Leimkugelstraße 6
D-45141 Essen, Germany
E-Mail: matthias.klumpp@fom-ild.de
www.fom-ild.de

KEYWORDS
Green Logistics, Green Bullwhip Effect, Sustainability and Volatility Interdependence

ABSTRACT
Sustainable and green logistics concepts pose an important trend in global supply chains. An interesting research question in this field is the topic of a possible interdependence between green logistics concepts and overall and upstream progressive volatility in supply chains, discussed in logistics literature e.g. as ‘bullwhip effect’.

This research paper describes existing knowledge regarding the bullwhip effect and green logistics in order to proceed to a volatility simulation analysis of specific and relevant green logistics instruments to the whole supply chain. By this concept draft for a green bullwhip effect first glances on the possible quantitative effects in supply chains are possible.

INTRODUCTION
In logistics management and research one important objective is an understanding as well as the reduction of increasing order lot sizes through the supply chain (upstream) - the bullwhip effect (Agrawal et al. 2009; Beamon 1999; Campuzano and Mula 2011; Carranza Torres and Villegas Moran 2006; Chatfield 2004; Coppini et al. 2010; Corsten 2004; Helbing 2003; Hwang et al. 2005; Kelepouros et al. 2008; Lee et al. 1997; Metters 1997; Özelkan and Lim 2008; Paik and Bagchi 2007; Pati et al. 2010; Taylor 1999).

A standard bullwhip effect is shown in the following picture with a small increase in customer orders (right hand side) and increasing order lot sizes at the manufacturing and suppliers stage.

The reasons for this effect are information gaps, psychological biases (security) and insufficient co-ordination and planning among the supply chain partners (Chen 2000; Cho and Lee 2011; Jaksic and Rusjan 2008). Therefore countermeasures are directed towards enhanced information exchange and transparency as well as coordination and integrated planning approaches in the supply chain (Wright and Yuan 2008).

GREEN LOGISTICS CONCEPTS
Green logistics concepts are discussed in a manifold of literature contributions (Aronsson et al. 2008; DHL 2010; Murphy and Poist 2000; Murphy et al. 1996; Polonski 2001; Sommer 2007; Sundarakani et al. 2010; Trunik 2006; Wilson 2009) and entail a plethora of instruments and measures in order to reduce energy consumption and emissions by logistics processes.

The following figure 2 by the author gives an example of an integrated concept: Four implementation areas of definition and measurement (e.g. PCF, carbon emission calculation in logistics, global standards as e.g. GHG or BSI), reduction of energy input and emissions output in logistics, compensation as management area in terms of carbon compensation programs as well as the fourth perspective of global reach with the request of globally integrated transport concepts e.g. in sustainable multi-modal transports.

Figure 1: Standard Bullwhip Effect

Figure 2: Integrated Green Logistics Concept
But as current state of logistics research it has to be recognized that though many concepts are devised a lack of simulative analysis of unintended effects exists. The following conceptual description of a green bullwhip effect wants to contribute to that specific lack by describing one specific possible consequence of green logistics measures to the supply chain in general.

**GREEN BULLWHIP EFFECT**

Combining the two described theory fields leads to the question which impact green logistics concepts and instruments will have on traditional dynamics in supply chains, for example the specific research question addressed herein: “Will volatility and bullwhip effect amplitudes increase due to green logistics instruments?”

This hypothesis can be depicted in the following figure 3 by the author where the - excess - increase in order lot sizes for each step of the supply chain is expressed in colored fields.

![Green Bullwhip Effect diagram](image)

**Figure 3: Green Bullwhip Effect**

**CASE STUDY PREPARATION**

In preparation for a case study simulation not yet implemented the following thoughts have been established as conceptual framework - and have to be combined with new variables and assumptions regarding lead times and lot sizes in supply chain order processes in respect towards green logistics measures. The following table gives an impression of these impacts of different green logistics instruments.

<table>
<thead>
<tr>
<th>Green Logistics Instrument</th>
<th>Transmission Character</th>
<th>Influence on Flexibility</th>
<th>Influence on Volatility V</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Electric-driven trucks</td>
<td>Restriction of transport range</td>
<td>Negative influence due to shorter range</td>
<td>Increasing V due to feared shortages</td>
</tr>
<tr>
<td>(b) Reduction of empty tours (trucks)</td>
<td>Reduction of shipment intervals</td>
<td>Negative influence due to longer spacing</td>
<td>Increasing V due to feared shortages</td>
</tr>
<tr>
<td>(c) Slow steering (ships)</td>
<td>Longer travel period &amp; more ships needed</td>
<td>Negative influence - increased travel time</td>
<td>Increasing V due to feared shortages</td>
</tr>
<tr>
<td>(d) Use of biofuel (planes)</td>
<td>Change of speed and range</td>
<td>Positive influence due to higher range</td>
<td>Decreasing V due to less shortage fear</td>
</tr>
<tr>
<td>(e) Carbon dioxide emissions trading (airlines)</td>
<td>Reduction of flight intervals</td>
<td>Negative influence due to decrease in capacity</td>
<td>Increasing V due to restricted capacity and rising prices</td>
</tr>
</tbody>
</table>

Table 1: Green Logistics Instruments Impact on Supply Chain Volatility

Altogether it can be recognized that most green logistics instruments will cause an increase in volatility and therefore ceteris paribus to increasing order lot size levels. This could trigger quite significant increases in costs throughout the supply chain as known from quantitative standard bullwhip analyses and simulations.

The described case study is planned to be implemented in a simulation at the beginning of 2012 with first data from the EU airline carbon emissions trading system which started in 2011. Further modeling e.g. with GAMS is planned.

**CONCLUSIONS**

The presented results have established a first idea about the shape of a green logistics bullwhip effect in logistics. But such simulations and thereafter piloting tests in business practice are very much needed as many legislative steps as e.g. the carbon emissions trading costs for airlines are decided upon today but will reward changes in transport times, volatility and total supply chain costs in the far future. This could also enhance the overall evaluation of such instruments prior to legislative action and introduction in order to avoid unnecessary economic burdens in the future.

Further thoughts about the green bullwhip effect would include for example:

- Specific, time-sensitive industries as e.g. fashion and electronics industries with short lead and delivery times should reassess their supply chains and stock levels in order to avoid shortages and panic orders once green logistics instruments are going to be put in place. Maybe even specified simulation models regarding the quantitative effects of a green bullwhip effect should be conducted specifically for such industries.
- Second an overall strategy concept and simulation analysis for green logistics instruments should be implemented in order to assume better understanding of supply chain-wide consequences of green logistics instruments.
- Third further case studies and simulations are necessary in specific fields and industries in order to prove the existence of the described green bullwhip effect in business practice.

These examples and thoughts emphasize the important need for further research in this field of a green logistics bullwhip effect.

**ACKNOWLEDGEMENTS**

This research was partly funded by the NRW department for economic development and the EU Regional Development Fund for the research project “LOGFOR” (2009-2012).

**REFERENCES**


DHL 2010. Delivering Tomorrow, Bonn.


BIOGRAPHY

MATTHIAS KLUMPP studied economics and business administration after a vocational degree in logistics (Speditionskaufmann IHK Fulda) at University of Leipzig and IECs Strasbourg from 1995 to 1998 (Diplom-Kaufmann, Diplom-Volkswirt). Parallel to his professional consultant career in strategy management, logistics and education (zeb, conoscope) he obtained a PhD at University of Leipzig in 2007 and started as a professor at FOM University of Applied Sciences Essen.

He founded the Institute for Logistics and Service Management (ild) at FOM in 2009. He is member of the scientific committee for the German national excellence cluster “LogistikRuhr” funded by the German national ministry for research (BMBF) and has done research or EU ERASMUS teaching stays in Australia, China, Finland, France, Netherlands, Norway, Spain, Turkey and the USA.
GOAL BASED AGENTS IN MATERIAL FLOW SIMULATIONS - INTEGRATION OF AN AGENT PROGRAMMING FRAMEWORK IN THE DISCRETE EVENT SIMULATOR D³FACT

Alexander Klaas
Christoph Laroque
Hendrik Renken
Wilhelm Dangelmaier

Business Computing, especially CIM
Heinz Nixdorf Institute
University of Paderborn
Fürstenallee 11
D-33102 Paderborn, Germany
E-mail: {aklaas|laro|funsheep|whd}@hni.upb.de

ABSTRACT

The wide acceptance of material flow simulation as well as the growth of computational power has led to a constantly growing level of detail in simulation modeling in the area of manufacturing and logistics systems. With the growing level of detail, both the simulation model and the specific modeling of (cooperative) behavior become more and more complex. This comes especially into play regarding processes, where autonomous entities act within their specific environment according to an overall goal (group of forklifts, workers, e.g.). In this paper, we present our approach to integrate the BDI-based agent modeling framework GOAL with our material flow simulation tool d³fact, in order to overcome these modeling issues and provide an integrated framework with highly-detailed modeling capabilities. We describe our approach, implementation and realization using a small example and conclude the following steps of this development.

INTRODUCTION

Simulation is becoming an increasingly important tool for decision makers in various fields of application. In the design and control of manufacturing systems, very complex processes need to be represented in a simulation model in order to accurately predict a system’s performance. Discrete event simulation (DES) software, such as our own material flow simulation tool d³fact, is in most cases well suited for modeling the dynamic processes and their interactions within a manufacturing system.

However, DES lacks sophisticated methods for modeling the (often cooperative) behavior of specific entities acting in such systems, for example autonomous vehicles (AGVs) as well as human workers, which transport and handle goods. In the last years and with the continuous improvement of computational power, the models for simulation and analysis of these manufacturing systems became more and more detailed and thereby, the need for a comfortable method to model behavior is constantly growing. Here, agent based modeling is a suitable and widely used way in describing (cooperative) behavior. Solving the issue to extend and integrate the material flow simulation d³fact with agent based modeling abilities is the focus of our work presented in this paper.

As the behavioral description must specify, what the agent does in each state of the simulated system, it is infeasible to program the agent as a list of actions to take in each state due to the complexity of the possible scenarios. In the field of artificial intelligence, goal-based agents (Russell and Norvig (2009)) have been developed that allow to specify what an agent wants to achieve (its goals), rather than how.

Specifically, the Belief-Desire-Intention (BDI) model (Wooldridge (2002)) is a known method to program such goal-based agents. There exist a wide range of BDI-based framework implementations (Bordini et al. (2006)), which aid the user in specifying the agent’s behavior in standardized programming languages. The frameworks implement the reasoning system that build up the internal model of an agent’s environment and determine an action depending on the model and the specific goals, specified by the modeler. In sum, these features let us choose to integrate a BDI framework into our material flow simulator d³fact.

The main issue in realizing such an integration is that agent programming is typically done in a logic based, knowledge representation language such as Prolog (which is well suited for the reasoning system), while d³fact, like most other DES software, is written in an imperative programming language (in our case the object-oriented programming language JAVA). So, the main challenge is to allow the agents access to the environment in d³fact in such a way that only little work during modeling and little computational overhead during simulation is created.

We identified the GOAL Agent Programming Language (van Riemsdijk and Hindriks (2009)) as a suitable framework for our integration. Major advantages are: it is open source, provides an exchangeable communications middleware and works with the Java based Environment Interface Standard
(EIS) (Behrens et al. (2011)) for generalized access of an agent to its environment and methods for agent management. To be more specific, our major contribution can be found in the realization of a seamless integration and high usability of agent behavior modeling within d²fact. A small example will show later on, that only little work is required by a modeler to include entities with agent behavior in a simulation model. With the realized integration, simulation models as well as single entities of a simulation model may be equipped with agent behavior with a minimal amount of customization to the specific application scenario. In particular, our integration provides mechanism for the automatic initialization of the GOAL platform and creation of agents, if an entity with agent behavior exists in a simulation model. We developed a specific d²fact EIS extension that automatically creates and sends percept from within the d²fact simulation model to the corresponding agents in the GOAL environment. It also provides defined hook up points for the modeler in order to specify how actions by the agents are to be executed. Through the EIS extension, GOAL and d²fact have been connected and we additionally provide a novel synchronization between them.

Our integration is demonstrated in a demo application that should illustrate how to use GOAL agents within d²fact to model cooperative behavior. The simple demo includes four autonomously controlled vehicles working in cooperation to pick up goods. The corresponding application of this in real manufacturing systems could be, for example, a group of forklifts, controlled by individual drivers that try to maximize their output within a distribution center.

The following section provides an overview of the related work to be known for the later sections. Then, we describe the different components that constitute the agent integration in the Concept section. Consequently, details, how various issues were solved and how agents were integrated in an example, are given in the Details section. We give a summary and outlook in the section Conclusion.

RELATED WORK

This section reviews previous work regarding both DES and agent based software.

d²fact

In our discrete event simulation software d²fact (Renken et al. (2011)) we use a concept called composition and aggregation to remove the need for a static type hierarchy in our building block definitions. While static inheritance type hierarchies are simple to understand since it is natural for us humans to arrange objects in a taxonomy (Sommerville (2004), Shaw and Garlan (1996)), they can become hard to maintain because of the limited possibilities to enhance such a hierarchy. With the two patterns composition and aggregation a type hierarchy can be simplified or even completely removed (Gregory (2009), Deacon (2005)). The benefit of this concept is exposed when it is used to combine different subsystems into one object, while these subsystems were never meant to be combined.

In d²fact an object is represented by a dynamic set of 'properties'. While there are very different approaches to identify a set of properties, we use a generic container concept to sustain the object-oriented approach common in current simulation model architectures. The container type provides methods to manage its properties (add, delete, get by key, etc.). Simple properties like numerical values or strings are passive, meaning they do not react to state changes and also do not cause them. These properties are aggregated (the weak ownership): The container object owns them but they are not bound to the life cycle of the container object. The "logic" instead, is an active property, because it does react to state changes. E.g., when an event is caught, the "logic" initiates the processing of the event as a perception. Composition strongly binds a property to a container, which means the property is bound to the container’s life cycle and also receives a reference to it. Through this reference the property can access the container, the simulation core, the model and other objects within the model.

We are able to attach so called "listeners" to various objects and also properties. Through these properties, objects are able to get notified when various events or changes happen in the simulated system, for example when other properties are changed. The listener concept allows for handling dynamics in the system, as well as implementing measuring figures such as throughput of a simulation model.

Intelligent Agents

An agent is a concept that is defined by an encapsulated software program that perceives an environment through pieces of information called percepts, and acts upon that environment through actions (Russell and Norvig (2009)). Agents are usually designed to address some problem (their goal) that would be hard to solve with traditional software paradigms. They are characterized by four key features:

- **Autonomy**, there is no direct input from the user or other systems, agents act on their own.
- **Reactivity**, as they are able to react to changes in their environment.
- **Proactivity**, which means that through internal goal seeking, they proactively take actions.
- **Social Behavior**, as agents are able to communicate with others.

Communication allows the use of decentralized multi agent systems, in which agents work in a group. The definition does not specify how an agent is structured internally, and indeed many different kind of agents have been developed. We focus on goal-based agents, which have a concept of a goal to reach and are able to deduce actions from it, based on a continuously updated internal model of the environment.
Belief-Desire-Intention Model (BDI)

The Belief-Desire-Intention Model (BDI) allows programming of a goal-based agent in an intuitive and structured way, inspired by how humans reason (Wooldridge (2002)). An agent is structured in three main parts that form a thought process in succession, triggered by an input, as illustrated in Figure 1.

- **Beliefs**: A set of information representing the agent’s view of the world. The beliefs are stored by handling input with a belief revision function (BRF), which takes input and manipulates the belief set accordingly.

- **Desires**: The state of the world which the agent would like to achieve or be in and that can be optionally reached from the current set of beliefs.

- **Intentions**: From the set of desires, the agent chooses a particular one that it intends to reach after filtering. Intentions can be seen as high level plans that trigger a specific action to follow the intention.

![Figure 1: Schematic overview of the BDI model. Source: Wooldridge (2002)](image)

GOAL Agent Programming Language

The GOAL Agent Programming Language is a framework for programming BDI related agents (van Riemsdijk and Hindriks (2009)). It allows programming in different knowledge based languages, including Prolog (using SWI-Prolog for interpretation) and comes with an IDE for doing so. Additionally, agent management, communications (at the knowledge level) and Java API for control are provided. Programming an agent allows declarative specification of a goal. GOAL agents are defined in six sections that loosely correspond to the BDI structure:

- **perceptrules**: define how percepts change the current beliefs. They update the internal model to correspond to the updated environment.

- **beliefs**: the set current beliefs.

- **goals**: using the same language as the beliefs, explicit goals are defined.

- **program**: defines a strategy to choose actions based on the current beliefs and the goals.

- **actionspec**: defines the available actions with pre-conditions (when is it possible to execute an action?) and postconditions (what does the action change in the world?) using the same language as beliefs and goals. This allows the agent to determine which actions are available for execution and choose the best one, based on how each changes the internal model.

- **knowledge**: Static information regarding the environment and its laws, used by the other sections.

Environment Interface Standard (EIS)

Because “agents” are used in very different fields of applications, normally the term “environment” in which an agent works (gets his perceptions, interacts with) is intentionally undefined. This is where EIS comes into play. EIS was developed at the University of Delft (Behrens et al. (2010; 2011)) as a proposal for an environment standard that makes arbitrary environment models accessible by an agent. EIS does not make any assumptions on the environment or the agent platform or its implementation.

There are two main layers in EIS to work with: The Environment Management System is responsible for the initialization of the platform and for the environment model. It also starts the Agents-Entities System that maps the agents to some controllable entities in the environment model. To connect EIS to a given environment, the user can extend EIS Java classes and provide own methods to send percepts from, as well as execute actions in the environment.

There exists previous work in using GOAL in conjunction with EIS to use BDI agents in an untypical environment, for example to control artificial enemies in the Unreal Tournament video game (Digum (2011)). The authors were able to successfully connect to the game by extending EIS. The solution allows programming AI behavior at a high level, using actions such as “collect” that tells an agent to collect items in the world.

CONCEPT

Our approach aims to integrate the ability to conveniently model BDI based agents within our existing DES software d³fact, while allowing transparent access of an agent to the simulated environment. We address the issue of bridging the two modeling paradigms - Prolog based behavior descriptions and Java based simulation - by building on the EIS proposal. Our resulting integration solution is illustrated in Figure 2.

d³fact on the one side simulates the environment, which is represented by a simulation model. As a DES software, it essentially depicts processes with regards to time by executing a series of events. d³fact’s container based simulation model format (see related work) allows the user to equip an entity with agent behavior, marking it to be controlled by the specified GOAL program. For example, a truck’s low level functions, its physics and visual representation would still be
simulated by d³fact, a corresponding agent would control its actions however.

The agent behavior is written as a standard GOAL file in the Prolog based agent programming language. GOAL’s functions are to interpret the program and run the BDI steps to select actions, and also to provide communication between agents. The GOAL platform and the agents needs to be started and initialized from within d³fact, a task that the module “d³fact AgentBehavior” manages, using the GOAL API. This is done automatically whenever agents are used in a simulation model and requires no additional work by the modeler.

As a crucial component to bridge both GOAL and d³fact, we introduce the “d³fact EIS Extension”. It creates abstract percepts for the agents from the environment’s state, and acts out the agent’s chosen action by manipulating the environment and simulation entities accordingly. Per default, any changes in the d³fact simulation are automatically sent to GOAL as a percept, so that defining percepts for each agent type is not necessarily needed. It is merely required that the modeler defines the possible abstract actions for an agent type once in Java methods. For example, an action “brake” for a vehicle would have to be defined as a reduction of its acceleration value.

In order to extend an existing simulation model with agent behavior, the modeler simply needs to specify which entities are controlled by agents and define the possible actions. Transformation from the DES environment to the agent’s world and vice versa is done automatically, as well as all management and communication tasks. The modeler is able to focus on modeling the agent’s behavior in the powerful GOAL agent programming language, instead of dealing with integration issues.

Some remaining challenges with realizing the integration lie in the synchronization between advancing the d³fact simulation and performing the agent computations within GOAL, which do not rely on a compatible time model, and automatic linking between entities in the simulation model and the corresponding agent instances in GOAL.

The following section addresses those challenges and details how the integration was implemented.

DETAILS

In this section, we present our implemented agent management and integration solution in depth by walking through a developed demo application, which serves as a validation and showcase for the new agent modeling capabilities of d³fact. The purposely simple demo simulates a scenario in which four forklifts aim to collect a box in a 2D space on an 8x8 grid (see Figure 3). Notably, the behavior of each forklift is controlled by a GOAL agent. The agents are able to communicate with each other: they broadcast their distance to the box to each other (within the GOAL framework) and the forklift closest to it moves to pick up the box. The application displays the scene in a very simple 2D visualization.

Figure 2: Overview of the integration.

Figure 3: A screenshot of the demo application. In this situation, forklift 2 would move and pick up the box.

Figure 4: A simulation model with agent behavior. Regular and agent specific properties are illustrated.
ulation model, as outlined in the related work section, defines objects with certain active and passive properties. In our demo, the objects corresponding to the forklifts have an x and y position, as they are able to move in 2D, and an ID (Figure 4).

To use GOAL agents in a simulation model, we have developed two properties that integrate seamlessly in d³fact’s model structure (highlighted in Figure 4). The Entity property, which handles actions and percepts, attaches to simulation objects that shall be controlled or sensed by agents. The AgentBehavior property manages agents and attaches to the simulation model as a whole. Both Entity and AgentBehavior are active, composited properties of the simulation objects, as they react to changes, e.g. new percepts, while the position and ID properties are passive and aggregated.

Agent management

When a simulation model with the AgentBehavior property is loaded, routines within the property automatically initialize the GOAL platform through the Java API. It creates agent instances within GOAL by loading the appropriate APL program file, in which the agent behavior is defined. Furthermore, it instantiates the d³fact EIS Extension and through a series of method calls, connects that EIS instance to the GOAL agents. Finally, it is required that each agent is connected to the corresponding Entity property of the forklift object that the agent controls. As AgentBehavior is attached to the simulation model, it is able to automatically find all Entity properties and assign them by their ID.

It shall be noted, that all GOAL related code is encapsulated within the properties. This has the advantage that simulation models without agent behavior are loaded without initializing GOAL or even the need for GOAL to be installed.

Percepts

The Entity property serves as a hook up point between EIS and the d³fact simulation object. It automatically creates percepts (as an EIS Percept object) when the environment changes using the d³fact listener concept (see related work). Entity gets notified when another property, in our demo particularly a position property of forklifts and the box, was modified.

When the percept reaches the GOAL agent, it may update its beliefs (the internal model of the environment) as specified in its perceptorules. The forklift agents have a simple internal model - they store their own and the target’s positions as beliefs. Therefore, whenever an agent perceives a position change, it needs to update the value. The following percept rule (written in Prolog) achieves this for the agent’s x position. When X is the new perceived position, ß the old one stored as a belief, then ß is deleted and X stored:

```prolog
if bel( percept(posX(X)), posX(A) ) then delete( posX(A) ) + insert( posX(X) )
```

Actions

Actions that an agent can execute are defined in GOAL as actionspecs. As actions take place in d³fact, the modeler must specify in Java methods how an action affects the simulation model. The method returns a Percept as a direct result of an action. The forklifts have four actions, all with empty preconditions, as they may freely move in four directions (left, right, top, down) in space. For example, the action moveLeft decreases its x position by one. ThePosX property is modified (set to its old value minus one) accordingly in the following Java code:

```java
public Percept moveLeft() {
    getElement().setProperty(StringID("PosX"),
        (Integer) getElement().getProperty(
            StringID("PosX")) - 1);
    return null;
}
```

Note that the Java method name (moveLeft()) is the same as the action name in Prolog (moveLeft). This is crucial for the integration to work.

Actions are sent through the EIS as text strings during runtime and in order to execute them, the according Java method is called through Java reflections (by looking up the according method name to the string).

The goal (to reach the box) is expressed in Prolog as

```
atTarget :- posX(X, Y), target(X, Y)
```

meaning that the forklift is at coordinates X and Y, the same as the target’s. Other knowledge that the agent possesses is for example a sense of direction. When the x coordinate of the target is smaller than its own x coordinate, the forklift infers the information that the target is to the left:

```
targetIsLeft :- posX(X), targetX(TX), X<TX
```

Using the definition of the goal and additional knowledge, the program part of the GOAL agent defines a strategy to reach the goal. It selects an action to execute, based on the correct beliefs. The following Prolog code specifies that a forklift should move to the left if it has been assigned to pick up the box (hasJob), and the box is to the left:

```
if (bel( hasJob, targetIsLeft )) then moveLeft.
```

Synchronisation

Agents and DES software do not share a common time model. While GOAL works in rounds - in each round, all agents update their model based on any new percepts, and output an action - while d³fact works on explicit time models. Each event gets executed at a defined time. The events however are influenced by which actions the agents select, so that both time models intertwine.

For the integration, GOAL and d³fact work in alternating Turns. When d³fact advances its time, it lets GOAL perform one agent round. Whenever agents perform an action, d³fact is able to execute the action in a way that respects physical
laws and that may delay the effect and subsequent percept in its time model. In the current implementation of the integration, \(d^2\text{fact}\) increases the time in fixed increment. However, more flexible step sizes are possible by determining when agents will take an action before performing a GOAL round, and these methods are in development.

To realize alternating GOAL and \(d^2\text{fact}\) turns, which need to run in separate threads due to the software architecture, we use a semaphore that switches between the two threads. When \(d^2\text{fact}\) advances time, it releases the semaphore and the Java code that controls GOAL is able to run and trigger an agent round. As long as agent computations take places, it keeps the semaphore, and releases it afterwards so that \(d^2\text{fact}\) simulation may resume.

Communication

GOAL provides an exchangeable middleware for agent communication at the knowledge level. We used the provided local middleware for performance reasons as our application run on a single machine. Other alternatives provide features such as support for sending messages over network to distribute the agent system over multiple computers.

Defining the message content is done in the program section of the agents. The forklifts communicate to decide which one picks up the box. They exchange their own distance to the box whenever a new one has been perceived (and the according percept rule inserts the literal gotNewTarget):

\[
\text{if bel}(\text{gotNewTarget}, \text{id}(\text{ID}), \text{distance}(\text{D}))
\quad \text{then send(}\text{allother, iHaveDistance(ID, D)})\]

\text{ID} and \text{D} are bound to the agent’s stored \text{id} and the computed distance to the box and then sent using GOAL’s provided send method. The code is remarkably simple and easy to understand for a modeler. The advantages of using GOAL’s communication system are highlighted in this example. While in an own message system, appropriate ways to encode the information would be needed, communicating at the knowledge level enables transmission of abstract pieces of information such as the Prolog compound term \(\text{iHaveDistance}(\text{ID}, \text{D})\). Additionally, all addressing issues are taken care of by GOAL, the modeler simply uses the keyword \text{allother} and does not need to worry about whether other agents are run on the same machine or across a network.

CONCLUSION

In our paper, we present the successful integration of the BDI-based agent modeling framework GOAL in our material flow simulation tool \(d^2\text{fact}\). A simple example is used to describe some details about the implementation. The integration allows the modeler to specify complex cooperative behavior in a high level language with little work required to define access to the established DES environment or allow communication between the agents. The GOAL software is automatically loaded when needed and run in synchronization with \(d^2\text{fact}\).

The next planned steps are now to underline the applicability of our integration with a more “real-world” example and to continuously improve some aspects within today’s implementation. As one aspect, time synchronization has to be improved in future. Moreover, distributed computing of the agents (using communication across a network) to achieve a higher scalability of the system is another open area of research.

REFERENCES


WAREHOUSE SIMULATION
SIMULATION IN THE WAREHOUSE DESIGN AND MANAGEMENT CONTEXT: A SURVEY

Carla A. S. Geraldes
Algoritmi Research Center
School of Technology and Management
Polytechnic Institute of Bragança
Campus de Santa Apolónia, Apartado 1134
5301-857 Bragança, Portugal
E-mail: carlag@ipb.pt

Maria Sameiro Carvalho and Guilherme A B Pereira
Algoritmi Research Center
School of Engineering
University of Minho
Campus de Gualtar
4710-057 Braga, Portugal
E-mail: {sameirojgut}@dps.uminho.pt

KEYWORDS
Warehouse design and planning. Warehouse management. Logistics. Simulation.

ABSTRACT

The dynamic character of today’s competitive markets forces organizations to an incessant reassessment in an effort to respond to the continuous challenges. Warehouses as an important link in most supply chains should be an integral part of this effort. Hence, warehouses must continually be reevaluated to ensure that they are consistent with both the market’s demands and the management’s strategies.

The aim of this paper is to survey how simulation technique could represent one decision support system in the warehousing context.

INTRODUCTION

Market competition requires continuous improvement in the design and operation of supply chains (SC). A supply chain can be considered as a network of entities (Figure 1). The efficiency and effectiveness of any SC is highly determined by the performance of the entities in the network. In this context, warehouses play an important role in companies’ supply chains.

On the other hand, modern supply chain principles compel companies to reduce or eliminate inventory levels. Additionally warehouses require labour, capital and information technologies, which are expensive resources. So, why do we still need warehousing?

According to Bartholdi and Hackman (2006) there are four main reasons why warehouses are useful:

- To consolidate products in order to reduce transportation costs and to provide customer service;
- To take advantage of economies of scale;
- To provide value-added processing and
- To reduce response time.

Rouwenhorst et al. (2000) classified warehouse decisions into strategic decisions, tactical decisions and operational decisions. Strategic decisions are the long term decisions and always mean high investments. The two main issues are concerned with the design of the process flow and with the selection of the types of warehousing systems. Tactical management decisions are medium term decisions based on the outcomes of the strategic decisions. The tactical decisions have a lower impact than the strategic decisions, but still require some investments and therefore should not be reconsidered too often. At the operational level, processes have to be carried out within the constraints set by the strategic and tactical decisions made at the higher levels. At this level, the concern includes the operational policies such as storage policies, picking and batching policies. In short, strategic decisions create limits to decisions taken at the tactical and operational levels and tactical decisions create limits to operational decisions.

Once warehouse decisions are strongly interrelated, design, planning and control of warehousing systems is a highly complex task where frequently conflicting objectives impose specific trade-offs.

Despite the various decision models available in scientific literature, the majority addressed isolated or simplified problems solved with analytic or heuristic methods in order to provide the best solution. However, most real problems are unfortunately not well-defined and often cannot be reduced to
multiple isolated sub-problems. Therefore, given the associate complexity and the stochastic nature of these systems simulation can be considered as a powerful technique to apply, as a decision support system, within a warehouse design and planning context.

Simulation modelling (in particular, discrete-event simulation) is a common paradigm for analysing complex systems. In fact, it is possible to reproduce and test different decision-making scenarios in order to determine, in advance, the level of optimality and robustness of a given strategy.

The aim of this paper is to explore how simulation technique has been used as a decision support tool to help practitioners and academic researchers in a warehouse design and planning context.

After an introduction to warehouse design and planning decisions, a review is proposed in order to analyse the scope of use and the main benefits reported from the adoption of the simulation technique. Finally some considerations are provided.

WAREHOUSING

According to Van den Berg and Zijm (1999) and Rouwenhorst et al. (2000) warehousing processes concerns those material handling activities that take place within the warehouse functional areas, i.e. receiving, storage, order-picking, accumulation and sorting, and shipping of products.

Figure 2 shows the typical functional areas and flows within warehouses.

![Diagram of warehouse functions and flows]

Figure 2: Typical warehouse functions and flows

At the receiving area products are unloaded and inspected to verify any quantity and quality inconsistency. Afterwards items are transferred to a storage zone or are placed directly to the shipping area (this is called a cross-docking operation). We can distinguish two types of storage areas: reserve storage area and forward or picking area. The reserve area is the place for products to stay until they are required by costumers’ orders. The picking area is a relatively small area typically used to store fast moving products. Most of the flows between areas are the result of replenishment processes. Order picking is one of the most important functions in most warehouses. Stock keeping units (SKU) are retrieved from their storage positions based on customers’ orders and moved to the accumulation and sorting area or directly to the shipment area. The picked units are then grouped by customer order, packaged and stacked on the right unit load and transferred to the shipping area.

The design, planning and control of warehouse systems include a large number of decisions from a functional description, through a technical specification, to equipment selection and determination of the layout, and to the definition of operating policies. The overall structure decision determines the material flow patterns within the warehouse, the specification of functional areas and the flows between areas. Sizing and dimensioning decisions determine the total size of the warehouse as well as the space allocation among functional areas. Layout definition is the detailed configuration within a functional area and equipment decisions define an automation level for the warehouse and identify equipment types. Finally operating policies refer to storage, picking and routing decisions.

A brief description of problem category is given in Table 1.

<p>| Table 1: Description of warehouse design and operations problems (adapted from Gu et al., 2007) |</p>
<table>
<thead>
<tr>
<th>Design and Operation Problems</th>
<th>Decisions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Structure</td>
<td>Material flow</td>
</tr>
<tr>
<td></td>
<td>Functional areas</td>
</tr>
<tr>
<td>Sizing and Dimensioning</td>
<td>Warehouse size</td>
</tr>
<tr>
<td></td>
<td>Size of the functional areas</td>
</tr>
<tr>
<td>Equipment Selection</td>
<td>Level of automation</td>
</tr>
<tr>
<td></td>
<td>Storage equipment selection</td>
</tr>
<tr>
<td></td>
<td>Material handling equipment selection</td>
</tr>
<tr>
<td>Layout</td>
<td>Aisle orientation</td>
</tr>
<tr>
<td></td>
<td>Number, length and width of aisles</td>
</tr>
<tr>
<td>Receiving &amp; Shipping</td>
<td>Dock assignment</td>
</tr>
<tr>
<td></td>
<td>Dispatch schedule</td>
</tr>
<tr>
<td>Storage</td>
<td>Assignment and allocation of SKU to functional areas</td>
</tr>
<tr>
<td></td>
<td>Storage location assignment</td>
</tr>
<tr>
<td>Order picking</td>
<td>Batch size</td>
</tr>
<tr>
<td></td>
<td>Picking policy</td>
</tr>
<tr>
<td></td>
<td>Routing policy</td>
</tr>
</tbody>
</table>

The big challenge for both academic researchers and
warehouse practitioners is to get flexible structures capable to respond to changing conditions, to improve space utilization and to reduce congestion and movement.

LITERATURE SURVEY

The presented survey has been conducted over the scientific literature in order to establish which general objectives simulation is called to solve within the warehousing context.

Reader may note that the survey considers references that involve decisions related with: (i) Warehouse Design; and (ii) Warehouse Operations.

Warehouse design

Gray et al. (1992) developed an integrated approach to the design and operation of an order-consolidation warehouse. This approach included warehouse layout, equipment and technology selection, item location, zoning, picker routing, pick generation list and order batching. Due to the complexity of the overall problem, they developed a multi-stage hierarchical decision approach. The hierarchical approach used a sequence of coordinated mathematical models to evaluate the major economic trade-offs and to reduce the decision space to a few alternatives. Then, authors used simulation technique for validation and fine tuning of the resulting design and operating policies for a typical order-consolidation warehouse. A simulation model, written in SIMSCRIPT language, was validated using the analytical results, and in turn provided support for the approximations made in developing the mathematical models. Authors also compared the simulation results with the outcomes of the analytical models and concluded that the proposed approach was robust and accurate. The proposed approach was illustrated using a case study of an automotive spare-parts distribution centre.

Caron et al. (2000) presented a simulation approach to efficient layout design of the picking area in picker-to-part systems. The aim of this study was to compute the expected travel distance per tour considering both within aisle and cross aisle travel. The simulation model, written in Visual Basic, was essentially divided into five steps:

1. System definition;
2. Generation of samples to pick;
3. Definition of the picking tour for each sample;
4. Computation of the picking travel distance, and
5. Statistical analysis of the output data.

For the purpose of the study authors assumed that the storage capacity of the picking system was given and, consequently, the total length of the picking shelves was known. Experimental results showed that: (i) the expected picking travel distance may be significantly influenced by the number of aisles; (ii) layouts with a minimum number of aisles are preferred for a large number of picks; (iii) multi-aisle layouts are preferable for few picks per route and (iv) for intermediate number of picks the number of aisles is less crucial.

Ekren and Heragu (2009) presented a simulation based regression analysis for rack configuration of an autonomous vehicle storage and retrieval system (AVS/RS). The authors developed a mathematical function for rack configuration of an AVS/RS that reflects the relationship between the response and the factors of the system. The simulation model, developed using ARENA® 12.0, was used to determine the independent input variables (factors) levels of the regression model.

Warehouse operations

Caron et al. (1998) evaluated and compared the expected travel distance for different routing strategies in low-level picker-to-part systems. The expected travel distance was first evaluated by means of analytic models taking into account the main systems parameters. Then, a simulation model of the picking system, written in Visual Basic, was used to test the accuracy of the analytic model. The simulation model was used to calculate the length of randomly generated routes according to the chosen routing policy.

With the purpose of maintaining the picker’s regular workload Jane (2000) proposed two heuristic methods: one enlarged the order picking system by a new zone during the picking periods and the other reduced the system by one zone during the slack periods. The proposed methods were verified through empirical data and simulation experiments using Visual FoxPro 5.0 and Microsoft Excel 97.

Liu (1999) used a simulation study, using WITNESS simulator, in real word data to reveal the benefits of using clustering techniques for optimizing the storage policies and the picking process. The simulation studies revealed that the storage policy obtained by clustering techniques could significantly reduce the travel distance or time for picking operations within a warehouse.

Lin and Lu (1999) proposed a two-phase procedure to determine order picking strategies. An analytic method was first employed to classify all customers’ orders into five different categories. Then, computer simulation was used to generate the appropriate picking strategy for each orders category. ProModel simulation software was used in this study.

Macro and Salmi (2002) developed a simulation model to quantify rack utilization and storage capacity of different configurations of pallet racks in a warehouse. Using ProModel simulation language, the authors developed a universal warehouse storage simulation model that can be used as a tool to analyse existing warehouse systems, and as a method to experiment with different storage and process options. In particular, the simulation model can be used to answer the following concerns:

- Warehouse capacity due to demand growth;
- Efficiency of the storage policy;
- Equipment and resources limitations;
- Performance of the order picking policy;
• Effect of the production run size.

The proposed model has successfully been used in two different cases: (i) medium volume, low SKU warehouse and (ii) medium volume, high SKU warehouse. For each model three significant output metrics were calculated, namely: rack utilization by rack type and the average and maximum number of pallets in storage.

Petersen and Aase (2004) examined storage, picking and routing policies simultaneously. Authors’ aim was to evaluate several storing, picking and routing policies to determine which policy or combination of policies provided the greatest reduction in total picking time when compared to an existing baseline scenario. For that purpose a simulation model was used based on the operations of a given distribution centre.

Hwang and Cho (2006) developed a systematic approach for the performance evaluation of order picking warehouse system considering mathematical and simulation methods. A mathematical model was proposed to analyse the performance of the order picking system including the performance of the transport system. The developed performance evaluation model was based on the material handling cost for both cases of throughput and waiting due to congestion. Then, a simulation model using AutoMod simulator was used to validate the mathematical model.

Gagliardi et al. (2007) presented a discrete event simulation model that focus in the order picking and shelves replenishment processes and, as a first application, was used to discuss the impact of storage space allocation on the system performance. The simulation model has been coded in VB.net (Visual Basic 7.0) language and eight working weeks of a real high throughput warehouse which handles more than 12 million of case annually were simulated. The gathered results showed a large variability which did not allowed identifying a dominant rule.

Gopakumar et al. (2008) using ARENA® 10.0 developed a simulation model that was used as a tool to analyse an existing warehouse incoming receiving operations. Specifically, the model was proposed to evaluate methods to allocate docks to the incoming trucks in a systematic and efficient manner.

Kofjač et al. (2009) presented a real case study of inventory optimization in an automotive company. An anticipative approach was performed using simulation models supported by inventory control algorithms on selected samples of representative items. The simulation model, built using Matlab, yielded significant changes. The authors also considered that the simulator’s transparency and visual presentation can play a significant role in the learning process of using methodologies that they are unfamiliar with.

Recently Chan and Chan (2011) presented a real case study regarding the storage assignment problem of a manual-pick and multi-level rack warehouse. Authors’ aim was to determine which combination of factors improved the performance of the storage assignment policy of a manual-

pick warehouse in terms of travel distance and order retrieval time through a simulation approach. For that purpose three models of the same warehouse layout were built, using ARENA® language, with respect to three different routing policies, and twenty-seven combinations of factors (i.e. 3 storage assignment policies x 3 routing policies x 3 pick densities) were simulated. Experimental results were conducted on a multi-level rack warehouse. The gathered results showed that different combination of factors have different performance under different performance indicators. For example, for the order retrieval time and travel distance performance indicators, the performance of each combination of factors can be different. The one which performs better in terms of one performance indicator can result in the worst for the other indicator. For example, this study showed that horizontal ABC class-based storage has opposite performance under the two indicators.

SURVEY ANALYSIS

From the presented literature survey, it is possible to say that simulation has mostly been used as a mean to validate the performance of warehouse design and operations policies. In particular, some of the operational research models and results reported in the scientific literature were validated upon simulation approaches. For example, a mixed optimization-simulation approach is found in Gray et al. (1992), Liu (1999) and Hwang and Cho (2006).

It is also important to notice that the number of references concerning warehouse operations is bigger than the number concerning warehouse design decisions. In particular, solutions aiming to evaluate the combination of storage policies and order picking policies are widely considered.

Given that decisions are strongly interrelated, for example product location is dependent on warehouse configuration, order picking strategy and technology in place, most of the research in the literature deals with specific practical cases. Thus, a general contribution seems to be scarce.

We believe that we are still missing many references on the application of simulation to warehousing problems. Nonetheless, our intention was not to provide a complete survey about the issue (we would like to do it in the near future), but to call the attention to the potential of the simulation technique in the field of the warehouse management.

CONCLUSIONS

When designing a warehouse several decisions have to be made. For example, the size and layout of the warehouse, the selection of the appropriate storage and handling equipment and the adoption of the best storage, picking and routing policies. Thus, due to the tremendous amount of possible alternatives and the interdependency of several decisions warehouse design and planning became a very complex issue.
In such a complex context a decision support system (DSS) which combine simulation and analytical techniques can be of great help. Optimization methods will help to take decisions within certain assumptions and then simulation technique can be used to validate analytical solutions, evaluate sensitivity of those solutions and also to analyse the system behaviour in presence of uncertainty.

Simulation modelling technique allows the evaluation of operating performance prior to the implementation of a system. It enables:

(i) powerful what-if analysis and;
(ii) permits the comparison of several operational alternatives.

These features are a common background from the survey reported in this paper, which shows how simulation can be successfully adopted in different studies related with warehousing.

We also think that simulation is probably much more widely used in practice than is evident from this literature survey. This is due to the understandable reluctance to publish studies from a confidential nature.

FUTURE WORK

One important challenge in warehousing context is the integration and coordination of several activities in the warehouses. In particular, an important issue is to integrate design and operating decisions minimizing the system global cost.

In this context, a two-stage approach to solve warehousing problems and take advantages of the system dynamics can be taken:

1. Using optimization models to generate solutions taking into account some of the most important cost parameters;
2. Using simulation models to evaluate the solutions generated in the first phase.

![Diagram](image)

Figure 3: A DSS scheme for warehousing decisions

Future work can be carried out on the development of a DSS (Figure 3) that combines analytical and simulation techniques.

REFERENCES


**AUTHOR BIOGRAPHIES**

**Carla A. S. Geraldes** was born in Porto, Portugal. She graduated in Mechanical Engineering at the University of Porto, Portugal and holds an MSc degree in Industrial Engineering – Logistic and Distribution from University of Minho, Portugal. At the moment she is doing her PhD research at the University of Minho, Portugal and is an Assistant Professor at the Department of Industrial Management of the Polytechnic Institute of Bragança, Portugal. Her e-mail address is: carlag@ipb.pt.

**Maria Sameiro Carvalho** was born in Braga, Portugal. She graduated in Computer and Systems Engineering in the University of Minho, Portugal. She holds an MSc degree in Transportation Planning and Engineering and a PhD degree in Transportation Planning from the University of Leeds, UK. She is Associate Professor at the Department of Production and Systems Engineering, of University of Minho, Portugal. She is also a researcher of the Systems Engineering, Optimization and Operations Research Group of the Algoritmi Research Center. Her main research interests are in Operational Research, Transportation and Logistic. Her e-mail address is: sameiro@dps.uminho.pt.

**Guilherme A B Pereira** was born in 1961 in Porto, Portugal. He graduated in Industrial Engineering and Management at the University of Minho, Portugal. He holds an MSc degree in Operational Research and a PhD degree in Manufacturing and Mechanical Engineering from the University of Birmingham, UK. His main research interests are Operational Research and Simulation. His e-mail address is: gui@dps.uminho.pt.
AN INTEGRATED APPROACH FOR WAREHOUSE DESIGN AND PLANNING

Carla A. S. Geraldes  
Algoritmi Research Center  
School of Technology and Management  
Polytechnic Institute of Bragança  
Campus de Santa Apolónia, Apartado 1134  
5301-857 Bragança, Portugal  
E-mail: carlag@ipb.pt

Maria Sameiro Carvalho and Guilherme A B Pereira  
Algoritmi Research Center  
School of Engineering  
University of Minho  
Campus de Guimarães  
4710-057 Braga, Portugal  
E-mail: {sameiro|gui}@dps.uminho.pt

KEYWORDS  
Warehouse design and planning. Warehouse management.  
Inventory management. Logistics. Modelling.

ABSTRACT  
Warehouse design and planning is a great challenge in the field of Supply Chain Management. Inventory level management, warehouse design and operations, and customers’ requirements are examples of important challenges in this context. Throughout this work we discuss a mathematical model aiming to support some warehouse management decisions and inventory decisions. Our aim is to show model’s potentialities and weaknesses when applied to real world problems and to identify challenging research opportunities for developing more global warehouse decision support models to fill the gap between researchers and warehouse practitioners.

INTRODUCTION  
Within a supply chain network, products need to be physically moved from one location to another. During this process, they may be buffered or stored at certain facilities (warehouses) for a certain period of time. In this context, warehouses play an important role in supply chains and are a key aspect in a very demanding, competitive and uncertain market. Although many companies examined the possibilities of direct supply to customers, there are still many circumstances where this is not appropriate. According to Bartholdi and Hackman (2006) there are four main reasons why warehouses are useful:

- To consolidate products in order to reduce transportation costs and to provide customer service;
- To take advantage of economies of scale;
- To provide value-added processing and
- To reduce response time.

Thus, warehouses will continue to be an important node at the logistic network.

In distribution logistics where market competition requires higher performances from warehouses, companies are compelled to continuously improve the design and planning of warehouse operations. Furthermore, the ever-increasing variety of products, the constant changes in customer demands and the adoption of management philosophies also bring new challenges to reach flexible structures that provide quality, efficiency and effectiveness of the logistics operations. In practice, warehouses must be modular, adaptable, compact, accessible and flexible, and must be capable to respond to changing conditions, to improve space utilization and to reduce congestion and movement.

Warehouse design and planning typically runs from a functional description, through a technical specification, to equipment selection and determination of the layout.

![Figure 1: Framework for design and operation problems (adapted from Gu et al., 2007).](image)

Figure 1 illustrates the five major decisions involved in warehouse design according to Gu et al. (2007). The overall structure decision determines the material flow patterns within the warehouse, the specification of functional areas and the flows between areas. Sizing and dimensioning decisions determine the total size of the warehouse as well as the space allocation among functional areas. Layout definition is the detailed configuration within a functional area and equipment decisions define an automation level for the warehouse and identify equipment types. Finally operating policies refer to storage, picking and routing decisions.

Hassan (2002) presented a framework for the design of warehouse layout. The proposed framework accounts for several factors and operations of warehousing such us:

1. Specification of warehouse type and purpose;
2. Analysis and forecasting demand;
3. Definition of operating policies;
4. Establishment of inventory levels;
5. Class formation;
6. Definition of functional areas and general layout;
7. Storage partition;
8. Selection of equipment for handling and storage;
9. Design of aisles;
10. Determination of space requirements;
11. Location and number of I/O points;
12. Location and number of docks;
13. Arrangement of storage;
14. Zone formation.

Once warehouse decisions are strongly interrelated, warehouse design is a highly complex task where frequently conflicting objectives impose specific trade-offs.

Although interrelated, decisions are dealt independently in a pyramidal top-down approach. Strategic decisions create limits to decisions taken at the tactical and operational levels and tactical decisions limits operational decisions. Also decisions taken at each level are handled independently and sequentially (Van den Berg 1999).

The majority of scientific research studies addresses isolate problems. However, most real problems are unfortunately not well-defined and often cannot be reduced to multiple isolated sub-problems. Therefore, warehouse design often requires a mixture of analytical skills and creativity. Anyhow, research aiming an integration of various decisions models and methods is badly needed in order to develop a methodology for systematic warehouse design (Rouwenhorst et al. 2000).

In this paper we discuss a high-level warehouse and inventory model, adapted from Strack and Pochet (2010). The model jointly integrates:

- The size of the functional areas;
- The assignment of products to storage locations in the warehouse;
- The allocation of products to warehouse systems;
- The replenishment decision in the inventory management.

In the next section, we will make a brief review of the literature available on warehouse design and planning. Then, the integrated model for warehouse design developed by Strack and Pochet (2010) will be extended, analysed and some computational results presented. Finally some conclusions and future work are mentioned.

WAREHOUSE MANAGEMENT

Warehousing is concerned with all the material handling activities that take place within the warehouse. They include the receiving of goods, storage, order-picking, accumulation and sorting and shipping. Basically, one can distinguish two types of warehouses: distribution warehouses and production warehouses. According to Van den Berg and Zijm (1999), a distribution warehouse is a warehouse in which products from different suppliers are collected (and sometimes assembled) for delivery to a number of customers. On the other hand, a production warehouse is used for the storage of raw materials, semi-finished products and finished products in a production facility.

**Warehouse activities**

In this section we consider the flow of material in a warehouse. There are many activities that occur at a warehouse. Typically, distribution warehouses receive products - Stock Keeping Units (SKU) - from suppliers, unload products from the transport carrier, store products, receive orders from customers, assemble orders, repack SKUs and ship them to their final destination. Frequently, products arrive packaged on large scale units and are packaged and shipped on small units. For example, SKU may arrive in full pallets but must be shipped in cases.

Figure 2 shows the typical functional areas and flows within warehouses. Next, a short description of the most common areas and product flows is presented.

![Figure 2: Typical warehouse functions and flows (adapted from Tompkins et al., 2003).](image)

At the receiving area products are unloaded and inspected to verify any quantity and quality inconsistency. Afterwards items are transferred to a storage zone or are placed directly to the shipping area (this is called a cross-docking operation). We can distinguish two types of storage areas: reserve storage area and forward or picking area. The reserve area is the place for products to stay until they are required by costumers’ orders. The picking area is a relatively small area typically used to store fast moving products. Most of the flows between areas are the result of replenishment processes. Order picking is one of the most important functions in most warehouses. SKU are retrieved from their storage positions based on customers’ orders and moved to the accumulation and sorting area or directly to the shipment area. The picked units are then grouped by customer order, packaged and stacked on the right unit load and transferred to the shipping area.

**Warehouse design and planning**

Warehouse design can be defined as a structured approach of
decision making at distinct decision levels in an attempt to meet a number of well-defined performance criteria. At each level, multiple decisions are interrelated and therefore it is necessary to cluster relevant problems that are to be solved simultaneously. According to Rouwenhorst et al. (2000) a warehouse design problem is a “coherent cluster of decisions” and they define decisions to be coherent when a sequential optimization does not guarantee a globally optimal solution.

The design of a warehouse is a highly complex problem. It includes a large number of interrelated decisions involving warehouses processes, warehouse resources and warehouses organizations (Heragu, 2005). Rouwenhorst et al. (2000) classify management decisions concerning warehousing into strategic decisions, tactical decisions and operational decisions. Strategic decisions are long term decisions and always mean high investments. The two main issues are concerned with the design of the process flow and with the selection of the types of warehousing systems. Tactical management decisions are medium term decisions based on the outcomes of the strategic decisions. The tactical decisions have a lower impact than the strategic decisions, but still require some investments and should therefore not be reconsidered too often. At the operational level, processes have to be carried out within the constraints set by the strategic and tactical decisions made at the higher levels. At this level, the concern includes the operational policies such as storage policies and picking operations.

After determining warehouse location and its size, layout decisions must include areas definition and what size should be allocated to each functional area. The forward-reserve problem (FRP) is the problem of assigning products to the functional areas. In this problem the critical decision concerns the choice of products that will be stored in the forward area. Van den Berg et al. (1998) proposed a binary programming model to solve de FRP in the case of unit load replenishment and presented efficient heuristics that provide tight performances guaranties. Those replenishments can occur during busy or idle picking periods. The objective was to minimize the number of urgent or concurrent replenishments of the forward area during the busy periods.

Although addressing this problem is a strategic decision problem, it is strongly associated upon some tactical problems such as how the items will be distributed among the functional areas. However, the approach usually adopted is to solve the problems sequentially by generating multiples alternatives for the functional area size problem and then determine how the products can be allocated for each of the alternatives.

Gray et al. (1992) developed an integrated approach to the design and operation of a typical order-consolidation warehouse. This approach included warehouse layout, equipment and technology selection, item location, zoning, picker routing, pick generation list and order batching. Due to the complexity of the overall problem, they developed a multi-stage hierarchical decision approach. The hierarchical approach used a sequence of coordinated mathematical models to evaluate the major economic trade-offs and to reduce the decision space to a few alternatives. They also used simulation technique for validation and fine tuning of the resulting design and operating policies.

Heragu et al. (2005) developed a mathematical model and a heuristic algorithm that jointly determines the functional areas size and the product allocation in a way that minimizes the total material handling and storage costs. The proposed model uses real data readily available to a warehouse manager and considers realistic constraints.

Geraldes et al. (2008) adapted the mixed-integer programming model proposed by Heragu et al. (2005) to tackle the storage allocation and assignment problems during the redesign process of a Portuguese company warehouse.

In addition to warehouse management decisions, an appropriate inventory policy may result in a reduction of the total warehousing costs and can also improve the efficiency of the operating policies within the warehouse. The adoption of new management philosophies compels companies to eliminate or reduce inventory levels. The aim of inventory management is to minimize total operating costs satisfying customer service requirements (Ghani et al. 2004).

To accomplish this, an optimal ordering policy will answer to questions such as when to order and how much to order. There exist two different inventory policies (Hadley and Whitin, 1963): continuous review policy and the periodic review policy. The first policy implies that the stock level will be monitored continuously. A fixed quantity will be ordered when the stock reaches the reorder point. In the second policy, the stock level is checked after a fixed period of time and an ordering decision will be made to complete the stock to an upper limit, if necessary. The operating costs taken into account are the procurement costs, the holding costs and the shortage costs. Those basic policies can be adapted to take into account special situations such as single or multi-item models with or without a constraint on the total storage space, deterministic or stochastic demands, lost sales, etc.

MATHEMATICAL MODEL FORMULATION

The model presented by Strack and Pochet (2010), probably the most robust approach found in this area, integrating multiple aspects as mentioned above, still assumes fixed capacities for both forward and reserve areas. Our approach will then try to give this model the capacity of obtaining optimal sizes for functional areas, i.e., functional areas would be model decision variables.

We consider a warehouse with the following functional areas: receiving, reserve, forward and shipping. Thus, the three following material flow patterns are possible:

- Flow 1: Receiving → Reserve → Shipping
- Flow 2: Receiving → Reserve → Forward → Shipping
- Flow 3: Receiving → Forward → Shipping
The forward area is divided into locations and each product in the forward area is allocated to a number of locations. Before the picking period, the forward area is replenished in advance from the reserve area. Nevertheless, if the stock level in the forward area reaches the reorder point, to avoid shortages, concurrent replenishments can take place during the picking period. The issues that are simultaneously addressed are the decision of the flow pattern taken by the different products in the warehouse and the quantity of the products allocated to the forward and/or the reserve area. In addition, the optimal frequency of the external supplies will be optimized as well as the safety stocks.

Model assumptions

We assume that the total space of available storage space is known. Nevertheless, it is possible to rent external storage space if the space available in the warehouse is insufficient to store all products.

Due to its purpose, the forward area will be handled through a dedicate storage policy. So, if a product is not available in the warehouse the location of this product will be empty and there will be unused space. On the other hand, in the reserve area, a random storage policy is assumed.

During the picking period we will consider the following activities: (i) concurrent and advanced replenishments of the forward area from the reserve area; (ii) picking from the forward and reserve areas; (iii) external supply of the forward and reserve areas.

Finally, concerning the external supply of the products, we will assume an inventory control policy based on continuous review policy (reorder point system).

Model formulation

In formulating the model, the following notation is used.

Parameters:

The indexes used are \( i: 1, \ldots, I \) to denote products and \( j: 1, \ldots, J \) to denote the number of locations in the forward area.

- \( \alpha_i \): Number of units of product \( i \) that can be stored in a single location of the forward area
- \( \text{CapaF} \): Available storage capacity of the warehouse
- \( \text{CostRepA} \): Cost of advanced replenishment
- \( \text{CostRepC} \): Cost of concurrent replenishment
- \( \text{PickF} \): Picking cost in the forward area
- \( \text{PickR} \): Picking cost in the reserve area
- \( \text{CostR} \): Reception cost for the reserve area
- \( \text{CostF} \): Reception cost for the forward area
- \( \text{CostCapS} \): Additional capacity cost
- \( \text{CostCar} \): Inventory carrying cost
- \( \text{CostAcqu}_i \): Acquisition cost of product \( i \)

\( \text{CostShort} \): Shortage cost

\( E(U_i) \): Expected value of the demand of product \( i \)

\( E(p_i) \): Expected value of the number of picks of product \( i \)

\( d_i \): Demand of product \( i \) during supply lead time

\( \mu_i \): Average demand of product \( i \) during supply lead time

\( \sigma_i^2 \): Standard deviation of demand of product \( i \) during supply lead time

\( u_{ij} \): The increase in the expected number of replenishment if we allocate an additional location in the forward area to product \( i \)

Decision variables:

\[
x_{ij} = \begin{cases} 
1 & \text{if product } i \text{ has a Flow 2 pattern with } j \text{ locations allocated in the forward area} \\
0 & \text{otherwise}
\end{cases}
\]

\[
y_i = \begin{cases} 
1 & \text{if product } i \text{ has a Flow 3 pattern} \\
0 & \text{otherwise}
\end{cases}
\]

\[
z_i = \begin{cases} 
1 & \text{if product } i \text{ has a Flow 1 pattern} \\
0 & \text{otherwise}
\end{cases}
\]

\( \text{CapaR} \): Capacity of the reserve area

\( \text{CapaF} \): Capacity of the forward area

\( \text{CapS} \): External storage capacity

\( Q_i \): Replenishment quantity of product \( i \)

\( r_i \): Reorder point of product \( i \)

The general formulation of the model can be stated as:

\[
\begin{align*}
\min & \quad \sum_{i=1}^{I} \text{CostRepA} \times x_{i1} + \sum_{i=1}^{I} \sum_{j=1}^{J} \text{CostRepC} \times x_{ij} \times u_{ij} \\
& + \sum_{i=1}^{I} \text{CostR} \times E(U_i) \times \left( x_{i1} + x_{1j} \right) + \sum_{i=1}^{I} \text{CostF} \times y_i \times \frac{E(U_i)}{Q_i} \\
& + \sum_{i=1}^{I} \text{PickF} \times E(p_i) \times \left( x_{i1} + y_i \right) + \sum_{i=1}^{I} \text{PickR} \times E(p_i) \times z_i \\
& + \text{CostCapS} \times \text{CapS} \\
& + \sum_{i=1}^{I} \text{CostCar} \times \left( \frac{Q_i}{2} + r_i - \mu_i \right) + \sum_{i=1}^{I} \text{CostAcqu}_i \times E(U_i) \\
& + \sum_{i=1}^{I} \text{CostShort} \times \left( \frac{E(U_i)}{Q_i} \right) \int_{r_i}^{\infty} \left( d_i - r_i \right) f \left( d_i \right) \, dd_i
\end{align*}
\]

Subject to:

\[
x_{ij} \leq x_{i,j-1} \quad \forall i, j \geq 2 \tag{2}
\]

\[
\sum_{j=1}^{J} \left( x_{ij} + z_i + y_i \right) = 1 \quad \forall i \tag{3}
\]

\[
\sum_{i=1}^{I} \left( \sum_{j=1}^{J} x_{ij} \right) + \left( \frac{Q_i + r_i - \mu_i}{\alpha_i} \right) y_i \leq \text{CapaF} \tag{4}
\]
\[
\sum_{i=1}^{j} \left( \frac{Q_i}{2} + r_i - \mu_i \right) (x_i + x_{ij}) - \sum_{j=1}^{i} \alpha_i x_{ij} \right] \leq \text{CapaR} + \text{CapS} \tag{5}
\]

\[\text{CapaF} + \text{CapaR} \leq \text{CapaT} \tag{6}\]

\[\text{LL}_R \leq \text{CapaR} \leq \text{UL}_R \tag{7}\]

\[\text{LL}_F \leq \text{CapaF} \leq \text{UL}_F \tag{8}\]

\[\text{CapaF}, \text{CapaR}, \text{CapS}, Q_i, r_i \geq 0 \tag{9}\]

\[x_{ij}, y_{ij}, z_i \in \{0, 1\} \forall i,j \tag{10}\]

The objective function (1) is the expected warehouse and inventory costs per picking period. Concerning the warehouse costs, we have taken into account the: (i) costs of advance and concurrent replenishments of the forward area; (ii) reception costs; (iii) picking costs of the forward and reserve areas; (iv) rental cost of the additional storage capacity. The traditional inventory costs are composed by the carrying costs, the acquisition costs and by the shortage costs.

The model’s integrity is observed by ensuring sequencing constraints (2), specifying that a jth location can be assigned to a product i only if j – 1 locations have already been assigned. In addition, each product can only follow one flow pattern in the warehouse (3). Constraints (4)-(5) ensure that the space constraints for the forward and reserve areas are met, and constraint (6) guarantees that the total available space in the warehouse is not exceeded. Constraints (7)-(8) serve to enforce upper and lower limits on the space that can be allocated to forward and reserve areas. Finally, a set of variables must be nonnegative (9) and another is considered binary (10).

Comparatively to the original model this formulation adds two new decision variables, since the size of the reserve are forward areas are now unknown, and three new constraints (6)-(7)-(8).

### Mathematical model analysis and methodology

The above programming model integrates inventory and management decisions. The reception costs involve the warehouse (i.e. \(y_{ij}, z_i, x_{ij}\)) and the inventory (i.e. \(Q_i\)) variables which highlights the link between warehouse and inventory decisions. However, it makes the objective function (1) and constraints (4)-(5) nonlinear and consequently the model is a mixed-integer nonlinear programming model (MINLP) with a large number of variables and constraints.

To demonstrate the computational complexity involved, the model was solved using LINGO12.0 commercial solver on an Intel Core 2Duo 1.4 GHz CPU and 3GB RAM. For a randomly generated instance with 5 products (SKU) and a warehouse with a total available storage space of 5 locations we have to solve a problem with a total of 48 decision variables of which 35 are integer, and with 29 constraints (3 of them nonlinear). It is a very small size instance for which we only were capable to find local optimums within three hours of CPU time.

Given the complexity of solving this model to optimality, such as Strack and Pochet (2010), we suggest to solve the model sequentially. For this propose we decompose our model in two sub-models: (i) an inventory management sub-model and (ii) a warehouse management sub-model. These two sub-models are solved sequentially: first we solve the inventory sub-model and then the optimal values of the inventory variables are fixed and used to solve the warehouse sub-model.

To obtain the inventory management sub-model we have to eliminate the costs and constraints related to the warehouse management problem. We also condensed the model assuming a customer service level for each product and an approximation of the objective function assuming that the reception costs are independent of the flow pattern taken by the product. Moreover, the capacity constraints for the reserve and forward areas are relaxed. By this, we mean that we will consider only one capacity constraint for the entire storage area.

The resulting inventory management sub-model is a nonlinear programming model defined as:

\[
\begin{align*}
\min \sum_{i=1}^{j} \text{CostCar} \times \left( \frac{Q_i}{2} + r_i - \mu_i \right) + \sum_{i=1}^{j} c_{\text{Capq}} \times E(U_i) \\
+ \sum_{i=1}^{j} \text{CostShort} \times \left( \frac{E(U_i)}{Q_i} \right) \times \int_{r_i}^{\infty} (d_i - r_i) f(d_i) \, dd_i \\
+ \sum_{i=1}^{j} \text{CostRecp} \times \left( \frac{E(U_i)}{Q_i} \right) + \text{CostCapS} \times \text{CapaS}
\end{align*}
\tag{11}\]

Subject to:

\[\sum_{i=1}^{j} \left( Q_i + r_i - \mu_i \right) \leq \text{CapaT} + \text{CapS} \tag{12}\]

\[\text{CapS}, Q_i \geq 0 \tag{13}\]

The considered objective function (11) is independent of the products flow patterns and consequently independent of warehouse decisions. Also, the capacity constraint (12) is independent of the flow patterns taken by the different products. Nevertheless, the inventory variables will be dependent of the available storage capacity, and in a certain way we can consider that this sub-model integrates an inventory decision and a warehouse decision.

Warehouse management sub-model is obtained fixing the inventory variable \(Q_i\), based on the solution of the inventory sub-model.

The resulting warehouse sub-model is a mixed-integer problem that can be stated as:
\[
\begin{align*}
\min & \sum_{i=1}^{l} \text{CostRepA} \times x_{i1} + \sum_{i=1}^{l} \sum_{j=1}^{l} \text{CostRepC} \times x_{ij} \times u_{ij} \\
& + \sum_{i=1}^{l} \text{CostR} \times \frac{E(U_i)}{Q_i} \times (z_i + x_{i1}) + \sum_{i=1}^{l} \text{CostF} \times y_i + \frac{E(U_i)}{Q_i} \\
& + \sum_{i=1}^{l} \text{PickF} \times E(p_i) \times (x_{i1} + y_i) + \sum_{i=1}^{l} \text{PickR} \times E(p_i) \times z_i \\
& + \text{CostCapS} \times \text{CapS}
\end{align*}
\] (14)

Subject to:

\[
x_{ij} \leq x_{i(j-1)} \quad \forall i, j \geq 2
\] (15)

\[
\sum_{i=1}^{l} (x_{ij} + z_i + y_i) = 1 \quad \forall i
\] (16)

\[
\sum_{i=1}^{l} (\sum_{j=1}^{l} x_{ij}) + (\frac{Q_i + n_i - \mu_i^2}{a_i}) y_i \leq \text{CapaF}
\] (17)

\[
\sum_{i=1}^{l} \left( \frac{Q_i}{2} + r_i - \frac{\mu_i^2}{a_i} \right) (z_i + x_{i1}) - \sum_{j=1}^{l} a_i x_{ij} \leq \text{CapaR} + \text{CapaS}
\] (18)

\[
\text{CapaF} + \text{CapaR} \leq \text{CapaT}
\] (19)

\[
\text{LL}_R \leq \text{CapaR} \leq \text{UL}_R
\] (20)

\[
\text{LL}_F \leq \text{CapaF} \leq \text{UL}_F
\] (21)

\[
\text{CapaF}, \text{CapaR}, \text{CapaS} \geq 0
\] (22)

\[
x_{ij}, y_i, z_i \in \{0, 1\} \quad \forall i, j
\] (23)

Now, in the warehouse management sub-model the flow pattern variables and the size of the functional areas (reserve and forward) will be optimized taking into account the global storage capacity and the upper and lower limits imposed to the functional areas. The optimal value of the additional capacity is re-optimized.

Computational results

To evaluate the computational performance involved on solving the proposed sub-models test problems were solved using LINGO 12.0 commercial solver on an Intel Core 2Duo 1.4 GHz CPU and 3GB RAM. Instances for different scenarios (Table 1) were randomly generated. Table 2 shows parameter values used to generate the testing problems.

Table 1: Analyzed scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>SKU [units]</th>
<th>Total Storage Capacity [No. of Locations]</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>II</td>
<td>100</td>
<td>70</td>
</tr>
<tr>
<td>III</td>
<td>500</td>
<td>300</td>
</tr>
<tr>
<td>IV</td>
<td>1000</td>
<td>600</td>
</tr>
<tr>
<td>V</td>
<td>5000</td>
<td>3000</td>
</tr>
</tbody>
</table>

Table 2: Parameter values for the numerical examples

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CostRepA</td>
<td>10</td>
</tr>
<tr>
<td>CostRepC</td>
<td>25</td>
</tr>
<tr>
<td>PickF</td>
<td>3</td>
</tr>
<tr>
<td>PickR</td>
<td>10</td>
</tr>
<tr>
<td>CostR</td>
<td>3</td>
</tr>
<tr>
<td>CostF</td>
<td>5</td>
</tr>
<tr>
<td>CostCapS</td>
<td>50</td>
</tr>
<tr>
<td>CostCar</td>
<td>3</td>
</tr>
<tr>
<td>CostShort</td>
<td>100</td>
</tr>
<tr>
<td>$U_i$</td>
<td>Uniform $[1, 50]$</td>
</tr>
<tr>
<td>$p_i$</td>
<td>Uniform $[1, 5]$</td>
</tr>
<tr>
<td>$d_i$</td>
<td>$N(\mu_i, \sigma_i)$</td>
</tr>
</tbody>
</table>

The computational results obtained for the sub-models are shown in Tables 3 and 4.

Table 3: Inventory sub-model computational results

<table>
<thead>
<tr>
<th>Scenario</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total variables*</td>
<td>11</td>
<td>101</td>
<td>501</td>
<td>1001</td>
</tr>
<tr>
<td></td>
<td>Iterations</td>
<td>167</td>
<td>593</td>
<td>1887</td>
<td>17418</td>
</tr>
<tr>
<td></td>
<td>CPU time [mm:ss]</td>
<td>00:01</td>
<td>00:16</td>
<td>00:48</td>
<td>03:55</td>
</tr>
</tbody>
</table>

* Variables involved in the model’s nonlinear relationships.

Table 4: Warehouse sub-model computational results

<table>
<thead>
<tr>
<th>Scenario</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>IV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total variables</td>
<td>43</td>
<td>2203</td>
<td>38503</td>
<td>152003</td>
</tr>
<tr>
<td></td>
<td>Binary variables</td>
<td>40</td>
<td>2200</td>
<td>38500</td>
<td>152000</td>
</tr>
<tr>
<td></td>
<td>No. of constraints</td>
<td>24</td>
<td>2004</td>
<td>37504</td>
<td>150004</td>
</tr>
<tr>
<td></td>
<td>Iterations</td>
<td>0</td>
<td>0</td>
<td>849</td>
<td>2798</td>
</tr>
<tr>
<td></td>
<td>CPU time [mm:ss]</td>
<td>00:01</td>
<td>00:01</td>
<td>00:17</td>
<td>02:10</td>
</tr>
</tbody>
</table>

** Due to the size of the generator matrix, the computer did not have sufficient memory.

As it can be seen it was possible to analytically solve both sub-models to optimality for the first four scenarios in a very satisfactory computational time. Nevertheless, for large instances (Scenario V), the number of binary variables and constraints of the warehouse management sub-model increases considerably. Consequently solving the model using a branch-
and-bound based algorithm for mixed-integer programming problems takes significant computational time and memory.

CONCLUSIONS AND FUTURE WORK

In an uncoordinated manner inventory decisions are taken without any warehouse considerations and vice-verse. In this work our aim was to show the value of integrating some decisions that occur during the design and planning of a warehouse.

For that purpose a mathematical model for inventory and warehouse management was extended and analysed. The proposed mixed-integer nonlinear programming model jointly integrates: (i) the size of the functional areas; (ii) the assignment decision of products to storage locations; (iii) the allocation decision and; (iv) the replenishment decision.

Nevertheless, and due to the complexity of the analytical model obtained, an optimal global solution is definitely difficult to achieve. For this reason, the model was decomposed in two sub-problems which were solved using a hierarchical sequential approach.

First a nonlinear inventory model was solved taking into account only the total capacity of the available storage. The second sub-model refers to warehouse decision. Model was obtained fixing the inventory variables bases on the solution of the first sub-problem. This sub-model allows us to jointly determine the flow pattern for each product, the dimensions of the functional areas and the eventual need to rent some external storage area.

To have a single decision model that integrates several decisions concerning warehouse design and planning is a very complex problem due: to the large amount of information to be processed; to the tremendous amount of existing alternatives; to the existence of various and often conflicting objectives and to the uncertainty inherent in the material flow into, through and out of the warehouse.

As Ashayeri and Gelders (1985) we also believe that an analytical approach can be used to solve simplified models and simulation technique can be used to validate the models and to incorporate dynamic aspects not yet included in the model. For example, we could use the gathered solution of the considered sub-problems and then simulation can be used to introduce demand fluctuations or operational decisions related with storage or picking policies. Therefore, with the application of this dual technique, we believe that the major features of both techniques can be enjoyed in the warehouse design and planning problem.

These considerations allow us to say that there exist many challenging opportunities for developing more global warehouse decision support models.

REFERENCES


HUMANITARIAN LOGISTICS DEPOT LOCATION MODEL

Christof Kandel
Hella Abidi
Matthias Klumpp
Institute for Logistics and Service
Management (ild)
FOM University of Applied Sciences
Leinkugelstraße 6
D-45141 Essen, Germany
E-Mail: christof.kandel@fom-ild.de

KEYWORDS
Humanitarian Logistics, Depot Location Model, Network Storage, Response Depot Simulation

ABSTRACT
Lack of coordination and collaboration among participants of humanitarian relief chains, high costs in the area of Humanitarian Logistics and long transit times of goods build this research paper on the idea that the response depot location should be improved to reach the required delivery time of 24/48 hours by producing lower transport and logistics cost in case of a disaster in the world. With this in mind this research article describes the humanitarian relief chain, the depot locations of two global players such as International Federation of the Red Cross and the UN World Food Program in a disaster relief. Furthermore the typical location problem in logistics will be solved especially for one single global depot location in Humanitarian Logistics by developing and using a mathematical location planning model. Finally to outline the research strategic locations of depots for more effective humanitarian supply chains will be compared. The focus is to develop a mathematical model based on the fact that there are not many mathematical solution programs for strategic and tactical decisions in the area of Humanitarian Logistics. According to a literature review in the period of 2005 till 2010 in scientific and academic publications in journals and conferences proceedings only 7 publications focused on research methodology such as mathematic models in Humanitarian Logistics.

As an introduction the instantaneous situation of the humanitarian operation relief chain is shown. In the second chapter the problem description is given before a location model is introduced. In the fourth chapter a solution is calculated and compared with existing response depot locations in the fifth chapter. It ends with a conclusion and an outlook. The aim of the paper at hand is to discuss the strategically and tactical decision of locations of response depots in the area of Humanitarian Logistics in case of an efficient and effective structuring of humanitarian relief operations.

INTRODUCTION
The topic Humanitarian Logistics recently becomes very important: The number of disasters has increased significantly in the last 30 years and multiplied six times between 1974 and 2010. In 2010, 304 disasters occurred. 167 of them were natural disasters which were released by Epidemic, Earthquake, Volcano, Flood, Rockfall, Landslide, Storm, Heat Wave, Cold Wave, Drought and Wildfire. 137 were man-made disasters such as war, political crisis, chemical accident or car accident. The total economic damage was 218 billion USD (Bevere et al. 2011). “In 2010, the number of reported disasters approximated the annual average disaster occurrence during 2000 to 2009 (387)” (Guha-Sapir et al. 2010). Also the number of victims increased. In 2009, 198.7 million people were affected or died which caused an economic damage of 68 billion USD. Then in 2010, 217.3 million victims were caused by natural disasters, which is less than the annual average of 2000-2009 of 227.5 million victims but the economic damage caused by natural disasters was more than 25 % higher than the annual average of 98.9 billion USD (Guha-Sapir et al. 2010). The earthquake in Haiti and the flood in Pakistan had the highest impacts and wreak a huge economic damage in 2010. The year 2010 is the seventh strongest year with disasters, impacts and affected people in history since 1970 (Bevere et al. 2011). In summarizing, it can be stated that 2010 was a year of extreme weather occasions such as flood and earthquake and in Asia the highest economic damage occurred. According to Fritz Institute, a high increase of impacts and occurrence of technical and natural disasters is expected in the future (Thomas, A., Kopczak, L.R. 2005). This shows how significant and essential an effective and efficient Humanitarian Logistics and location planning is. In order to outline the importance of Humanitarian Logistics a literature review was conducted: Kovacz and Spens determined 2007 that there were not enough research or publications in academic journals about Humanitarian Logistics; furthermore the available research publications did not indicate quantitative methodologies. The literature review in the period time of 2005 till 2010 in scientific and academic publications in journals and conferences proceeding which were sourced from the science and management journal databases such as Emerald, Business Source Premier and Science Direct only focused on 7 publications with research methodology such as mathematic models. The majority academic research on this topic was published in 2005 after the Indian Ocean Tsunami in 2004 (Kovács and Tatham 2009). Following scientific and academic papers were published dealing with the topic of Humanitarian Logistics: 2004 Beamon about the comparison of humanitarian relief chain and commercial supply chains, 2005 Pettit and Beresford; 2007 Tovia; 2009 Maon et al. presented the Humanitarian Logistics model, 2006 Beamon
and Kotleba describe the inventory management in humanitarian relief, 2008 Beamon and Balci showed an example for performance measurement, 2009 Kovác and Spens; 2010 Chandes and Pach gives an overview about the challenges in Humanitarian Logistics, 2008 Balci et al. issued a model of last mile distribution also Vitoriano et al. gave an idea of an optimization model for humanitarian aid distribution in particular for Haiti, 2010 Balci et al. and 2010 Taham and Kovác treat the problematic and current case besides the topic of optimization of the humanitarian response depot the coordination in the humanitarian field. The most similar paper to this research paper is the publication of Balci and Beamon in 2008 about facility location planning. The paper of Balci and Beamon mainly implied a mathematical model for establishing a distribution network in the whole world. In their publication they develop a model that defines the number and location of distribution center in a relief network and the amount of items which can be stocked at each distribution center to supply affected people in case of disaster in a short time. The aim of this research paper is to define one main depot which can assure an effective and an efficient supply by respecting the balance between the lower transport costs, lower transit time, lowest distance to every country in the world in case of disasters. Also this research paper illustrates the idea of establishment of one central depot and it can be seen as new to most NGOs and GOs as existing research literature in this area.

**PROBLEM DESCRIPTION**

First of all the definition of Humanitarian Logistics has to be presented to develop a better understanding of humanitarian operation relief and a disaster effect: A disaster occurs damage of property, human suffering and represents a disruption of society. Disaster is not only an unexpected catastrophe but the slow buildup of environmental, political, or economic factors impact on a vulnerability of a society (Plapp 2004). Hoyois et al. denote a disaster as “a situation or event, which overwhelms local capacity, necessitating a request to national or international level for external assistance; an unforeseen and often sudden event that causes great damage, destruction and human suffering” (Hoyois et al. 2007). Furthermore a disaster can be seen as “an emergency of such severity and magnitude that the resultant combination of deaths, injuries, illness and property damage cannot be effectively managed with routine procedures or resources. These events can be caused by nature, equipment malfunction, human error or biological hazards and diseases” (Landesmann 2005). CRED classifies the various disaster types as followed:

<table>
<thead>
<tr>
<th>Biological</th>
<th>Geophysical</th>
<th>Hydrological</th>
<th>Meteorological</th>
<th>Climatological</th>
</tr>
</thead>
<tbody>
<tr>
<td>-Epidemic:</td>
<td>Vind, Bacterial, Parastic, Fungal, Poison Infections</td>
<td>-Earthquake</td>
<td>-Volcano</td>
<td>-Mass Movement</td>
</tr>
<tr>
<td>-Vind</td>
<td>-Rockfall, Landslide, Avalanche, Subsidence</td>
<td>-General Flood</td>
<td>-Flash Flood</td>
<td>-Surge Coastal Flood</td>
</tr>
<tr>
<td>-Bacterial</td>
<td></td>
<td>-Storm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-Parastic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-Fungal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-Poison Infections</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disease</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-Insect Infestation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-Animal Stampede</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This argues for the range activities of Humanitarian Logistics and their efficient organization. Then Humanitarian Logistics “required to procure, store and distribute supplies for the assistance of beneficiaries” in case of a disaster (Howden 2009). Logistics in the humanitarian sense strives to prevent and minimize starvation, disease and death as an effect of natural, technological or man-made disasters by effectively providing basic needs such as food, water, shelter and medical care to all victims (Blecken 2010). Humanitarian Logistics is defined as the process of planning, implementing and controlling the efficient, cost-effective flow and storage of goods and materials, as well as related information, from the point of origin to the point of consumption for the purpose of alleviating the suffering of vulnerable people” (Thomas and Kopczak 2005).

The stakeholders of Humanitarian Logistics are beneficiaries, donors, the media, host government, disaster response operations, regional organizations an in-country operations (Thomas 2003). An efficient Humanitarian Logistics commits a coordination between the stakeholders, optimization of response depots, distribution network, performance measurement and short delivery times by including following functions such as (Howden 2009):

- warehousing (storage and handling)
- transportation (planes, cars, trucks, rail, ship)
- security, fleet management
- procurement (material, food, medical, water, sanitation, sources of energy)
- asset management, build management
- communication and information technology (radio, internet, phones)

in an environment with limited or no existing of infrastructure and communication (Rickard 2006). This complex logistic structure although resembles to commercial logistic but they are many weaknesses in (Fuessel et al. 2006):

- coordination among the stakeholder
- procurement process which is fragmented
- non-transparent contracts and conditions
- out of stock situation and commodity shortages
- infrequent orders
- overstretched storage and delivery routes

The above mentioned Humanitarian Logistics challenges argue for the main emphasis addressed in this paper. Humanitarian Logistics performance can be increased by the optimum location of humanitarian response depot in the whole world, then the efficient handling of fleet management and minimize of transport distance mean minimize of transport costs. All these aspects maximize the output such as food, medical, shelter, water sanitation deliveries and all the needs of beneficiaries in a short transit time and the out of stock situation will be decreased. To create a mathematical model for the optimum of the location of humanitarian response depot following assumptions for simplification has to be made in order to achieve a useful result (Table 2):

1) The needed data was adapted from CRED database: 11 large disaster events over the period 2000-2010 were used as input-data for the mathematical model. CRED determined following criteria in their database.
for a designation of a disaster and one of them have to be fulfilled:
- At least 10 persons were reported as death
- At least 100 persons were injured and need immediately help
- International help is needed
- Declaration the disaster as an emergency

II) The geo-coordinates of each main city affected were set as coordinates for disasters.

III) In reality the frame of the globe is an ellipse. The authors decide for a simplification of the model to assume the globe as a ball.

IV) The authors define the first urgent supplies item which will be needed in the first 72 hours in such disaster event.

V) Each weight for the first urgent supplies were found out from the items catalogue of IFRC and calculated as transport weight for each person affected (Tab.3). After these steps following tables were finally used as input data to create a structural model and to define the necessary attribute.

<table>
<thead>
<tr>
<th>Kit</th>
<th>Weight [kg]</th>
<th>Quantity of people for each kit</th>
<th>Weight/ Person [kg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domestic</td>
<td>24</td>
<td>5</td>
<td>4.800</td>
</tr>
<tr>
<td>Sanitation</td>
<td>329</td>
<td>5</td>
<td>65.800</td>
</tr>
<tr>
<td>Health</td>
<td>998</td>
<td>30000</td>
<td>0.033</td>
</tr>
<tr>
<td>Tent</td>
<td>55</td>
<td>5</td>
<td>11.000</td>
</tr>
<tr>
<td>Shelter</td>
<td>20</td>
<td>5</td>
<td>4.000</td>
</tr>
<tr>
<td>Water</td>
<td>7000</td>
<td>10000</td>
<td>0.700</td>
</tr>
<tr>
<td>sum [kg]</td>
<td></td>
<td></td>
<td>86.333</td>
</tr>
<tr>
<td>sum [tons]</td>
<td></td>
<td></td>
<td>0.0863</td>
</tr>
</tbody>
</table>

Table 2: Database for a location model and calculation

<table>
<thead>
<tr>
<th>i</th>
<th>popular name</th>
<th>main countries affected</th>
<th>date</th>
<th>type of hazard</th>
<th>main city affected</th>
<th>latitude $\varphi$</th>
<th>longitude $\lambda$</th>
<th>people affected</th>
<th>transport weight [t]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Haiti Earthquake</td>
<td>Haiti</td>
<td>2010-01-12</td>
<td>Earthquake</td>
<td>Port-au-Prince</td>
<td>18</td>
<td>32</td>
<td>18.53</td>
<td>-72</td>
</tr>
<tr>
<td>2</td>
<td>Sichuan Earthquake</td>
<td>China</td>
<td>2008-05-12</td>
<td>Earthquake</td>
<td>Chengdu</td>
<td>30</td>
<td>40</td>
<td>30.67</td>
<td>104</td>
</tr>
<tr>
<td>3</td>
<td>Cyclon Narjis</td>
<td>Myanmar</td>
<td>2008-05-02</td>
<td>Tropical cyclone</td>
<td>Yangon</td>
<td>16</td>
<td>47</td>
<td>16.78</td>
<td>96</td>
</tr>
<tr>
<td>4</td>
<td>Java Earthquake</td>
<td>Indonesia</td>
<td>2006-05-27</td>
<td>Earthquake</td>
<td>Yogyakarta</td>
<td>-7</td>
<td>-48</td>
<td>-7.80</td>
<td>110</td>
</tr>
<tr>
<td>5</td>
<td>Kaddimii Earthquake</td>
<td>Pakistan</td>
<td>2005-10-08</td>
<td>Earthquake</td>
<td>Muzaffarabad</td>
<td>31</td>
<td>31</td>
<td>31.35</td>
<td>73</td>
</tr>
<tr>
<td>6</td>
<td>Hurricane Katrina</td>
<td>United States</td>
<td>2005-08-29</td>
<td>Tropical cyclone</td>
<td>New Orleans</td>
<td>29</td>
<td>57</td>
<td>29.95</td>
<td>-90</td>
</tr>
<tr>
<td>7</td>
<td>Mumbai Floods</td>
<td>India</td>
<td>2005-07-26</td>
<td>Flood</td>
<td>Mumbai</td>
<td>18</td>
<td>58</td>
<td>18.97</td>
<td>72</td>
</tr>
<tr>
<td>8</td>
<td>South Asian Tsunami</td>
<td>Indonesia etc.</td>
<td>2005-12-24</td>
<td>Tsunami</td>
<td>Banda Aceh</td>
<td>5</td>
<td>33</td>
<td>5.55</td>
<td>95</td>
</tr>
<tr>
<td>9</td>
<td>Bam Earthquake</td>
<td>Iran</td>
<td>2005-12-23</td>
<td>Earthquake</td>
<td>Bam</td>
<td>29</td>
<td>6</td>
<td>29.10</td>
<td>58</td>
</tr>
<tr>
<td>10</td>
<td>Dresden Floods</td>
<td>Germany</td>
<td>2005-08-11</td>
<td>Flood</td>
<td>Dresden</td>
<td>51</td>
<td>5</td>
<td>51.05</td>
<td>13</td>
</tr>
<tr>
<td>11</td>
<td>Gujarat Earthquake</td>
<td>India</td>
<td>2001-01-26</td>
<td>Earthquake</td>
<td>Bhuj</td>
<td>23</td>
<td>15</td>
<td>23.25</td>
<td>69</td>
</tr>
</tbody>
</table>

LOCATION MODEL

To find the facility location of one quick response depot for Humanitarian Logistics the weber facility location methodology can be used. This standard method of operations research is used to find a facility location ‘in the open countryside’. The aim of this method is to estimate an optimal facility location in plane for a set of customers by using Euclidean distance measurement (Domschke and Drex1 1996).

In this research article a facility location is searched on a surface of a ball (the globe) for a set of disasters with given coordinates on the surface of this ball. Therefore it is important how long a distance on this surface between two of these coordinates is (figure 1).

Equation (1) shows the relation between the apex angle $W$ in radian, the radius of the ball and the geo-coordinates of the points $(\varphi, \lambda, \rho, \lambda_0)$ defining $W$:

$$W = r \cos^{-1}(\cos \varphi \cos \varphi_0 \cos(\lambda - \lambda_0) + \sin \varphi \sin \varphi_0)$$

(1)

This distance measurement methodology has to be integrated in Weber’s facility location problem instead of the Euclidean distance measurement $d_1$.

Therefore the following parameters have to be set:
- average earth radius is $r = 6,371$ km.
- set of customers as disasters with geo-coordinates $(\rho, \lambda)$ in radian.
- demand $b_i$ of each customer as weight of first urgent supplies per capita times number of people affected.

So the target function of the model is the following with the coordinates $(\rho, \lambda)$ as target coordinates of the centralized quick response depot.

$$\min \sum a_i b_i \cos^{-1}(\cos \varphi \cos \varphi_0 \cos(\lambda - \lambda_0) + \sin \varphi \sin \varphi_0) \cdot 6371$$

(2)

This function minimizes the total transportation performance in tons kilometers for all disasters depending on the localization of the quick response depot. Since the partial derivatives of equation (2) set equal to zero cannot be solved neither for $\rho$ nor for $\lambda$, an analytical optimum cannot be estimated. Because for solving the Euclidean single facility.

Figure 1: Distance calculation methodology
location problem the same problem occurs, Miehle developed an iterative method to estimate the optimal location. By using this method only few iteration steps suffice for receiving a satisfying accuracy (Miehle 1958). This iterative method for the location problem in plane has to be transferred to this facility location problem on the surface of a ball as follows:

First the partial derivatives of (2) with respect to ρ and λ have to be estimated:

\[
dF \over dp := \sum_{i=1}^{n} \frac{6371 b_i (\sin \varphi_i \cos \varphi - \cos \varphi \sin \varphi \cos (\lambda - \lambda_i))}{\sqrt{1 - (\cos \varphi_i \cos \varphi \cos (\lambda - \lambda_i) + \sin \varphi \sin \varphi_i)^2}} = 0 \quad (3)
\]

\[
dF \over d\lambda := \sum_{i=1}^{n} \frac{6371 b_i \cos \varphi \cos \varphi_i \sin (\lambda - \lambda_i)}{\sqrt{1 - (\cos \varphi_i \cos \varphi \cos (\lambda - \lambda_i) + \sin \varphi \sin \varphi_i)^2}} = 0 \quad (4)
\]

Then (3) and (4) have to be solved for ρ and for λ as far as possible and with the received equations (5) and (6) the ‘Miehle’-algorithm can be executed. It starts with an arbitrary chosen set of coordinates, insert as ρ₀ and λ₀ and the results ρ₁ and λ₁ has to be set as ρ₀ and λ₀ in the next iteration step as long as the difference ε between ρ₀ and ρ₁ as well as λ₀ and λ₁ is smaller than a pre-defined boundary.

\[
\varphi^1 = \cos^{-1} \frac{\sum_{i=1}^{n} (-6371 b_i \cos \varphi_i \sin \varphi \cos (\lambda^0 - \lambda_i))}{\sum_{i=1}^{n} (-6371 b_i \sin \varphi_i \cos \varphi \sin (\lambda^0 - \lambda_i) + \sin \varphi_i \sin \varphi_i)^2} \quad (5)
\]

\[
\lambda^1 = \cos^{-1} \frac{\sum_{i=1}^{n} (-6371 b_i \cos \varphi_i \sin \varphi \cos (\lambda^0 - \lambda_i))}{\sum_{i=1}^{n} (-6371 b_i \cos \varphi_i \sin \varphi_i \cos (\lambda^0 - \lambda_i) + \sin \varphi_i \sin \varphi_i)^2} \quad (6)
\]

**RESPONSE DEPOT COMPUTATION**

With the developed iterative method coordinates of the facility location of one quick response depot can be estimated and the total function value can be calculated by using assumptions and input data in tables 3 and 4. The calculation was done with the help of a ‘by-foot’ calculation within a spreadsheet. The iteration boundary was set to ε < 0.1 and the start coordinates (-1.28333°; 36.81667°) are in Nairobi, Kenya. The boundary ε was reached after the 16th iteration. The estimated coordinates for a centralized quick response depot are (31.0265°; 103.7466°). Comparing this result with the input data, it can be seen that the optimal location is within Chengdu, the main city affected of the Sichuan earthquake in 2008. The reason for this is comprehensible: This earthquake affected most people by far in comparison with other disaster events so the transport performance of this disaster has so much impact on the target function that the optimal location of the depot is near to the location of this event.

Thus a decision of another computation was made to estimate the optimal facility location again, but less the data of the Sichuan Earthquake. With the remaining ten large disaster events the optimal location within the favored accuracy was reached after the 11th iteration. This time the estimated coordinates are (29.6760°; 73.3332°). This time the estimated location is in Pakistan near the boundary to India. Also this result is reproducible because with Kashmir Earthquake in Pakistan and Bam and Gujrat Earthquake in India the three largest disasters according to the transport weight are in the immediate vicinity of the estimated location.

This reasons that the transport weight impacts the target function maybe too strong proportional to the distance for estimating the optimal location of a centralized single quick response depot for future. By reason of this finding it has to be questioned which parameter is more important for a humanitarian relief chain:

a) Operating time to reach the disaster location as soon as possible from the depot. Therefore the distance is the main parameter and this can be computed comparatively reliable because in the world a few hot-spots which are susceptible to natural disasters like e.g. earthquakes according to the movements of the tectonic plates (Thomas and Kopczak 2005).

b) Transport performance to get the relief items with minimal resources to the disaster location. Therefore the transport weight is the crucial parameter. Thereby historical input data is unfavorable because the probability that a disaster occurs a second time in the same dimension is quite low.

Because a) seems to be the more crucial factor for relief supply chains, a third computation was made to estimate the optimal location according to operating time which is equal to the target of minimizing the total distance for all disasters, provided that the transport capacity is unrestricted. This was realized by setting the transport weights for every disaster equal to 1. A satisfying accuracy was reached after the 7th iteration. This time the optimal facility location has the geo-coordinates (50.2006°; 74.6191°) and is in Kazakhstan. By contemplating this position on a landscape the impression received that Kazakhstan could be the balance point of the total land of the world and this would be a time optimum for reaching every corner of the earth’s land. Figure 2 shows a coordinate system on a landscape to illustrate the positions of the large disasters and the computed location of a centralized single quick response depot for Humanitarian Logistics.
COMPARISON OF HUMANITARIAN RESPONSE DEPOT SETUP

To get an impression about the quality of a centralized single quick response depot, the computed solution was compared to the depot setup of the two big relief organizations IFRC and WFP according to overall distances and overall transport performances. The IFRC maintain three locations for storing relief supplies and the WFP five locations. These depots provide logistic service like delivery of humanitarian relief items within 24 hours at latest 48 hours to any location around the world by a disaster event as well as procurement, customs declaration services and storage of commodities. The most stored commodities in this network are medical kits, shelter items, IT equipment and mobile office simultaneous serve this as the specific relief items which needed in the first time in a disaster event. For computing the overall distance and the corresponding transport performance for these two setups, it is assumed that the delivery of relief items is always executed from the depot nearest to the disaster location. This was computed with equation (2). The results are shown in table 4. As the results show that suggestion of a single centralized depot causes the highest transport effort to supply the affected people with relief items. But it has to be considered that the maintenance of stocks causes costs and the more stocks have to be maintained the expensive are the fixed costs of the relief chain. The central idea is that the overall costs of maintenance and transportation are lower if only one centralized quick response depot would be operated than three or five depots.

Table 4: Comparision of humanitarian response depot setups

<table>
<thead>
<tr>
<th>Organization</th>
<th>Number of depot</th>
<th>Overall distance [km]</th>
<th>Overall transport performance [t.km]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>1</td>
<td>58,320</td>
<td>29,611.766.349</td>
</tr>
<tr>
<td>IFRC</td>
<td>3</td>
<td>21,845</td>
<td>19,584,886.662</td>
</tr>
<tr>
<td>WFP</td>
<td>5</td>
<td>17,946</td>
<td>18,690,466.711</td>
</tr>
</tbody>
</table>

Another supposition is that the quick delivery of the relief supplies would be faster if it is organized from one depot because less coordination effort is necessary and this reduction of reaction time allows a more efficient planning of the relief distribution network to secure permanent deliveries to the disaster location over a longer period of time.

CONCLUSIONS AND FUTURE DEVELOPMENT

Humanitarian Logistics is an important sector which request for tools to support actions and decisions in this field. After the extensive literature review the idea of an establishment of one central response depot in the world has not been addressed in literature until this time. In this research paper a mathematical model was constructed to present the location of a central response depot where many first supplies have to be stored as a pre-positioned stock. It can also be seen as a concept of preparedness. The authors suggest that with a central response depot the delivery time of the first supplies needed would be reduced, the distance to every country in the world would be minimized and an effective and an efficient logistic organization could be established towards the affected people in case of disaster. This work proposes a new approach for the future which is based upon preventive action, cost, time, reliability and security. Furthermore the idea of an implementation of one HUB center is also possible and is not yet established in the sense of Humanitarian Logistics. Then the first main problem in Humanitarian Logistics is also the coordination of relief items at the local distribution center in a disaster location. Some stakeholders send their relief items uncoordinated to the disaster location and overload the available place which can be small cause of destroyed areas by natural or man-made disasters. The second main problem in Humanitarian Logistics is that the stakeholders do not cooperate as the ideal meaning of supply chain management; these problems could be reduced and solved by a deployment of one global response depot as a HUB center in Humanitarian Logististics. For example if shipment evaluation, tracking and tracing also an overview about the incoming and outgoing shipments are existing, an increase of the performance measurement is possible by operation of one HUB center. This model is specially created for Humanitarian Logistics and has been used for a data set of the greatest disaster which occurred 2001 to 2010 to illustrate a promising result. Future research could better use forecast data to present an even more convincing result. Possible is also to find a central country on the earth based upon geo-coordinates. Hereby is to verify the infrastructure, the traffic system and the procurement opportunities to cover the preparedness of a disaster response in effective and efficient way. Future research also needs to develop of a hub and spoke system which is suitable to humanitarian relief operations and where all humanitarian organizations could participate in an efficient logistics network and could
therefore deliver the supplies within 48 hours to the most affected people in disaster locations throughout the world.

REFERENCES
Thomas, A., Kopczak, L.R. (2005). From Logistics to supply chain management; The path forward in the humanitarian sector, white paper, Fritz Institute, San Francisco, CA.

 BIOGRAPHIES
CHRISTOF KANDEL studied industrial engineering at University Duisburg-Essen. During his study he worked at VOITH paper in Krefeld in the department of project management and afterwards at SIEMENS Energy in Duisburg in the department “procurement & logistics”. After his studies he began to work at FOM ild – Institute for Logistics and Service Management in December 2010 as research assistant.

HELLA ABIDI began her career at DACHSER in Germany and Switzerland and studied part-time business administration at FOM University of Applied Sciences. After finishing her study she started working at FOM ild – Institute for Logistics and Service Management as research assistant in March 2011.

MATHIAS KLUMPP is professor for business administration, especially logistics at FOM University of Applied Sciences and scientific director of FOM ild – Institute for Logistics and Service Management.
INVENTORY LOGISTICS
WORKLOAD CONTROL METHODS STABILITY

Nuno O. Fernandes\textsuperscript{1,2} & S. Carmo-Silva\textsuperscript{2}
\textsuperscript{1}Polytechnic Institute of Castelo Branco, School of Technology
Av. Empresário, 6000-767, Castelo Branco, Portugal
\textsuperscript{2}University of Minho, CITEP, Dept. of Production and Systems
Campus de Gualtar, 4710-057, Braga, Portugal
E-mail: nogf@ipcb.pt; scarmo@dps.uminho.pt

KEYWORDS
Workload control, stability, simulation.

ABSTRACT

In this paper we compare and discuss workload control (WLC) methods that differ with respect to workload accounting. A main issue addressed in the study is the methods stability, i.e. their sensitivity to changes on operation parameters. The paper seeks to aid manufacturers at improving the basis for setting and readjusting these parameters. Simulation results for a random routing job shop indicates that WLC based on the probabilistic workload accounting offers higher stability than based on the corrected aggregate load, but only to changes in one of the studied parameters - the release period. The primary performance measure studied was the percentage of tardy orders.

INTRODUCTION

Workload Control (WLC) is a Production Planning and Control approach focussed on controlling the flow times of orders through the production system (Stevenson et al. 2009). This is done by means of input and output control decisions, mainly at three hierarchical levels namely order entry, order release and priority dispatching. Order release has been described as an important and the core part of WLC (Missbauer, 2009). It determines the type, amount and point in time of orders to release into the shop floor for achieving a balanced and controlled workload on the shop floor. An order release mechanism is used in combination with a pre-shop pool of orders to be released. This buffers the shop floor against dynamics of order arrival. Setting WLC parameters for efficient and effective shop floor control is not an easy task (Land 2006). There is a need for knowing in advance which parameters have a strong influence on system performance and the effect of parameter setting changes. In this paper, to better understand the sensitivity on parameter changes in a production system, WLC stability is examined under the probabilistic and corrected aggregate load workload accounting approaches, using simulation. Stability is measured by their sensitivity to changes on single operating parameters, such as workload norms, release periods and time limits. The percentage of tardy orders is the primary performance measure used.

SIMULATION STUDY

A simulation study was carried out using Arena® software. During simulation runs, data were collected under system steady-state. The length of each run was for 50,000 time units including a warm-up period of 10,000 time units. The average values of 100 independent replications are presented as results. Figure 1 is screenshot of the simulation model.

![Figure 1: Arena Simulation Model.](image)

Model and test example

The study is based on a job shop model used in simulation studies of the WLC concept, e.g. Oosterman et al. (2000) The assumptions of the simulation model are as follows: The shop consists of six capacity groups each with a single machine; Order arrivals occur according to an exponential distribution with a mean inter-arrival time that results in a machine utilisation rate of 90%; Due dates of orders are set by assigning to the order arriving time an discrete uniform distributed allowance between 35 and 60 time units; Routing lengths are uniformly distributed between 1 and 6 operations, without return visits; Machines capacities are identical and remain constant over time; A machine can perform only one operation at a time on any order; An order can be processed by only one machine at a time; Order pre-emption is not allowed; Processing times are assumed to be stochastic following a 2-Erlang distribution with a mean of 1 time unit per order; Set-up times are assumed to be negligible; transfer times between capacity groups are assumed to be zero.

It is assumed that orders come from a planning system or from customers flow directly into the pre-shop pool. These orders are reviewed periodically and considered for release according to the latest release date $LR_D$. This is determined for each order $j$ by backward scheduling from the order's due date $DD_j$, using the planned lead times $LT_w$ of each capacity.
group \( w \) in the set \( S_j \) of the capacity groups required by the routing of \( j \) (Equation 1). A planned lead time of 5 time units per operation was used in the model:

\[
L RD_j = DD_j - \sum_{w \in S_j} T_{w}
\]  

(1)

Only those orders with \( LR D \) that falls within a specific time limit (\( \theta \)) are candidates for release. A time limit of 60 time units was used in the study. Order releases take place periodically at time intervals of length \( T \) - the release period. An order is released only if, as a consequence, in none of the capacity groups in its routing the workload exceeds its norm \( A_w \), that is:

\[
L_w + F_{jw} \cdot O_{jw} \leq A_w, \text{ for all } w \in S_j
\]

(2)

Where \( L_w \) is the current workload of \( w \) and \( F_{jw} \cdot O_{jw} \) is a fraction of the operation time \( O_{jw} \) of \( j \) at \( w \) that should be added to the current load, dependent on the workload accounting method used. If an order is taken to be released the workload of each capacity group in its routing is updated according to the load contribution \( F_{jw} \cdot O_{jw} \), otherwise it remains in the pre-shop pool for eventual release in the next release time. If an order cannot be released the next order in queue is considered for release. This procedure is repeated, as long as workload norms allow it or all orders in the pool have been considered for release. Therefore, only a subset of orders currently waiting in the pool is released each time order release is activated. In the simulation study order release is activated every 5 time units just as in the model by (Osterman et al, 2000). This means that the release period \( T \) is 5. Each time an operation is finished, the load contribution of the order is removed from the corresponding capacity group.

Since WLC keeps the queues small on the shop floor, the importance of priority dispatching tends to be low. This is the reason why simple dispatching rules such as First Come First Served (FCFS) may be used (Bechte 1994). Because of this and considering that FCFS supports the predictability of flow times this rule is used in this study.

Workload accounting

One of the key functions of WLC mechanisms, with particular relevance to order release, is workload accounting over time. It establishes when and how much of the load of a released order (job) should be allocated to each capacity group, e.g. machines or groups of functionally identical machines, in its processing routing.

Two workload accounting approaches are considered in the simulation study: the probabilistic and the corrected aggregate load. Following the probabilistic approach the contribution \( F_{jw} \) of a released order \( j \) to the workload of a capacity group \( w \) in its routing, depends on the “probability” of the order to reach \( w \) within the next planning period \( P \). Equation 3 is used when \( LPG \), the loading percentage is equal for all \( w \) (Breithaupt et al. 1994).

\[
F_{jw} = \frac{100}{LPG} \left( \frac{k_{jw} - 1}{k_{jw}} \right)
\]

(3)

Where, \( k_{jw} \) refers to the current distance, in terms of the number of operations still to be completed before the arrival of job \( j \) at capacity group \( w \). \( LPG \) is given by (Breithaupt et al. 1994):

\[
LPG = \left( 1 + \frac{LT_{w, j}}{p} \right) \times 100
\]

(4)

Like in (Perona and Portioli, 1998) we distinguish between the planning period and the release or checking period. The planning period is the period of time that is taken into consideration to estimate the “probability” of the order to reach a capacity group, and therefore determining the contribution, dependent on that probability, to the capacity group direct load.

Following the corrected aggregate approach the load contribution of an order at the moment of release to the workload of a capacity group is depreciated according to the position of the group in the routing of the order. Equation 5 is used when \( L_{w} \) are equal for all \( w \) (Oosterman et al. 2000).

\[
F_{jw} = \frac{1}{n_{jw}}
\]

(5)

Where, \( n_{jw} \) refers to the position number of \( w \) in the routing of \( j \). Contrarily to the probabilistic approach \( F_{jw} \) does not change during the progress of the order until it reaches \( w \).

SIMULATION RESULTS

The key performance measure on which we focus is the percentage of tardy orders, referring to orders that are completed after the due date.

In the rest of the paper we use the terminology of Oosterman at al. (2000) by designating the probabilistic and the corrected aggregate approaches as A and B respectively.

Both methods, A and B, require a number of parameters to be specified for order release (Land, 2006 and Perona and Portioli, 1998) namely, workload norms (\( A_w \)), planned lead times (\( LT_{w,j} \)), a release frequency, which is function of release period \( T \), and a time limit (\( \theta \)) that prevents jobs from being released too early.

In order to adequately control shop floor activity, it is important to know the impact of the parameters on the performance measures. Jodlbauer and Huber (2008) denote stability as the sensitivity of an approach to changes on single operating parameters. The higher the stability, the smaller the change in the performance measures.

To evaluate the impact of each operation parameter on performance, they were changed one at a time by 50% in relation to reference values indicated in previous section and to a workload norm that results in the lowest percentage of tardy orders – that is the optimum operating point of the logistic performance curve (Land 2006). The stability of each method in terms of the relative change in the percentage of tardy orders (\( e_x \)) when moving away from the reference value of each parameter, is determined by:

\[
e_x = \frac{t_x - t_x^*}{(t_x + t_x^*)/2}
\]

(6)

where, \( t_x \) and \( t_x^* \) are the values of the percentage of tardy orders when changing a single operating parameter from its reference value \( x \) to \( x^* \). Recorded results at the optimum operating points are shown in Table 1. This table shows that, for example, reducing the release period by 50% in method
A decreases the percentage of tardy orders by 32.9%, while reducing the time limit by identical percentage, increases the percentage of tardy order by 12.3%.

<table>
<thead>
<tr>
<th>Method</th>
<th>Parameter</th>
<th>Percent tardy</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>$T$</td>
<td>-32.9</td>
</tr>
<tr>
<td></td>
<td>$\theta$</td>
<td>12.3</td>
</tr>
<tr>
<td></td>
<td>$LT_c$</td>
<td>12.7</td>
</tr>
<tr>
<td></td>
<td>$\Delta_c$</td>
<td>135.3</td>
</tr>
<tr>
<td>B'</td>
<td>$T$</td>
<td>-55.2</td>
</tr>
<tr>
<td></td>
<td>$\theta$</td>
<td>13.2</td>
</tr>
<tr>
<td></td>
<td>$LT_c$</td>
<td>8.7</td>
</tr>
<tr>
<td></td>
<td>$\Delta_c$</td>
<td>140.3</td>
</tr>
</tbody>
</table>

Table 1. WLC methods stability (% change).

According to results we can conclude that the percentage of tardy orders is primarily influenced by workload norms ($\Delta_c$) and secondarily by the release period ($T$). The later shows a greater influence on approach B' than on A. It is worth pointing out that workload norms may have, in terms of magnitude, an asymmetric impact on performance. In fact, it was observed in simulations that tightening workload norms beyond the optimum operating point had a stronger impact on performance than loosening them. In order to improve our understanding of the impact of the release period on performance of methods A and B', this was tested at additional levels, namely $T \in \{1; 2.5; 5; 7\}$. Figure 2 shows production logistic curves of the influence of the release period $T$ on the performance of methods A and B' for different values of shop flow time, i.e. the time that elapses between order release and order completion.

CONCLUSION

This study evaluates the stability of WLC methods, i.e. their sensitivity to changes on operating parameters, in a job shop with highly variable routings. WLC settings studied only differ with respect to workload accounting approaches used, namely the probabilistic and the corrected aggregated one. Our findings indicate that the corrected aggregate load approach tends to be less stable, i.e. more sensitive than the probabilistic approach to one of the parameters - the length of the release period. It was also observed that the release period influences the relative performance of the two approaches. For high release frequencies, i.e. short release period lengths, the corrected aggregate load approach performs better than the probabilistic one, while under low release frequencies it is the probabilistic approach that performs better.

Findings also indicate that, in addition to the workload norm levels, the release period is an operating parameter with a major influence on system performance, measured by the percentage of tardy orders. Thus when implementing the WLC concept the focus should be on these parameters. Further research should be conducted to examine WLC methods stability in real word manufacturing environments as a contribution to bridge the gap between theory and practice of workload control.

ACKNOWLEDGEMENTS

This work had the financial support of FCT-Fundação para a Ciência e Tecnologia of Portugal under the project PEst-OE/EME/UI0252/2011. We also acknowledge the institutional support given by GÊ Power Controls Portugal and University of Minho.

REFERENCES

SIMULATION AS TOOL TO ESTIMATE THE HOMOGENIZATION EFFICIENCY OF BULK MATERIAL HANDLING

Mark B. Duinkerken
Dingena L. Schott
Jos T. van der Geest
Paul Stoop
Gabriel Lodewijks

Faculty of Mechanical, Maritime and Material Engineering
Department Marine and Transport Technology
Delft University of Technology
Mekelweg 2, 2628 CD, Delft
The Netherlands
e-mail: m.b.duinkerken@tudelft.nl

KEYWORDS
Bulk Terminal, Homogenization, Material Handling

ABSTRACT

In bulk material handling, constant quality of the material plays an important role in different industries, e.g. steel production, power generation in coal fired power plants and cement industry. In general, fluctuations in quality are reduced by mixing the material. A systematic way to achieve a more constant quality for large quantities is using blending piles at the stockyards. However, also in other stages in the supply chain the material is reshuffled as a side effect of handling the material (i.e. stacking and reclaiming on a storage pile, loading and unloading of ships).

The purpose of homogenization is to reduce the quality fluctuation in bulk material. Existing theory on homogenization focuses mainly on blending piles. This paper describes the development of a more generic model that includes the different handling processes of bulk material, like loading and unloading of dry bulk carriers and barges. The model can be used to quantify the homogenization efficiency of handling processes, and thus enables to determine optimal loading and unloading strategies of bulk ships.

Goal of the research is twofold: first the developed model is validated using the results of existing blending pile theory. Then, a number of cases is studied for which the traditional theoretical approach is not sufficient.

INTRODUCTION

Within the dry bulk materials handling, a typical supply chain of coal from mining to end user (e.g. power plant) might contain the following handling stages:

- mining dry bulk material (e.g. coal, iron ore, limestone)
- transport to and storage on a stockyard (stockpile or blending pile)
- loading material into deep sea vessels or other transport modes (train, overland belt conveyor)
- transport to an import bulk terminal
- unloading of vessels
- storage on stockyard (stockpile or blending pile)
- transport to blast furnace of power plant

Typical handling rates are up to 10,000 tons of material per hour.

The quality of a specific bulk material is defined by a material property like caloric value, metallic content etc. However, even if the average quality is satisfactory, large fluctuations of the material properties can occur, f.i. due to the geographical origin of the raw materials. Each handling stage in the supply chain will (deliberate or not) reduce these fluctuations; this is sketched in figure 1.

![Figure 1: Fluctuations in input and output streams of a handling process](image)

For deliberate homogenization, blending piles are commonly used to gain a more constant material quality on a large-scale. This is done by systematic stacking and reclaiming of the material in a pile. The stacking takes place in longitudinal direction, whereas the material is reclaimed from the front of the pile intersecting all the stacked layers at once as shown in figure 2.
Figure 2: Large-scale homogenization by a blending pile  
(left: stacking; right: reclaiming)

On a smaller scale reduction of fluctuations will also take place due to the other handling processes involved in the supply chain, such as loading and unloading of vessels. In literature no work was found on the homogenization performance of bulk materials handling processes other than storage in blending piles and mammoth silos.

Loading of vessels is mostly done in a continuous way, by positioning a belt conveyor head above the hold. The loading strategies might vary from keeping the head on a fixed position and travel from time to time to another position, to continuous movement of the loader within time. Unloading of vessels can be done by continuous unloaders, but the majority of the dry bulk is unloaded using grabs. There the operator is in charge of choosing positions above the hold.

This research focuses on the determination of the homogenization efficiency that can be obtained during all bulk handling processes. This includes not only the use of blending piles, but also the loading and unloading of vessels. To model the handling process and to determine the effect of different loading and unloading strategies and parameters a simulation model is built. The model is validated by means of the analytical solutions developed in the nineteen sixties and seventies (section 2). Section 3 discusses the model assumptions and choices as well as the implementation of the model. Then, model verification and validation are given and in the final sections experimental results and conclusions are presented.

EXISTING LITERATURE ON BLENDING AND HOMOGENIZATION

In the nineteen sixties and seventies several people (Van der Mooren 1967), (Gerstel 1979), (Schofield 1979, 1980) have investigated homogenization and have set up theories (Van der Mooren 1967; Gerstel 1979) to describe the homogenization efficiency of blending piles. In this theory the geometry of a blending pile is represented by multiple layers with identical length, width and height. The reclaiming process is assumed to intersect all the layers of the pile, which might be the case for some types of reclaimers (bridge type reclaimer, scraper), but is not a valid assumption for other types (stacker-reclaimer, grab reclaimer).

Others (Pavlouidakis and Agioutantis 2003a, 2003b) developed a blending pile simulation program to investigate the effects of different operating parameters on the blending and homogenization performance, e.g. the storage capacity, the stacking and reclaiming method, and the number of layers. However, also within this research the geometry of the blending piles is simplified by leaving out the end and begin cones of the blending piles. The work shows some results of the model, but has not been compared with the developed analytical solutions by Gerstel and Van der Mooren. In addition, comparison of results from the simulations with reality was not presented.

The most recent work on homogenization was on the homogenization efficiency of mammoth silos (Schott 2004), where the addressed shortcomings for the specific mammoth silo geometry were overcome by developing a calculation tool for the homogenization efficiency.

There are several major shortcomings associated with these theories. First, they are simplified and do not take the actual geometry of the blending piles into account (the theories of Gerstel, Pavlouidakis). Second, the results of the theories are based on more (Gerstel, Pavlouidakis) or less (Schott) simplified models of the input properties. These simplified stochastic models are useful to determine the homogenization efficiency of blending piles and multiple bins analytically. However, nowadays more complicated pre-processing stages may give cause for the use of more complex time series models for the input properties. Finally, the use of different types of reclaiming processes demands a more detailed analysis of the unloading process.

SIMULATION MODEL

In order to study the homogenization efficiency of a combined loading/unloading operation a discrete event simulation model of a bulk handling system is built. Main characteristic is the modelling of a (continuous) input flow of bulk material with the use of discrete bulkelements. The model consists of two important phases. In the first phase, loading, for each bulkelement in the input stream the actual location in the intermediate pile (blending pile, ship hold) is determined. Second, based on the specifications of the unloading process, the bulkelements in each reclaiming step are determined, homogenized and put in the output stream.

The method thus tracks the positions of each individual bulkelement from the input stream, as they are first loaded and then unloaded. However, the physical interaction between elements is not modelled as is the case in discrete elements methods (DEM). Thus, the dynamic behaviour of elements during loading and unloading is not modelled, only the static positions of the bulkelements in the storage pile and in the output stream. The only interaction between elements results from the fact that each position in the intermediate storage pile can only be occupied by 1 element at the time. Furthermore, the shape of a pile should mimic the angle of repose of the specified material. Thus, when loading an element into the pile its position in the pile is influenced by the positions of the previous elements in that pile.
In fact, the model will calculate the order and property of the bulkelements in an output stream, given a certain input stream and the characteristics of the handling processes (see figure 3). From each of these two streams, the fluctuations in element property values can be determined and the homogenization efficiency can be calculated.

Figure 3: Bulkelements in input and output stream

The model contains four important component types, namely bulkelement, loader, pile and unloader. The bulkelements are representing the bulk material, the loader models the input stream of material, the pile defines the organization of the bulkelements during storage and the unloader is used to simulate the reclaiming process and the composition of the output stream.

Assumptions and model choices

The component 'bulkelement' is used to model the bulk material. Each bulkelement has an attribute-value to define the property of that element. A bulkelement is a box-shaped, homogenous mass of bulk material, which volume is user-defined. Based on chosen element volume, the width, depth and height of the bulkelement are calculated. Width and depth are equal, and the ratio between width and height is used to define the angle of repose of the material. If from two adjacent stacks of bulkelements one stack has exactly one element less than the neighbouring stack, then the resulting slope that is defined by the two stacks equals the repose-angle of the bulk material (see figure 4).

Figure 4: Angle of repose in adjacent stacks of bulkelements

In the model, the loader is the input generator for the bulkelements. Bulkelements are generated as a single input stream, similar to a material stream from a belt-conveyor. With a fixed inter-arrival time, the loader creates a new bulkelement and assigns a property-value. The property value is determined to simulate a user defined (periodic) signal; f.i. white noise, sinusoidal, square-wave. The loader drops the bulkelement at a position representing the belt conveyor head; this position can be a fixed (x,y)-coordinate (resulting in a cone-shaped stockpile) or the drop-position can change between consecutive drops according to a user defined pattern.

The pile is used for modelling the storage phase of the transfer operation. After the bulkelement is dropped by the unloader, it will fall on top of a stack of other elements. If there exists an adjacent stack with at least 2 less elements, the dropped bulkelement will move to that position. The falling will continue until no lower position can be found. The pile can resemble either a ships hold or a stockpile. In the first case, material flow is bounded by fixed walls that prevent further falling.

The unloader component models the reclaiming process. All bulkelements are removed from the pile and will form an output stream. Two reclaiming methods are modelled, a so-called 'slicer' and a 'grab' (see figure 5). The slicer will deconstruct the pile slice by slice: an entire cross-section of the pile is removed, homogenized, and put in the output stream. The grab will remove a box-shaped sub-section of the pile. The entire content of a slice or grab is assumed to be homogeneous. Therefore, the average property of the elements in a slice or grab is calculated and assigned to all the bulkelements in that slice or grab.

For both methods the trajectory of the unloader along the pile is of importance. To describe these trajectories, an x- and y-axis is defined. The x-direction is defined in the longitudinal direction of the pile, the y-axis is perpendicular to the pile.

The slicer works in the longitudinal direction of the pile; it starts at one side of the pile at x=0, takes all the elements at that x-position, and then moves to the next position, thus increasing x. The grab will start at one corner of the pile at (x=0, y=0), grabs until no material is left and then increases y. After an entire cross-section is removed, the grab will move to the next position in the x-direction. For both methods, it is assumed that after a reclaim action, the remaining elements in the pile do not fall or slide.

Figure 5: Model view of a grab

Performance measurement

In homogenization theory, the homogenization efficiency $\eta$ is defined by the reduction of property fluctuation. The fluctuation of a stream is represented by the (statistical) deviation. The property deviation of the output-stream is divided by the deviation in the input stream (see equation 1).

$$\eta = \frac{\sigma_{out}}{\sigma_{in}}$$

A value of $\eta$ below 1 indicates that the homogeneity of the material is improved. The closer the efficiency is towards zero, the better the homogenization effect of a process.
Perfect homogenization would mean that the property of each bulkelement after the process is equal to the average of the input stream. Beside the input stream fluctuation also the loader trajectory and the unloader trajectory influence the homogenization efficiency.

**IMPLEMENTATION**

The model is implemented using Borland Delphi and the discrete event simulation package TOMAS (Veeke and Ottjes 2002). After implementation, verification has taken place using standard methods like balance checks, event tracing and run-time visualization. As is typical for a method that is based on discretization, it is expected that the accuracy of the model will depend on the user-specified volume of a bulkelement. The use of smaller bulkelements will improve the accuracy; eventually for small enough bulkelements the results will converge to a single value. To test this hypothesis a range of experiments is conducted with equal parameter settings and varying volumes for the bulkelement. For this group of experiments, a pile length of 0 is chosen; thus the loader stays in one position, resulting in a circular, cone-shaped pile. Furthermore, the input stream is a periodic square wave signal, alternating between +1 and -1 and consists of exactly 4 periods.

Result of this experiment is presented in figure 6.

**Table 1: Verification settings**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of elements</td>
<td>200 - 3276800</td>
</tr>
<tr>
<td>Square wave number of periods</td>
<td>8 - 2097152</td>
</tr>
<tr>
<td>Pile length</td>
<td>0 meter</td>
</tr>
<tr>
<td>Pile number of layers</td>
<td>n.a.</td>
</tr>
<tr>
<td>Pile volume</td>
<td>32768 m³</td>
</tr>
<tr>
<td>Unloader type</td>
<td>slicer</td>
</tr>
</tbody>
</table>

**Figure 6: Verification results**

From the results of the verification experiments it can be concluded that stable outcomes can be reached when enough elements are used to model the pile. In figure 6, the differences in outcomes when more then 4096 elements are used is small. It is suggested that for this specific problem size, 30.000 (or more) bulkelements will produce satisfying results.

**VALIDATION**

Validation experiments are executed to compare the model results with the analytical results of the theory of Gerstel (1979). This theory states that the efficiency for a stationary, ergodic signal is equal to the inverse of the square root of the number of layers, see equation 2.

\[ \eta = \frac{1}{\sqrt{N}} \]  

(2)

A series of experiments are conducted with a white noise input stream for the element property; other settings are found in table 2. The result is presented in figure 7.

**Table 2: Validation settings**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of elements</td>
<td>200 - 3276800</td>
</tr>
<tr>
<td>White noise deviation</td>
<td>0.25</td>
</tr>
<tr>
<td>Pile length</td>
<td>100 meter</td>
</tr>
<tr>
<td>Pile number of layers</td>
<td>2 - 256</td>
</tr>
<tr>
<td>Pile volume</td>
<td>167.8 m³</td>
</tr>
<tr>
<td>Unloader type</td>
<td>slicer</td>
</tr>
</tbody>
</table>

**Figure 7: Validation results**

As is shown in figure 7, the experimental result and the value calculated with Gerstel theory coincide. The largest relative difference was 0.42%. It can be concluded that the model results follow the available theory.

**EXPERIMENTS AND RESULTS**

Three categories of experiments are presented in this paper. First, instead of using a signal with only a white noise disturbance, a more realistic input stream is modelled. The second group of experiments studies the effect of the physical process of falling material and the resulting begin- and end cones of a realistic stockpile. Finally, the difference between a slice-type and a grab-type reclamer is shown.
Input stream characteristics

Beside the use of white noise input signals, the simulation model can run for a wide range of different types of input streams. A kind of ‘worst-case’ scenario might be an input stream that consists of an alternation between two different properties; for instance, when the input stream is actually blending two streams with a different average property. To study the models usability for this kind of scenario, experiments are conducted with a square wave signal. Of special importance is the relationship between the number of periods in the input stream and the number of layers in the pile. In the worst case, each layer would contain exactly 1 period, resulting in a poor homogenization. On the other hand, if the number of layers is exactly twice the number of periods, that would result in perfect homogenization.

<table>
<thead>
<tr>
<th>Table 3: Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of elements</td>
</tr>
<tr>
<td>Squarewave number of periods</td>
</tr>
<tr>
<td>Pile length</td>
</tr>
<tr>
<td>Pile number of layers</td>
</tr>
<tr>
<td>Pile volume</td>
</tr>
<tr>
<td>Unloader type</td>
</tr>
</tbody>
</table>

![Figure 8: Input stream characteristics](image)

Both extremes are clearly visible in the experimental results. With 50 periods in 100 layers, the efficiency is almost zero (which is perfect), with 100 periods in 100 layers the efficiency approaches 1. The minimal difference between 0 and the experimental value 0.028 (at 50 periods) is caused by the fact that falling of elements causes some mixing by itself.

Pile geometry

The analytical model of Gerstel assumes that a pile is built up with identical layers of material. In reality, material is dropped on top of a pile and falls along the slope, resulting in a ‘triangular’ shaped pile with begin- and end cone. In order to determine the effect of element falling and the effect of begin- and end cone, several experiments are done. The pile length is varied between 0 and 500 meter.

![Figure 9: Pile geometry](image)

In above figure, a pile length of 0 means a round cone. It can be seen that the impact of the begin and end cone on the homogenization is large for short piles. For longer stockpiles, the effect of begin and end cone becomes neglectable.

Reclaiming methods

The behaviour of the slicer in the model is similar to the behaviour of a bridge or drum-type reclaimer. The grab resembles unloading a pile with a grab-crane. To study the difference in homogenization effects for both unloading methods, the results of the first group of experiments can be compared with experiments using a grab.

<table>
<thead>
<tr>
<th>Table 4: Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of elements</td>
</tr>
<tr>
<td>Squarewave number of periods</td>
</tr>
<tr>
<td>Pile length</td>
</tr>
<tr>
<td>Pile number of layers</td>
</tr>
<tr>
<td>Pile volume</td>
</tr>
<tr>
<td>Unloader type</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 5: Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of elements</td>
</tr>
<tr>
<td>Squarewave number of periods</td>
</tr>
<tr>
<td>Pile length</td>
</tr>
<tr>
<td>Pile number of layers</td>
</tr>
<tr>
<td>Pile volume</td>
</tr>
<tr>
<td>Unloader type</td>
</tr>
</tbody>
</table>
Figure 10: Reclaiming method

Comparison of the grab and slice reclaiming method shows that in general slicing leads to better homogenization. This can be explained by the fact that the volume content of a grab is much less than a slice. Furthermore, a grab intersects much less layers in the pile than a slice. Both effects result in less homogenization.

CONCLUSIONS

Finite element simulation is an excellent tool to study homogenization effects. As is shown by the validation experiments, the results of elaborate analytical calculations can be reproduced by the simulation model. Furthermore, the new approach enables a far more elaborate study of factors that influence the homogenization of handling processes. Results are given for other input stream characteristics, more realism in physical behaviour of bulk material falling and the effect of the use of other reclaiming methods.

More work can be done on the following subjects. First, the behaviour of elements after a reclaiming action has taken place. In reality, the material will fall or slide to a lower position if possible. Further, the model enables the study towards improvement in loader and unloader trajectories and strategies. Finally, more study of input stream characteristics is required. Instead of using periodic signals, the model allows the use of more realistic input properties, including actual sensor data. The use of real-world data would further improve the quality of the experimental outcomes.

REFERENCES


FLUID FLOW SIMULATION
CFD MODELING OF THE SDS2 INJECTION SYSTEM FOR CANU6 NUCLEAR REACTOR

I. Prisecaru, S. Nicolici and D. Dulpeac
Power Plant Engineering Faculty
Politehnica University of Bucharest
313 Splaiul Independentei, 060042, sector 6,
Bucharest, Romania
E-mail: prisec@cne.pub.ro,
nicolici.stefan@gmail.com danielu@cne.pub.ro

I. P. Nita
Subsidiary of Engineering and
Technology for Nuclear Projects
Magurele - Ilfov
409 Atomistilor Street, POB: 5204-MG-4
nitai@router.citnon.ro

KEYWORDS: Nuclear Engineering, CFD Modeling, Liquid Injection, CANDU Reactor.

ABSTRACT

This paper describes the work of developing a 3D CFD model for analyzing the liquid poison jet velocity and concentration which is necessary for the performance analysis of SDS2 of CANDU600. As a boundary condition the liquid poison injection rate through each injection hole was given and the 3D CFD model for analyzing the downstream flow and concentration field is developed and validated. As the ratio of the calandria tank radius and the injection hole diameter drilled at the injection nozzle is as big as 1000, the grid generation for the calandria is very difficult as the number of elements can easily exceed the computational load limit of the available work station. Thus to overcome this difficulty sections from calandria vessel were chosen to investigate the process. Utilizing the experiment data from Bhabha Atomic Research Center in India, the jet growth rate was validated and the results were encouraging. Based on the results, the 3D CFD model developed in this paper shows physically reasonable results and will be used for the analysis of SDS2 performance test along with the reactor physics codes.

INTRODUCTION

The purpose of the shutdown systems present in each nuclear reactor is to quickly reduce the reactor power during abnormal conditions that can lead to unwanted situations such as fuel and fuel channel defects. Thus, security systems are present to ensure reactor safety in any circumstances that may arise in reactor operation.

CANDU (CANada Deuterium-Uranium) is a Canadian-invented, pressurized heavy water nuclear reactor. It has two emergency shutdown units, called SDS1 and SDS2 (Shut-Down Systems), which are completely independent of each other (AECB 1993). They are also very different in design and operation method, but each one can stop the fission chain reaction during accident conditions. Since the objective of this study is the CFD (Computational Fluid Dynamics) modeling of the SDS2 injection system, SDS2 construction and operating characteristics are presented. Operation of SDS2 is a fast insertion of a large amount of poison in reactor moderator that will act as a strong absorber of thermal neutrons. The poison is a liquid solution of gadolinium nitrate (with a concentration maintained at 8000 ppm) and heavy water. The system has six identical injection lines, each consisting of:
- One tank containing a solution of gadolinium nitrate. Above the liquid level there is a float which will act as a piston;
- One connecting pipe line with the Calandria Vessel. On this line there is an normally open isolation valve;
- Injection Nozzle Subassembly. It has the role to distribute as efficiently as possible the liquid poison in the moderator;
- High-Pressure Injection System. Helium contained in high pressure cylinders is used as the drive;
- Pressure Balance Line. This ensures normal hydrostatic equilibrium in the system and prevents accidental poisoning of the moderator.

The injection nozzle subassembly (RAAN/CITON 2001), shown in Figure 1, with an alternative configuration, is a 2.20 in outside diameter (O.D. = 55.9 mm) zircaloy tube, with four rows of orifices along its length, having an internally threaded nut on the inlet end and a bayonet fitting in a nosepiece on the other end. During an injection the liquid poison enters the nozzles from the injection tube and emerges from subassembly orifices as jets which merge to form continuous poison curtains in the moderator. The tubing wall is thickened at the orifice positions to improve the penetration of the poison jets. The orifice diameter of 1/8 in (3.2 mm) was established by testing of the injection nozzles.

In each row, the orifices are arranged in groups of four, each group positioned to shoot its jets between the adjacent calandria tubes. This grouping of orifices significantly improves poison curtain growth since jets do not impinge directly on adjacent calandria tubes, and because closely spaced jets tend to reinforce one another. An even closer spacing was not used because the groups of jets took longer to merge into a continuous poison curtain than did the 1.25 in (32 mm) spacing used.
The injection nozzles are made with two orifice orientations. In one, the holes direct the poison jets parallel and perpendicular to calandria tubes; in the other, the holes direct the poison jets at 45° to the calandria tubes. These two orifice orientations make possible better coverage of the liquid jets and consequently a more effective injection. Figure 2 shows schematically the orientation of the injection orifices and positioning in the calandria vessel (Rouben 2010).

![Figure 1: Injection Nozzle Subassembly](image)

![Figure 2: Positions of Liquid-Poison-Injection Nozzles (a-Side elevation view; b-Plan view) - all dimensions are in cm](image)

This study follows to the previously analysis (Prisecaru et al. 2010) and aims to simulate using the computer code Ansys-CFX, 3D calandria vessel sections during the operation of SDS2 system. The results allow a better understanding of the injection and dispersion phenomena during and after the system operation.

**ANSYS-CFX MODELS DESCRIPTION**

Since the location of injection tubes in the moderator does not have symmetrical arrangement and the geometric dimensions varies from few millimeters to several meters (hole injection: 3.2 mm to calandria vessel diameter: 7.65 m) it was decided to model sections of calandria vessel that will provide adequate information on the phenomenon of poison injection into the moderator with reasonable demands on the computer system (Song et al. 2008).

**Selection of calandria vessel sections and mesh creation**

The first section chosen for modeling includes ¼ of the injection tube with holes positioned at 45° relative to calandria tubes (see Figure 3). Calandria channels placed in a square grid have a pitch length of 28.6 cm.

The mesh contains 2,366,199 elements and 440,776 nodes being composed of tetrahedrons. Curved surfaces are sectioned at an angle of 18°; the minimum size of the elements is 0.66 mm and corresponds to injection orifices faces.
The second section (see Figure 4) contains the injection holes of all the six injection nozzles between two calandria tubes. Given the fact that jets are directed between calandria tubes, injection can be considered to have identical behavior between any two calandria tubes but this assumption is valid only to describe the phenomena for a short period of time when the moderator streamlines are not yet affected by the upper gas cover. The symmetry of the injection nozzle in orifice positioning between two fuel channels allows to model only eight holes for each pipe.

![Figure 3: Section I geometry and boundary conditions](image)

The mesh grid for Section II contains 1,728,804 elements with 343,140 nodes.

![Figure 4: Section II geometry and boundary conditions](image)

**Field sub-models**

The research field of computational fluid dynamics is based on the basic conservation equations for mass, momentum, energy (and species concentration). By solving a set of nonlinear partial differential equations, more accurate results can be obtained, e.g. transient behavior, movement due to convection and transport due to laminar and turbulent diffusion.

The aim of computational fluid dynamics is to numerically solve the partial differential equations that govern fluid flows. Two general equations are available for simulations without heat transfer, namely:

- The mass of a fluid is conserved (continuity):

\[
\frac{\partial \rho}{\partial t} + \nabla (\rho \vec{v}) = 0, \tag{1}
\]

- The rate of change of momentum equals the sum of the forces on a fluid particle (Navier-Stokes, Newton’s 2nd law):

\[
\frac{\partial }{\partial t} (\rho \vec{v}) + \nabla (\rho \vec{v} \vec{v}) = -\nabla p + \nabla (\nabla \vec{v}) + \rho \vec{g}, \tag{2}
\]

in which \( \vec{\tau} \) represents the stress tensor given by:

\[
\vec{\tau} = \mu \left[ \nabla \vec{v} + \nabla \vec{v}^T \right] - \frac{2}{3} \nabla \cdot \vec{v} \vec{I}, \tag{3}
\]

The liquid solution of gadolinium nitrate dissolved in heavy water can be considered as one fluid consists of two parts well mixed at molecular level (ANSYS INC. 2010). Thus the fluid model will have two components and the mean properties are based on mass concentrations of gadolinium nitrate and heavy water.

For a multi-component fluid, scalar transport equations for velocity, pressure, temperature and other properties of interest are resolved. However, additional equations must be included to determine how the components are transported in the fluid.

The bulk motion of the fluid is modeled using single velocity, pressure, temperature and turbulence fields. The influence of the multiple components is felt only through property variation by virtue of differing properties for various components. Each component has its own equation for conservation of mass:

\[
\frac{\partial \tilde{\rho}_i}{\partial t} + \frac{\partial (\tilde{\rho}_i \tilde{U}_j)}{\partial x_j} = -\frac{\partial}{\partial x_j}\left( \tilde{\rho}_i \left( \tilde{U}_j - \tilde{U}_j \right) - \rho \left( \tilde{U}_j - \tilde{U}_j \right) \right), \tag{4}
\]

where \( \tilde{\rho}_i \) is the mass-average density of fluid component \( i \) in the mixture, that is, the mass of the component per unit volume, \( \tilde{U}_j = \sum (\tilde{\rho}_i \tilde{U}_j) / \tilde{\rho} \) is the mass-average velocity field, \( \tilde{U}_j \) is the mass-average velocity of fluid component \( i \), \( \rho \left( \tilde{U}_j - \tilde{U}_j \right) \) is the relative mass flux.

The relative mass flux term accounts for differential motion of the individual components. This term may be modeled in a number of ways to include effects of concentration gradients, a pressure gradient, external forces or temperature gradient. Of these possible sources of relative motion among the mixture components, the primary effect is that of concentration gradient. The model for this gives rise to a diffusion-like term:

\[
\rho \left( \tilde{U}_j - \tilde{U}_j \right) - \frac{\Gamma_j}{\tilde{\rho}} \frac{\partial \tilde{\rho}_i}{\partial x_j}, \tag{5}
\]

The molecular diffusion coefficient, \( \Gamma_j \), is assumed to be equal to \( \rho D_j \), where \( D_j \) is the kinematic diffusivity set on the fluid models tab for a domain in Ansys-CFXPre (ANSYS INC. 2010).
Turbulence modeling

Two-equation turbulence models are very widely used, as they offer a good compromise between numerical effort and computational accuracy (Kandakure 2008). Two-equation models are much more sophisticated than the zero equation models, both velocity and length scale are solved using separate transport equations.

The k-ε and k-ω two-equation models use the gradient diffusion hypothesis to relate the Reynolds stresses to the mean velocity gradients and the turbulent viscosity. The turbulent viscosity is modeled as the product of a turbulent velocity and a turbulent length scale.

In two-equation models, the turbulence velocity scale is computed from the turbulent kinetic energy, which is provided from the solution of its transport equation. The turbulent length scale is estimated from two properties of the turbulence field, usually the turbulent kinetic energy and its dissipation rate. The dissipation rate of the turbulent kinetic energy is provided from the solution of its transport equation.

The k-ε model is based on the eddy viscosity concept, so that \( \mu_{\text{eff}} = \mu + \mu_t \), where \( \mu_t \) is the turbulence viscosity. The k-ε model assumes that the turbulence viscosity is linked to the turbulence kinetic energy and the dissipation via the relation:

\[
\mu_t = C_\mu \rho \frac{k^2}{\varepsilon},
\]

where \( C_\mu \) is a constant. The values of \( k \) and \( \varepsilon \) come directly from the differential transport equations for the turbulent kinetic energy and the turbulence dissipation rate. In this paper the k-ε model is employed because it has proven to be stable and numerically robust and has a well established regime of predictive capability. For general purpose simulations, the k-ε model offers a good compromise in terms of accuracy and robustness.

Boundary and initial conditions

The boundary conditions imposed on the two models are presented in Figure 3 and Figure 4. The properties of D_2O and gadolinium nitrate solution are presented in Table I (Chae 2005).

<table>
<thead>
<tr>
<th>D_2O density</th>
<th>Gd(NO_3)_3 sol. density</th>
<th>Viscosity</th>
<th>Diffusion coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1098 kg/m^3</td>
<td>1127 kg/m^3</td>
<td>8.5e-4 kg/m s</td>
<td>5.6e-7 kg/m s</td>
</tr>
</tbody>
</table>

The poison injection velocity through the holes is assumed uniform and equal to 27.786 m/s, the concentration of Gd(NO_3)_3 at the entrance is 8,000 ppm. The helmet cover above the moderator level is modeled through an Open Boundary Condition with the relative pressure set at 0 Pa (reference pressure is 1 bar).

The time step is 0.02 s for Section I model and 0.01 s for Section II model, and the velocity field at t = 0 s is assumed to be zero. The Second Order Backward Euler scheme is used as integration method in time. This is an implicit time-stepping scheme, but is second-order accurate. It is applicable for constant and variable timestep sizes. The residual target RMS for the conservation equations is set at 1e-5.

RESULTS AND DISCUSSIONS

In Figure 5 is presented an isosurface for the Gd(NO_3)_3 concentration at t = 1 s. One can observe how the solution stream enters through the fuel channels, forming a continuous. Throughout the transient process the poison fronts cover the same distance regardless of the position between any two fuel channels. This confirms the hypothesis considered in the selection of Section II, that for the injection period the phenomena have axial symmetry relative to the injection nozzle.

Velocity vectors for a 45° injection nozzle at t = 0.98 s are shown in Fig.6. The magnitude of the vectors at the orifices exit prove that the jet momentum dissipates immediately after leaving the holes so that at the first channels row level the fluid speed is nearly an order of magnitude smaller. Liquid jet set in motion the adjacent fluid, creating areas where fresh liquid is drawn toward the jet. This considerably improves the poison dilution as the jet penetrates deeper through the fuel channels.

![Figure 5: Poison mass-fraction isosurface at t = 1 s](image)

![Figure 6: Velocity vectors at t = 0.98 s](image)

In the case of the Section II the concentrations of gadolinium nitrate are presented for two planes: perpendicular to the fuel channels (Figure 7) and parallel
to the channels (Figure 8). Note that the absolute character of the process in this case is similar to that of model I. In Figure 8 the diffusion phenomena is presented on a plane section that includes all the injection nozzles. At $t = 1$ s the $90^\circ$ and $45^\circ$ jets begin to interpenetrate, but one can clearly distinguish areas with high concentrations for each jet.

**Figure 7:** The poison concentration contour for $90^\circ$ nozzle

**Figure 8:** The poison concentration contour - parallel plane to the fuel channels

**Comparison with experimental data results**

The most important parameter characterizing the efficiency of the SDS2 operation is the jet penetration length in the moderator mass. Thus, to confirm the results obtained, the simulation lengths were compared with experimental results BARC2 test, (Chae 2001). From Figure 9 one can observe that the two sets of values are very close, but throughout the injection the Ansys-CFX model slightly underestimates the penetration length. This could be given by the different velocity profile at the injection nozzle exit; for the CFD models it is assumed a uniform velocity, although in the experimental test it has a certain variation.

**CONCLUSIONS**

The objective of this study was to simulate using the computer code Ansys-CFX, 3D calandria vessel sections during the operation of SDS2 system. Two models were studied and the poison concentration contours and velocity vectors were obtained.

![Figure 9: Comparison between jet penetration lengths: Ansys-CFX models and experimental BARC2 test](image)

During SDS2 operation (about 1 s) the jets from all six nozzles do not have an influence on each other, each having a similar evolution in time. But for times longer than 1 s, the poison fronts begin to interfere and a dilution process based on concentration gradients and jets momentum will further mix the poison in the moderator. For even longer times the poison spread will be given by the heavy water forced circulation in the cooling system.

The result of this paper can be further used in a nuclear reactor analysis of the SDS2 operation.

**REFERENCES**


HYBRID MODELING AND SIMULATION OF A
MULTI-PRODUCT REACTIVE DISTILLATION
COLUMN

Ramon R.H. Schieffelers * Edwin Zondervan ** Jacobus E. Rooda *
André B. de Haan **

* Department of Mechanical Engineering, Systems Engineering
** Department of Chemical Engineering, Process Systems Engineering
Eindhoven University of Technology, P. O. Box 513, 5600MB, Eindhoven, The
Netherlands

KEYWORDS
Chi, Dynamics, Hybrid modeling, Multi-product, Reactive dis-
tillation, Simulation.

ABSTRACT
In this paper we formulate a dynamic model of a multi-product
reactive distillation (RD) column. Such model can be used to
determine optimal product switching policies. The dynamic
RD model is hybrid in its nature; the material balances, the
kinetics, etc. are continuous, while the product switching policy
is discrete. Therefore, we propose to formulate the model in
the language Chi. This language is a hybrid process algebra
for modeling, simulation, verification and control of hybrid
systems. We demonstrate that a reactive distillation process
can be conveniently modeled in Chi and we also show how to
implement a PI controller with anti-windup to track the changes
in product quality by adaptation of the feed flow. The Chi
simulator is used to obtain the presented simulation results.

1. INTRODUCTION

A reactive distillation (RD) process is difficult to design and
to control because reaction and distillation have complicated
interactions. Most models reported in literature (see Almeida-
Rivera et al. (2004); Bezzo et al. (1999)), are steady-state
models and only few publications (see Taylor and Krishna
(2000); Peng et al. (2003)) are devoted to dynamic modeling
and control of a RD column. The dynamic behavior of RD is
poorly understood and for that reason reliable process models
are required.

Currently RD is increasingly used in the sector of specialty
chemicals because it may be operated as a continuous process
that can be used to switch (easily) from production of one
product to another. This is the reason that multi-product RD, as
many other chemical process systems, is hybrid in its nature;
part of the model is modeled in the continuous-time (CT)
domain, e.g. the material balances, the kinetics, etc., and part
is modeled in the discrete-event (DE) domain, e.g. the product
switching policy.

The contribution of this paper is twofold: 1) a dynamic for-
mulation of the RD column has been defined. It is used in a
control system to obtain a profile for the feed that switches
the product quality from one steady state to another (the product
transition). For this purpose, 2) we developed a dynamic model
of RD column and a PI feed controller with anti-windup in a
hybrid process algebra called $\chi$ (Chi). In the following parts
we introduce the concepts of RD and motivate our choice for $\chi$.

1.1 Reactive distillation

Traditionally, reactants are fed to a reactor in which they are
transformed into products. As the product leaves the reactor
together with unreacted reactants and possibly some side prod-
ucts, often a purification step is required, for example distil-
lation. In RD, the reactor and the distillation step are com-
bined into one unit operation. As reactants are transformed
into products, the product is immediately purified by means
of separation. This fast product removal is also advantageous
for the conversion, as the chemical equilibrium is shifted to
the right side of the reaction. Also the size of a RD column is
dramatically smaller than the traditional reactor-separator setup.
An illustrating example is the commercialization of the East-
man Chemical Methyl Acetate process that was traditionally
operated in a reactor and a separation train of 9 distillation
units. With the aid of RD, the process could be intensified; a
single unit could generate the same product quality and capacity
(see Krishna (2002)). Other advantages of RD are that it can
be operated in a continuous way and product transitions can
be relatively easy made. RD has become a mature technology
and in these days new applications can be found in specialty
chemicals. RD is proven useful for bio-fuel production (Mahfud
et al. (2007)), but also for fatty-acid ester production (de Jong
et al. (2009)) and for polyester production (Shah et al. (2009)).
In the latter case, the concentration of the polyester determines
the polymer grade that is obtained, where each polyester grade
can be marketed as a different product. The concentration
of polyester can be manipulated by changing the feed conditions
of the system and in that light RD for polyester production is
an interesting control problem. In this paper we develop a RD
model for a simpler reaction: $aA + bB \rightarrow cC$ (for example:
Ethylene oxide + water $\rightarrow$ ethylene glycol). We assume that
the concentration of formed product $C$ influences the product qual-
ity. This model is controlled by a PI controller that tracks the
changes in product quality (product transition) by adaptation of
the feed flow. Such process has continuities (e.g. the composi-
tions at the stages) and discontinuities (e.g. the switching of
the required product qualities). For this reason we propose to use a
hybrid formalism.
1.2 \( \chi \) (Chi)

Nowadays, there exists a variety of hybrid modeling formalisms and associated simulation tools. Considering only languages that use high-level language elements, i.e., language elements to specify systems in an intuitive way, independently of the implementation details of the languages and the solving algorithms used, there is a big diversity among them. For a classification and overview of formalisms, see Beek and Rooda (2000).

For modeling the RD column and its environment, including transport, distribution and storage of reactants/products in a multi-product environment, a formalism is required that includes high level language elements for continuous time modeling as well as high level language elements for discrete-event modeling. The language \( \chi \) (see Man and Schifferleers (2006); Beek et al. (2006b)) is such a hybrid formalism. It has formal semantics and a relatively straightforward elegant syntax that is very intuitive and highly suited to modeling. The intended use of \( \chi \) is for modeling, simulation, verification, and real-time control of industrial systems. It integrates continuous-time and discrete-event concepts, and enables analysis of the dynamic behavior of hybrid processes, of hybrid embedded systems, as well as of complete hybrid plants. In this paper, the language \( \chi \) is introduced and used to model a dynamic multi-product RD column controlled by a PI controller with anti-windup. In this case, both the plant and the controller are modeled in the continuous-time (CT) domain. The requested products, and accompanying product changes are modeled in the discrete-event (DE) domain. As described in Schifferleers et al. (2009), \( \chi \) facilitates studying plant/controller combinations modeled completely in the CT domain, modeled completely in the DE domain, or modeled in an arbitrary CT/DE combination.

1.3 Outline

The outline of this paper is as follows. In Section 2, a dynamic model for a RD column is formulated. The syntax and semantics of \( \chi \) are described in Section 3. Section 4 describes the \( \chi \) model of the reactive distillation column. Simulation results are shown in Section 5. Conclusions and an outlook for future work are given in Section 6.

2. MODEL FORMULATION

The reactive distillation model consists of \( I \) (\( i = 1, \ldots, I \)) stages. The first stage is the condenser from which distillate flows with flow \( D \). The last stage is the reboiler stage, from which bottom product flows with flow \( B \). At the feed stage the reactants are fed to the column with flow \( F \) and composition \( z_j \). The mixture contains components \( j \in \{ A, B, C \} \), from which \( A \) and \( B \) actually react into \( C \). In the feed stage, as well as in several other stages this chemical reaction takes place. Each of the stages contains the components in liquid phase \( x_{i,j} \) and in vapor phase \( y_{i,j} \). The vapor and liquid phases are related by relative volatility \( \alpha_{j} \). The liquid hold-up in a stage is denoted by \( M_i \). Vapor flows between the stages are expressed as \( V_i \) and liquid flows between the stages are expressed as \( L_i \). In Figure 1 a schematic overview of a 5-stage reactive distillation column is shown. The mass balance for a total condenser (\( i = 1 \)) and \( \forall j \):

\[
M_i \frac{dx_{i,j}}{dt} = V_{i+1} y_{i+1,j} - L_i x_{i,j} - D x_{i,j}
\]  

(1)

Fig. 1. Schematic overview of a 5-stage RD column.

The mass balance for a reactive stage \( i \) and \( \forall j \):

\[
M_i \frac{dx_{i,j}}{dt} = L_i x_{i,j+1} + V_{i+1} y_{i+1,j} - L_i x_{i,j} - V_{i} y_{i,j} + r_{i,j}
\]

(2)

The mass balance for a reactive feed stage \( i \) and \( \forall j \):

\[
M_i \frac{dx_{i,j}}{dt} = L_{i-1} x_{i,j} + V_{i+1} y_{i+1,j} - L_i x_{i,j} - V_{i} y_{i,j} + r_{i,j} + F z_j
\]

(3)

Finally, the mass balance for a reboiler (\( i = I \)) and \( \forall j \):

\[
M_i \frac{dx_{i,j}}{dt} = L_{i-1} x_{i-1,j} - V_i y_{i,j} - B x_{i,j}
\]

(4)

For the vapour/liquid composition at each stage (excluding the condenser), the relative volatilities are used:

\[
y_{i,j} = \frac{\alpha_{j} x_{i,j}}{\sum_{j} \alpha_{j} x_{i,j}}
\]

(5)

At the reactive stages a reaction occurs according to:

\[
\xi_A A + \xi_B B \xrightarrow{E_{x}} \xi_C C
\]

(6)

A typical example for RD is the production of ethylene glycol from ethylene oxide and water. The reaction rate for such reaction is given by:

\[
r_{i,j} = \frac{\xi}{k} x_{i,A} x_{i,B} \exp(-E_{A}/RT)
\]

(7)

In this model we assume negligible vapor hold-up and liquid hold-up at each stage is given, perfect control of levels using \( D \) and \( B \) (LV configuration), constant molar flows (which replace the energy balance), vapor-liquid equilibrium (VLE) on all stages, constant relative volatility for the VLE and a constant liquid hold-up (i.e., neglect flow dynamics). Each phase is perfectly mixed in each segment. With these assumptions the only variables are the mole fractions \( x_i \) of the components on each stage.

In this article, a five stage reactive distillation model has been used for simulations. Stage 1 is the condenser stage and Stage 5 is the reboiler stage (\( B = 15 [\text{mol/s}] \)). At Stage 3 feed is added (\( F = 40.0 [\text{mol/s}] \), \( z = (0.4, 0.6, 0.0) \)). At Stages 2, 3 and 4 reaction occurs. Kinetic rate constant \( k \) equals \( 40.000 [\text{mol/s}] \), activation energy \( E_A \) equals \( 20.000 [\text{J/mol}] \), and gas constant \( R \) equals \( 8.3145 [\text{J/molK}] \). For all stages we assume a homogeneous column temperature \( T \) equal \( 300.0 \) [°C]. Liquid hold-up \( M_i \), vapor flow \( V_i \), liquid flow \( L_i \) and initial composition \( x_i \) are
given in Table 1. Relative volatilities for components $j$ equal (2.34, 1.00, 0.44).

<table>
<thead>
<tr>
<th>Stage</th>
<th>$M_i$</th>
<th>$V_i$</th>
<th>$L_i$</th>
<th>$x_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>375.0</td>
<td>-</td>
<td>125.0</td>
<td>(0.33, 0.33, 0.34)</td>
</tr>
<tr>
<td>2</td>
<td>50.0</td>
<td>150.0</td>
<td>125.0</td>
<td>(0.33, 0.33, 0.34)</td>
</tr>
<tr>
<td>3</td>
<td>50.0</td>
<td>150.0</td>
<td>165.0</td>
<td>(0.33, 0.33, 0.34)</td>
</tr>
<tr>
<td>4</td>
<td>50.0</td>
<td>150.0</td>
<td>165.0</td>
<td>(0.33, 0.33, 0.34)</td>
</tr>
<tr>
<td>5</td>
<td>750.0</td>
<td>150.0</td>
<td>-</td>
<td>(0.33, 0.33, 0.34)</td>
</tr>
</tbody>
</table>

3. THE LANGUAGE $\chi$

In this section, the syntax and semantics of the language $\chi$ are discussed informally. We only discuss those concepts that are used in Section 4. In these models, only a limited number of constructs suited for discrete-event specification have been used. However, in future work these constructs, such as channels that can be used for rendez-vous communication between parallel processes, well-known imperative statements such as conditions, loops etc., and structured datatypes, such as lists, sets, tuples, etc., are required. This motivates the choice for a ‘truly’ hybrid formalism, instead of a continuous-time formalism that only includes a limited number of constructs to specify switching behavior. A more detailed explanation of $\chi$ can be found in Man and Schiffelers (2006); Beek et al. (2006b). Information about the freely available toolset can be found at Systems Engineering Group TU/e (2009). This toolset includes, amongst others, a stand-alone simulator (see Beek et al. (2006a)) equipped with symbolic and numerical solvers such as MAPLE (MapleSoft (2005)) and DASSL (Petzold (1983)), and a co-simulator (Beek et al. (2007)) for simulation of models that consist of subsystems modeled using MATLAB/SIMULINK (The MathWorks, Inc (2005)) and subsystems modeled in $\chi$.

3.1 Syntax

A $\chi$ model identified by name is of the following form:

$$\text{model name}() = [\begin{array}{l}
\text{var } s_1 : \text{type } s_1 = e_1, \ldots, s_k : \text{type } s_k = e_k, \\
\text{cont } x_1 : \text{type } x_1 = d_1, \ldots, x_m : \text{type } x_m = d_m, \\
\text{alg } y_1 : \text{type } y_1 = e_1, \ldots, y_n : \text{type } y_n = e_n, \\
\text{end}
\end{array}]$$

Here, type denotes a type, for instance bool, nat, or real. Notation var $s_1 : \text{type } s_1 = e_1, \ldots, s_k : \text{type } s_k = e_k$ denotes the declaration of discrete variables $s_1, \ldots, s_k$ with their respective types $\text{type } s_1, \ldots, \text{type } s_k$ and initial values $e_1, \ldots, e_k$. Similarly, notations cont $x_1 : \text{type } x_1 = d_1, \ldots, x_m : \text{type } x_m = d_m$ and alg $y_1 : \text{type } y_1 = e_1, \ldots, y_n : \text{type } y_n = e_n$ are used to declare continuous and algebraic variables, respectively.

The main differences between discrete, continuous, and algebraic variables are as follows. First, the values of discrete variables remain constant when model time progresses, the values of continuous variables may change according to a continuous function of time when model time progresses, and the values of algebraic variables may change according to a discontinuous function of time. Second, the values of the discrete and continuous variables do not change in action transitions unless such changes are explicitly specified, for example by assigning a new value. The values of algebraic variables can change arbitrarily in action transitions, unless such changes are explicitly restricted, for example by assigning a new value. Third, there is a difference between the different classes of variables with respect to how the resulting values of the variables in a transition relate to the starting values of the variables in the next transition. The resulting value of a discrete or continuous variable in a transition always equals its starting value in the next transition. For algebraic variables, there is no such relation. In most models, the values of discrete variables are defined by assignments, whereas the values of algebraic variables are defined by equations ((in)equalities).

The language consists, amongst others, of the following statements $p, q \in P$:

- $P := x_n := e_n$ (multi-) assignment statement
- $\text{eqn } u \rightarrow P$ equation statement
- $\text{delay } d \leadsto P$ delay statement
- $u \rightarrow P$ while do operator
- $P; P$ sequential composition operator
- $P || P$ parallel composition operator
- $lp(x_k, e_n)$ process instantiation statement

Here, $x_n$ denotes the (non-dotted) variables $x_1, \ldots, x_n$ such that $\text{time} \not\in \{x_n\}$, $e_n$ denotes the expressions $e_1, \ldots, e_n$, $u$ denotes a predicate over variables (including the variable time) and dotted continuous variables, $d$ denotes a numerical expression, and $lp$ denotes a process identifier. The operators are listed in descending order of their binding strength as follows $\leadsto \rightarrow$ . Parentheses may be used to group statements.

A multi-assignment statement $x_n := e_n$ denotes the assignment of the values of the expressions $e_n$ to the variables $x_n$.

An equation statement $\text{eqn } u$, usually in the form of a differential algebraic equation, restricts the allowed behavior of the continuous and algebraic variables in such a way that the value of the predicate $u$ remains true over time.

A delay statement $\text{delay } d$ delays for $d$ time units and then terminates by means of an internal action.

The while do operator $u \rightarrow p$ repeatedly executes $p$ as long as $u$ holds.

Sequential composition operator $p; q$ behaves as process term $p$ until $p$ terminates, and then continues to behave as process term $q$.

Parallelism can be specified by means of the parallel composition operator $p || q$. Parallel processes interact by means of shared variables. The parallel composition $p || q$ synchronizes the time behavior of $p$ and $q$ and interleaves the action behavior (including the instantaneous changes of variables) of $p$ and $q$. The synchronization of time behavior means that only the time behaviors that are allowed by both $p$ and $q$ are allowed by their parallel composition.

Process instantiation statement $lp(x_k, e_n)$, where $lp$ denotes a process label, enables (re-)use of a process definition. A process definition is specified once, but it can be instantiated many times, possibly with different parameters: external variables $x_k$ and expressions $e_n$.

$\chi$ specifications in which process instantiations $lp(x_k, e_n)$ are used have the following structure:

$$pd_1, \ldots, pd_j$$
where for each process instantiation \( l_p(x_k, e_n) \) occurring in process term \( q \), a matching process definition \( q_i \) (1 \( \leq i \leq j \)) of the form proc \( l_p(x_k, e_n) = \{ D : p \} \) must be present among the \( j \) process definitions \( q_1 \ldots q_j \). Here \( l_p \) denotes a process label, \( x_k \) denotes the "actual external" variables \( x_1, \ldots, x_k, e_n \) denotes the expressions \( e_1, \ldots, e_n \), \( x_k' : t_{x_k} \) denotes the "formal external" definitions \( x_k' : t_{x_k} \), and \( v_n : t_{v_n} \) denotes the "value parameter definitions" \( v_1, t_{v_1}, \ldots, v_n : t_{v_n} \). Notations \( D \) and \( D' \) denote declarations of local (discrete, continuous or algebraic) variables. The only free (i.e. non-local) variables that are allowed in process term \( p \) are formal external variables \( x_k' \) and value parameters \( v_n \) respectively. We assume that formal external variables \( x_k' \) and value parameters \( v_n \) are different.

3.2 Formal semantics

The semantics (meaning) of \( \chi \) is unambiguously defined by means of deduction rules in SOS style (see Plotkin (2004)) that associate a hybrid transition system with a \( \chi \) model as defined in Man and Schifflers (2006). Such hybrid transition system is a mathematical model that consists of action transitions and time transitions. Action transitions define instantaneous changes, where time does not change, to the values of variables. Time transitions involve the passing of time, where for all variables their trajectory as a function of time is defined. Its purpose is to serve as a basis for understanding, and for various kinds of analysis and verification.

4. MODELING THE RD COLUMN USING \( \chi \)

Each type of stage, condenser, reactive stage, feed stage, and reboiler, has been modeled by means of a process definition. The stages are connected with their incoming and outgoing liquid and vapor flows. This interaction has been modeled using shared variables \( V_i, V_o, L_i, \) and \( L_o \). Furthermore, when applicable, the process definitions are parameterized with the parameters as defined in Section 2.

Process definition Condenser models the mass balance of the condenser as described by (1).

\[
\text{proc Condenser}(\text{cont} = \text{x}_A, \text{x}_B, \text{x}_C) : \text{real} = \\
\quad \{ \begin{align*}
& \quad \text{eqn dot} \text{x}_A = (V_A - L-oA - D-xA) / M \\
& \quad \text{dot} \text{x}_B = (L_B - V_B - L-oB + B-xB) / M \\
& \quad \text{dot} \text{x}_C = (V_C - L-oC - D-xC) / M \\
& \quad L-oA = L-xA, L-oB = L-xB, L-oC = L-xC \\
\end{align*} \}
\]

Process definition ReactiveStage models (2), (5), and (7). Local (algebraic) variables are used to model liquid composition \( y_j \) and reaction rate \( r_j \) for each component \( j \in \{A, B, C\} \).

\[
\text{proc ReactiveStage}(\text{cont} = \text{x}_A, \text{x}_B, \text{x}_C) : \text{real} = \\
\quad \{ \begin{align*}
& \quad \text{eqn M dot} \text{x}_A = L-xA + V_A - L-oA + r_A \\
& \quad \text{M dot} \text{x}_B = L-B - V_B - L-oB + r_B \\
& \quad \text{M dot} \text{x}_C = L-C - V_C - L-oC + r_C \\
\end{align*} \}
\]

Process definition Reboiler contains (4) and (5). Again, local (algebraic) variables are used to model liquid composition \( y_j \) for each component \( j \in \{A, B, C\} \).

\[
\text{proc Reboiler}(\text{cont} = \text{x}_A, \text{x}_B, \text{x}_C) : \text{real} = \\
\quad \{ \begin{align*}
& \quad \text{eqn M dot} \text{x}_A = L-A - V_A - B-xA \\
& \quad \text{M dot} \text{x}_B = L-B - V_B - B-xB \\
& \quad \text{M dot} \text{x}_C = L-C - V_C - B-xC \\
& \quad y_A = a-xA / (a-xA + a-xB + a-xC) \\
& \quad y_B = a-xB / (a-xA + a-xB + a-xC) \\
& \quad y_C = a-xC / (a-xA + a-xB + a-xC) \\
& \quad V_A = V-A, V_B = V-B, V_C = V-C \\
\end{align*} \}
\]

Process definition PIController controls the RD column. It attempts to minimize error \( e \) by adjusting feed flow \( F \). Error \( e \) is defined as \( x_Cd - x_C \), where \( x_Cd \) denotes the desired composition, determined by the recipe, and \( x_C \) denotes the actual composition. Control signal \( Yc \) is defined as the summation of proportional part \( Kp \cdot e \) and integrator part \( Yi \). Since the manipulated variable (flow \( F \) ) is bounded, an anti-windup has been modeled in integrator part \( (Ki \cdot Ka \cdot Yc - F) \).

\[
\text{proc PIController}(\text{var} = \text{x}_C, \text{real}, \text{cont} = \text{x}_C, \text{real}, \text{alg} = \text{F} : \text{real} = \\
\quad \{ \begin{align*}
& \quad \text{val} \text{FMin}, \text{FMax}, \text{Kp}, \text{Ki}, \text{Ka} : \text{real} \quad \text{cont} = \text{Yi} : \text{real}, \text{alg} = \text{Yc}, \text{e} : \text{real} \\
\end{align*} \}
\]

317
5. SIMULATION RESULTS

In this section, simulation results are presented that are obtained using the \( \chi \) simulator. Figure 2 shows the open loop simulation results for a given feed \( F = 55 + 35 \cdot \sin(2 \pi \cdot \text{time}) \). The concentration \( xc5 \) at the outlet represents the product quality. It can be observed that the response is nonlinear. Since it is not the goal of this paper to derive a controller for this system, we restrict the feed to \( F \in [40, 90] \) to resemble a linear system for which a PI controller can be derived more easily. In future work, we deal with nonlinearities.

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{RD_openloop.png}
\caption{Openloop simulation results.}
\end{figure}

Fig. 2. Openloop simulation results.

In Figure 3, simulation results are shown of processing the recipes as defined in the \texttt{Recipe} process definition. The transition times between the several recipes (with different desired values \( xcD \) for \( xc5 \) should be minimal. For this reason a PI controller was implemented, in which concentration \( xc5 \) is controlled by \( F \). Note that the parameters of the PI controller (\( kP, ki, kA \)) are not tuned to obtain ‘optimal’ results.

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{RD_with_PIControl.png}
\caption{Controller profile for \( F \), response for \( xc5 \), and \( xcD \) is the desired composition.}
\end{figure}

6. CONCLUSIONS AND FUTURE WORK

In this paper, a dynamic formulation of a reactive distillation column has been defined. It has been used in a control system...
to obtain a profile for the feed that switches the product quality from one steady state to another (the product transition).

For modeling the RD column, as well as modeling its environment (which is future work), a formalism is required that includes high level language elements for continuous time modeling as well as high level language elements for discrete-event modeling. Process algebra $\chi$ is such a hybrid formalism. It has formal semantics and a relatively straightforward elegant syntax that is very intuitive and highly suited to modeling. In this paper, the language $\chi$ has been used to model the dynamic model of RD column controlled by a PI feed controller with anti-windup. The $\chi$ simulator has been used to obtain the open-loop responses as well as simulation results of the RD column controlled by the PI controller that tracks the changes in product quality (product transition) by adaptation of the feed flow.

Currently, we are extending the model with heat balances, a more realistic description of the vapor-liquid compositions, and changing from a three component reactive system to a four component system ($A + B \rightarrow C + D$), resembling an esterification reaction. Ultimately we intend to extend the model to a poly-esterification reaction including all side reactions. For a poly-esterification reaction the product composition $x_C$ determines the average chain length, often represented and measured inline as an acid-value.

Furthermore, the next release of the language $\chi$ will contain vector and matrix data types. This will improve the models significantly.

REFERENCES


Modeling, Simulation and Dynamic Identification of the Electro Hydraulic Proportional Valve

Bogdan Mihalescu
Constantin Călineu
Nicolae Vasiliu

Polytechnic University of Bucharest
Spl. Independenței 313
Bucharest, Romania
E-mail: {bogdan|calineu|vasiliu}@fluid-power.pub.ro

Abstract - The paper presents a new method for experimental identification of the dynamic characteristics of the proportional electro-hydraulic servo valve which contains a position transducer attached to the spool. The new method can replace the traditional method of measuring the input signal and the flow of the valve by considering the displacement of the valve’s spool as input for computing the Bode diagram. It combines the analytical model of the servo valve with the process of experimental identification in order to fully automate the procedure. The method was validated by direct measurement of the spool position for a chirp input signal. The experimental results are found in good agreement with the numerical simulation performed by AMESIM software.

INTRODUCTION
The modeling and simulation of any electro hydraulic control systems needs the performance specification of all the electric, hydraulic, and hybrid devices. Any manufacturer is obliged to provide those specifications to a certain degree. While the supplied specifications are of great value in choosing a certain component for a hydraulic control system, they offer a rather poor estimate for the performance level of the entire system. When a performance estimation of the system is required the design engineer must perform a complex numerical simulation. In this case the supplied specification will not suffice and a complete mathematical model of the hydraulic components is needed. Such mathematical models can be estimated by theoretical means and the coefficient of the equations obtained through a process of system identification. Another path for simulation would be to just identify the transfer function for the modeled object and insert it into the simulation grid of the system.

This paper describes the implementation of a simplified method for obtaining the transfer function of the electro-hydraulic proportional / servo valves which contains a position transducer attached to the spool.

The methods used to carry the test necessary for the dynamic identification are in compliance with the international standards regarding the testing of the four-way proportional valves. The test bench has been certified by the National Accreditation Body (RENAR). The software used for controlling the test procedures is written in LABView and enables the full automatization of the procedures, from the generation of appropriate input signals to recording the signals needed for system identification and computing the Bode diagram.

PRELIMINARY SIMULATIONS
In order to have a better understanding of the phenomenon involved in testing the electro-hydraulic valves and servo valves we first conducted some simulations in AMESim. The simulation didn’t involve just the valve but the entire test stand.

Fig. 1 Bode diagram simulated test bench with hydraulic cylinder
The aim of the preliminary simulations was to evaluate different methods of determining hydraulic parameters for the valve and for computing the frequency response diagrams. The reference was set by a test bench configuration built as recommended in ISO 10770:2009 with a low inertia, low friction hydraulic cylinder and a speed transducer attached to the cylinder’s piston (see figure 1).

![Bode diagram simulated test bench with metering orifice](image)

**Fig. 2 Bode diagram simulated test bench with metering orifice**

The simulation grid is comprised of both hydraulic components and control and data analysis components. Subsequent graphs show that the results of the simulations are in good agreement with the producer supplied specifications. This in turn validates the methods for investigating the effects of altering the test bench configuration. Figure 2 shows the same test bench model and the same valve but the flow metering device was changed from a hydraulic cylinder to a simple metering orifice and differential pressure transducers. The metering orifice was chosen according to figure 3 graph. The criteria for choosing the orifice diameter were the rated flow of the valve and the desired pressure drop across the orifice, as given by 1.

\[
Q = c_d \cdot A_o \cdot \sqrt{\frac{2}{\rho} (p_1 - p_2)}
\]  

(1)

Results from the preliminary simulations are given both as time graph, showing in parallel the input and output variables for both test stand configurations, and as frequency response graph showing the attenuation and phase lag of the output signals as frequency increases.

![Diagram for orifice diameter in regard to flow and desired pressure drop](image)

**Fig. 3 Diagram for orifice diameter in regard to flow and desired pressure drop**

**SIMULATION RESULTS**

Results from the preliminary simulations are given both as time graph, showing in parallel the input and output variables for both test stand configurations, and as frequency response graph showing the attenuation and phase lag of the output signals as frequency increases.

![Bode diagram obtained with hydraulic cylinder by simulation: a) magnitude versus frequency; b) phase lag versus frequency](image)

**Fig. 4 Bode diagram obtained with hydraulic cylinder by simulation: a) magnitude versus frequency; b) phase lag versus frequency**
As it can be observed by comparing figures 4 and 5 the two methods considered for measuring the flow of the valve and with that information determining the frequency response of the valve give comparable results. Thus it is possible to say that the simulation proved that as long as the testing conditions are strictly controlled and the measuring is done correctly any significant variable can be considered the output of the system.

We proceed to establish a more elaborate AMESim model for an electro-hydraulic proportional valve. This model has a position transducer attached to the spool. And since the flow of a proportional valve is dependent on the opening of the spool-body orifice (2 where b-width of the flow orifice x-spool displacement, the proportional valve spool is considered to be zero-lap), the positional signal for the spool can be considered an output variable for computing the frequency response.

\[ Q = c_x \cdot b \cdot x \cdot \sqrt{\frac{2 \cdot \Delta p}{\rho}} \]  (2)

Figure 6 shows the simulation network with a fully simulated valve spool a simplified electrical drive and a basic hydraulic power source set as prescribed in the testing standard. The frequency response diagram is presented in figure 7.
EXPERIMENTAL RESULTS

The experimentations were conducted using LABView software developed by the authors with the goal of automating the test procedures for obtaining the Bode diagram and for dynamic identification. To this end it was necessary to design both a data generation program (figure 8) and a data acquisition program. Furthermore the programs needed to be easily changeable and customizable so that they could be used for testing a wide range of hydraulic amplifiers.

The data generation program uses a standard method for generating a signal in LABView. It writes a waveform to the analog output buffer of a data acquisition board. Instead of choosing the waveform form a predefined set it actually computes the requested signal based on a mathematical formula supplied by the user. This in turn can induce delays but it was a necessary measure in order to obtain the desired level of flexibility. The delays were canceled out by carefully choosing the data to be processed in the main test program.

The results from the testing program are presented in figures 8 and 9. The program for system identification was also written in LABView. It uses a chirp signal in order to obtain the transfer function for the hydraulic amplifier under test. Validation was done by direct comparison of the identified model and the hydraulic amplifier responses in the time domain.

Figure 12 presents the step response for both identified models.
Fig. 10 Program for dynamic identification of servo-proportional valves showing the estimated transfer function for a first degree model.

Fig. 11 Dynamic identification of servo-proportional valves showing the estimated transfer function for second degree model.
CONCLUSIONS

The important gain of this research is the successful implementation of a test method for direct drive valves which uses a much simpler way to measure the flow of the valve and a high-tech computer system for “on the spot” data analysis.

The implemented method can be readily used for other types of 4 ways proportional hydraulic valves and furthermore the program developed for this paper is highly adaptable and new modules can be written and incorporated into it.

The modules used in the research program can form the basis for further researches into hydraulic system testing and identification.

The identified model are virtually identical as proven by the step response analysis, and anyone of them can be used to model the DFPplus proportional valve in a hydraulic system simulation.

BIBLIOGRAPHY


UNI-AXIAL COMPRESSION PREDICTION OF JET GROUTING COLUMNS USING SUPPORT VECTOR MACHINES

Joaquim Tinoco and António Gomes Correia
Centro do Território, Ambiente e Construção (C-TAC)
Escola de Engenharia, Universidade do Minho
Campus de Azurém, Guimarães,
Portugal
E-mail: jabitinocolage@civil.uminho.pt

Paulo Cortez
Centro Algoritmi, Departamento de Sistemas de Informação
Escola de Engenharia, Universidade do Minho
Campus de Azurém, Guimarães,
Portugal
E-mail: pcortez@dsi.uminho.pt

KEYWORDS
Soft soils, soil-cement mixtures, soil improvement, jet grouting, uniaxial compressive strength, regression, data mining, support vector machines, sensitivity analysis

ABSTRACT
Uniaxial compressive strength (UCS) is the mechanical properties currently used in geotechnical works design, namely in jet grouting (JG) treatments. However, when working with this soil improvement technology, due to its inherent geological complexity and high number of variables involved, such design is a hard, perhaps very hard task. To help in such task, a support vector machine (SVM), which is a data mining algorithm particularly adequate to explore high number of complex data, was trained to estimate UCS of JG samples extracted from real JG columns. In the present paper, the performance reached by SVM algorithm in UCS estimation is shown and discussed. Furthermore, the relation between mixture porosity and volumetric content of cement and the JG system were identified as key parameters by performing a 1-D sensitivity analysis. In addition, the effect and the interaction between the key variables in UCS estimation was measured and analyzed.

INTRODUCTION
Jet grouting (JG) technology is one of the most used soft-soil improvements methods (Falcao et al. 2000). According to JG technology, a high speed and pressure of grout (with or without other fluids) is injected into the subsoil, which cut and mixes the soil. At the end an improved mass of soil, often termed as Soilcrete is obtained. According to the number of fluids injected, three systems are conventionally in use: single, double and triple fluid system. Due to the heterogeneity of the soils, the constructive process of JG technology and nature of treatment fluid injected (normally water cement grout) there are many variables involved in treatment process (Nikkakhkan et al. 2010). Such conditions make the design of JG technology a complex geotechnical task. Nowadays, such design is almost performed based on empirical methods (Lee et al. 2005; Narendran et al. 1996), mainly in the initial project stages and in small scale geotechnical works where information is scarce. Therefore, and since these empirical methods are often too conservative and have a very limited applicability, the quality and the economy of the treatment can be compromised. Hence, and bearing in mind the high versatility of JG technology and its role in important geotechnical works, it is very important to develop rational models to estimate the effects of the different variables involved in JG process. On the other hand, in the last few years some powerful tool, incorporating advanced statistic analysis, has been developed and are able to automatically extract important rules from vast and complex data. Such tools, usually known as data mining (DM) techniques, has been successfully applied in several scientific areas namely in Civil Engineering domain (Lai and Serra 1994; Rezania and Javadi 2007). One of the most interesting DM algorithms is the Support Vector Machines (SVM), which was used in the present work and has the particularity to be applied in both classification and regression problems. SVM is especially useful to explore data with nonlinear relationships between several inputs and the target variable and has been successfully applied to solve geotechnical problems (Goh and Goh 2007; Tinoco et al. 2011b). The main criticism of “black box” DM techniques, such as SVM or artificial neural networks is the lack of explanatory power, i.e. the data-driven models are difficult to interpret by humans (Goh and Goh 2007). However, to overcome such drawback a sensitivity analysis (SA) procedure can be applied (Cortez and Embreehhs 2011). The performance reached by SVM algorithm trained with data collected directly from JG columns (JGS), with different JG parameters and soilcrete characteristics are shown and discussed in the present paper. Moreover, the key variables in UCS estimation are identified by applying a 1-D SA. Furthermore, the influence of the key variables in UCS estimation are quantified and discussed. In addition, and keep in mind a more realistic interpretation of the results a 2-D SA was performed to the first two key variables.

SUPPORT VECTOR MACHINES
Support Vector Machines are very specific class of algorithms, which is characterized by use of kernels, absence of local minima, sparseness of the solution and capacity control obtained by acting on the margin, or on number of support vectors. When compared with other types of base learners, such as the famous multilayer perceptron, SVM represents a significant enhancement in functionality. The supremacy of SVM lies in their use of non-linear kernel functions that implicitly map inputs into high dimensional feature spaces. In this feature spaces linear operations may be possible that may be separating hyperplane ($y_i = \omega_0 + \sum_{i=1}^{m} \omega_i \phi(x)$), related to a set of support vector points, in the feature space. Thus, although
SVMs are linear learning machines with respect to feature spaces, they are in effect non-linear in the original input space. These attractive features and promising empirical performance are responsible for its gain of popularity. SVM was initially proposed for classification problems by Vladimir Vapnik and his co-workers (Cortes and Vapnik 1995). Later, after the introduction of an alternative loss function proposed by Vapnik (Smola et al. 1996), called ε-insensitive loss function, was possible to apply SVM to a regression problems (Smola and Schölkopf 2004).

It is well known that SVM generalization performance (estimation accuracy) depends on a good setting of meta-parameters C, ε and the kernel parameters. The problem of optimal parameter selection is further complicated by the fact that SVM model complexity (and hence its generalization performance) depends on all three parameters. Parameter C controls the trade-off between complexity of the machine (Flatness) and the number of non-separable data points and may be viewed as a “regularization” parameter (Goh and Goh 2007). Parameter ε controls the width of the ε-insensitive zone, used to fit the training data. The value of ε can affect the number of support vectors used to construct the regression function. Hence, both C and ε values affect model complexity (but in a different way). Selecting a particular kernel type and kernel function parameters is usually based on application-domain knowledge and should reflect distribution of input (x) values of the training data. In the present work was adopted the popular Gaussian kernel, since it presents less parameters than other kernels (e.g. polynomial):

\[ K(x, x') = \exp(-\gamma \cdot \| x - x' \|^2), \gamma > 0 \]  

To reduce the search space, we adopt the heuristics proposed by Cherkassky and Ma (2004) to set the complexity penalty parameter, \( C=3 \) and the size of the insensitive tube \( \epsilon = \frac{\delta}{\sqrt{N}} \), where \( \delta = 1.5/N \times \sum_{i=1}^{N} (y_i - \bar{y})^2 \). \( \bar{y} \) is the value predicted by a 3-nearest neighbor algorithm and \( N \) the number of examples. The most important SVM parameter, the kernel parameter \( \gamma \), was set using a grid search within \( \{2^{-15}, 2^{-13}, ..., 2^2\} \), under an internal (i.e. applied over training data) 3-fold cross validation (Hastie et al. 2009).

All experiments were implemented in R tool (Team R 2009), using rmminer library (Cortez 2010), which is particularly suitable for SVM training. Before fitting the SVM model, the data attributes were standardized to a zero mean and one standard deviation and before analyzing the predictions, the outputs post-processed with the inverse transformation (Hastie et al. 2009).

**MODEL ASSESSMENT AND INTERPRETATION**

In regression problem, the main goal is to induce a model that minimizes an error measurement between observed and predicted values considering \( N \) examples. For this purpose three common metrics were calculated (Tinoco et al. 2011a): Mean Absolute Deviation (MAD), Root Mean Squared Error (RMSE) and Coefficient of Correlation (\( R^2 \)). The first two metrics should present lower values and \( R^2 \) should be close to the unit value. The regression error characteristic (REC) curve, which plots the error tolerance on the x-axis versus the percentage of points predicted within the tolerance on the y-axis (Bi and Bennett 2003), was also adopted during the analysis of the model performance.

To measure the generalization performance of the trained model were performed \( R \) runs under a cross validation approach. Under this scheme, the data are divided into \( k \) different subsets, being one used to test the model and the remains to fit it. At the end all data are used for training and testing. Yet, this method requires approximately \( k \) times more computation, because \( k \) models must be fitted. The final generalization estimate is evaluated by computing the MAD, RMSE and \( R^2 \) metrics for all \( N \) test samples.

Besides to the performance reached by a DM model it should be also possible to extract human understandable knowledge from the data. To do it a SA procedure (Cortez and Embrechts 2011) was applied. This procedure (1-D SA), which is applied after the training phase and analyzes the model responses when a given input is changed, allowing to quantify the relative importance of each variable. Such quantification is determined by successively holding all inputs at their average values, except one input attribute that is varied through its range of values \( (x_a \in \{x_1, ..., x_L\}) \), with \( j \in \{1, ..., L\} \) levels. The obtained responses \( \hat{y}_{aj} \) are stored and if there is a high gradient (\( S_j \)) observed, then this denotes a high input relevance \( (R_{a}) \), which is calculated by:

\[ R_a = \frac{S_a}{\sum_{j=1}^{L}S_j \times 100(\%)} \]  

For more input influence details, the Variable Effect Characteristic (VEC) curve was plotted. For a given input variable, the VEC curve plots the attribute \( L \) level values (x-axis) versus the sensitivity analysis responses (y-axis). Furthermore, aiming to achieve a more realistic interpretation of the models a 2-D SA was performed. Here, two variables are changed simultaneously and the response is measured. With the stored values it is possible to plot the VEC surface or VEC contour (see Cortez and Embrechts (2011)).

**JET GROUTING DATA**

To train and test SVM algorithm was used a dataset composed by 288 records. The tested samples were collected from different columns constructed under the same soil type, at different times and were kept inside a box until be tested in order to keep the water content and not be damaged. UCS (in MPa) was measured in unconfined compression tests with on sample strain instrumentation (Correia et al. 2009). The input variables were chosen based on the expert knowledge about soil-cement mixtures (Shibazaki 2004) and with the experience reached by authors in laboratory formulations studies (Tinoco et al. 2011a). Thus, the following set of eight input variables were chosen: relation between the mixture porosity and the volumetric content of cement \( (n/C_{c})^{2} \); age of the mixture (\( t \), days); JG method \( (JG_{M}) \); inverse of dry density of the soil-cement mixture \( (1/p_{d}) \) \( m^{3}/kg \); void ratio of the mixture \( (e) \); cement content \((\%)\); water content \((\%)\) and water/cement ratio \((W/C)\). The main statistics of both input and output variables are shown in Table 1. The soil under treatment was classified as lean clay (CL), with 39% of sand, 33% of silt, 27% of clay and 8.3% of organic matter. All columns were constructed with cement type CEM I 42.5 R.
Table 1: Summary of the input and output variables in UCS prediction of JGS

<table>
<thead>
<tr>
<th>Variable</th>
<th>Min</th>
<th>Max</th>
<th>$\bar{u}$</th>
<th>$\sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n/(C_{out})^4$</td>
<td>37.88</td>
<td>78.61</td>
<td>58.10</td>
<td>7.38</td>
</tr>
<tr>
<td>$t$</td>
<td>9.00</td>
<td>181.00</td>
<td>41.11</td>
<td>39.11</td>
</tr>
<tr>
<td>JGM</td>
<td>1.00</td>
<td>3.00</td>
<td>2.04</td>
<td>0.49</td>
</tr>
<tr>
<td>$f_{ps}$</td>
<td>5.63E4</td>
<td>1.40E3</td>
<td>8.20E4</td>
<td>1.21E4</td>
</tr>
<tr>
<td>$e$</td>
<td>0.56</td>
<td>2.85</td>
<td>1.25</td>
<td>0.33</td>
</tr>
<tr>
<td>$%C$</td>
<td>0.14</td>
<td>0.28</td>
<td>0.22</td>
<td>0.04</td>
</tr>
<tr>
<td>$\omega$</td>
<td>2.50</td>
<td>96.80</td>
<td>36.88</td>
<td>12.98</td>
</tr>
<tr>
<td>$W/C$</td>
<td>0.83</td>
<td>1.00</td>
<td>0.89</td>
<td>0.06</td>
</tr>
<tr>
<td>UCS</td>
<td>0.32</td>
<td>20.27</td>
<td>3.33</td>
<td>3.07</td>
</tr>
</tbody>
</table>

$\bar{u}$ – mean; $\sigma$ – standard deviation

RESULTS AND DISCUSSION

On figure 1, we compare the UCS of JGS measured with those predicted by SVM model for all 20 runs performed. In addition, it is also identified the areas for a prediction with an absolute deviation of 20%, 40% and 60%. As we can see, the accuracy reached by SVM model is relatively lower. Indeed, $R^2$ value is relatively worse (0.63±0.01) and the values for MAD and RMSE metrics are 1.29±0.01 MPa and 1.87±0.02 MPa respectively. However, observing the REC curve on figure 2, which shows the accuracy obtained for a given absolute deviation (in percentage), we can see a fast improvement on model accuracy. For example, to guaranty that the model is able to predict successfully 80% of the examples an error of 60% should be tolerated.

![Figures 1: Relation between UCS JG samples observed versus predicted by SVM model](image1.png)

![Figures 2: REC curve for SVM model](image2.png)

When evaluating a DM model, we should consider not only predictive accuracy but also model interpretability. In this paper, such interpretability is based on measuring which are the key input variables and how these affect the predicted output. To do so a 1-D and 2-D SA were applied.

Figure 3 give us an idea of the relative importance of each variable after applies a 1-D SA, with the correspondent t-

Based also on 1-D SA, the effect of each variable on UCS prediction was quantified. The VEC curves of the four key variables previously identified are shown in figure 4. The age of the mixture and the cement content has a positive impact in UCS prediction. However, its effect in UCS is different. The VEC curve of $t$ shows a convex shape that means that UCS increases quickly in the early ages and after that tend to stabilize (typical behavior of cement mixtures). In the other hand, VEC curve of $%C$ is almost linear. The remains two key variables have a similar effect on UCS prediction. UCS of JG samples decrease according to an exponential shape if $n/(C_{out})^4$ increase, i.e., increasing mixture porosity or decreasing cement content. We also can observe that the UCS decrease almost linearly with the JG method. That means that the biggest strength is reached by application of single fluid system.

All observation previously exposed based on a 1-D SA, such as the relative importance of each variable or VEC curves are very usefully to understand the behavior of JG mixtures. However, in this kind of analysis all variables are fixed in its mean value except one that is ranged from its minimum to maximum values. In real works this normally never happen.
Thus, in order to do a more realistic analysis we carried out a 2-D SA. In the next lines, the results of a 2-D SA for \( n/(C_{o})^{d} \) and \( JGM \) (the first two key variables) are exposed and discussed.

![Diagram](attachment:image1.png)

**Figures 3:** Relative importance of each variable according SVM model, quantified by a 1-D SA

![Diagram](attachment:image2.png)

**Figures 4:** VEC curves for the more relevant variables, according to SVM model measured by 1-D SA

In figure 5 we can see that \( JGM \) is the variable with the biggest interaction with \( n/(C_{o})^{d} \) in UCS prediction. Plotting of UCS prediction by SVM model when these two variables are changed simultaneously (see Figure 6), keeping the remains at their means values it is possible to observe that the highest strength is reached when single fluid system is applied and is produced a soil cement mixture with lower values of \( n/(C_{o})^{d} \). These results coincide with those obtained from the interpretation of the VEC curves for these two variables.

![Diagram](attachment:image3.png)

**Figures 5:** 2-D interaction with \( n/(C_{o})^{d} \) variable in UCS prediction according to SVM model, quantified by a 2-D SA

![Diagram](attachment:image4.png)

**Figures 6:** VEC surface for \( n/(C_{o})^{d} \) and \( JGM \) in UCS prediction, according to SVM model measured by 2-D SA

When a similar procedure was carried out with \( JGM \) (second most relevant input variable), was possible observe that the age of the mixture is the variable with the strongest interaction with \( JGM \) (18%). In addition, it was possible to see that the highest values of UCS are reached when single fluid system is applied and \( t \) is high. Furthermore, it is possible observe that for double and mainly for triple fluid system UCS increase slightly with the age of the mixture.

**FINAL REMARKS AND CONCLUSIONS**

In the present study support vector machines (SVM) were used to explore jet grouting (JG) data, collected directly from JG columns (JGS), in order to predict its uniaxial compressive strength (UCS). Although SVM model has experienced some difficulties to accurately estimate UCS of JGS over time, some important conclusions can be drawn. By performing a 1-D sensitivity analysis (SA), we have shown that the relation between the
mixture porosity and the volumetric content of cement (n/C\(_{60}\)) JG method (JGM), cement content (\%C) and age of the mixture (t), play an important role in UCS estimation over time. As expected, \%C and t has a positive impact in UCS prediction, underlining the exponential relationship of the latter with UCS. In the other hand, n/C\(_{60}\) and JGM have a negative impact in UCS estimation. It is appealing to observe that the JGM effect is approximately linear. Performing a 2-D-SA, it was observed that the highest UCS values are reached for single fluid system and for lower values of n/C\(_{60}\). In addition, UCS of JGS increases exponentially with t reaching the highest values for single fluid system. Moreover, when triple fluid system is applied UCS just slightly increases over time.

The knowledge obtained from the present study is a great contribution to understanding better the behavior of JG mixtures. As a result of this knowledge, the quality and the cost of JG treatment can be improved by controlling better parameters involved in JG process. As a future works, an attempt to improve the model predictive capability as well as its applicability will be done. In addition, SVM and other data mining algorithm will be applied to define predictive models of JG columns diameters as well as its stiffness.

ACKNOWLEDGEMENTS

The authors wish to thank to “Fundação para a Ciência e a Tecnologia” (FCT) for the financial support under the strategic project PEst-OE/ECI/UI4047/2011 and the doctoral Grant SFRH/BD/45781/2008. Also, the authors would like to thank the interest and financial support by Tecnasol-FGE.

REFERENCES


BIOGRAPHIES

JOAQUIM TINOCO was born in Braga, Portugal and went to the University of Minho, where he studied civil engineering and obtained his degree in 2008 (5-years graduation). Currently, he is developing the PhD works at University of Minho on “Application of Data Mining Techniques to Jet Grouting Columns Design”. His e-mail address is jatinoco@civil.uminho.pt.

ANTÓNIO GOMES CORREIA is full Professor at University of Minho and director of Territory, Environment and Construction Research Centre (C-TAC). His E-mail address is ace@civil.uminho.pt.

PAULO CORTEZ is Associate Professor at the Department of Information Systems of University of Minho and researcher at centro Algoritmi. His e-mail address is pcoratez@dsi.uminho.pt and his Web page can be found at http://www3.dsi.uminho.pt/pcortez.
WASTEWATER SIMULATION
COMPARATIVE MODELING OF HYDRAULIC LOAD OF WATER AND WASTEWATER NETWORKS

Jan Studzinski
Polish Academy of Sciences, Systems Research Institute
Newelska 6, 01-447 Warsaw, Poland
E-mail: studzinski@ibspan.waw.pl

KEYWORDS: Mathematical modeling, time series methods, neural nets, fuzzy sets models, water and wastewater networks.

ABSTRACT

In communal waterworks the water nets and wastewater nets create an uniform system of successive connected objects in which the work of the first one affects the work of another one. The key parameters for both objects are their hydraulic loads which have to be known while modeling the networks and the forecasted load values are needed while developing the algorithms for the predictive networks control. In the paper different algorithms to model the network loads of the communal water and wastewater nets are presented and compared regarding their computing velocity and accuracy. The computations concerned have been done with the real data received from the Polish waterworks in Rzeszow.

INTRODUCTION

The communal waterworks are responsible by means of water and wastewater nets for the production and distribution of drink water and for the sewage detention. In order to realize these goals some proper tools must be available that will ensure optimal operation of these networks, i.e. the operation which is energy saving, failure free and in accordance with the demanded operational settings concerning pressures and flows of water and wastewater. Such tools are the IT systems for the management of water and wastewater nets which are developed at the Systems Research Institute (IBS PAN) since a couple of years. In these systems the key programs are mathematical models of the networks and of the processes occurring there, amongs them the models to forecast some different states of the networks work. Using the trustworthy load forecasts one can improve the operational control of the networks by developing the applicable scenarios of pumps operation in the pump stations and of filling and emptying the storage reservoirs located on the water net. To model the load forecasts of the water and wastewater nets the time series algorithms, the neural nets and the fuzzy sets have been used.

THE COMPUTER PROGRAMS AND MEASUREMENT DATA USED

The programs used to develop the models are IDOL software developed at IBS PAN in case of the time series models (Nahorski & Studzinski, 1988), STATISTICA software developed by StatSoft (www.statsoft.pl/textbook) in case of neural nets and AMandD software (Wnuk, 2004) developed at the University of Technology Warsaw in case of fuzzy sets models. While modeling with the time series methods three calculation algorithms of Kalman (K), Clarke (Cl) and of the maximal likelihood (ML) have been used. While modeling with neural nets the MLP (Multi Layer Perceptron) networks with one hidden layer and in the case of fuzzy nets the TSK (Takagi, Sugeno, Kanga) models have been tested. The data to determine the models came form the waterworks in Rzeszow. By the water net modeling they were the daily water production values (WP) from the time period of 1.242 days and by the wastewater net modeling the values of the raw sewage inflow to the sewage treatment plant (SI), the rainfalls data for Rzeszow (RF) and the values of the water level in the river flowing through the city (WL) have been additionally considered. While modeling the water and wastewater nets the one-day-forecasts of the network loads have been determined. To evaluate the models the following criteria have been used (Bartkiewicz & Studzinski, 2010):

- standard deviation of the model error ($\sigma_\hat{y}$):

$$\sigma_\hat{y} = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (\hat{y}_t - \overline{\hat{y}})^2}$$

with $\overline{\hat{y}} = \frac{1}{N} \sum_{t=1}^{N} \hat{y}_t$

- standard deviations quotient ($\zeta$):

$$\zeta = \frac{\sigma_\hat{y}}{\sigma_y}$$

with $\sigma_y = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (y_t - \overline{y})^2}$ and $\overline{y} = \frac{1}{N} \sum_{t=1}^{N} y_t$

- Mean Squared Error (MSE):

$$MSE = \frac{1}{N} \sum (y_t - \hat{y}_t)^2$$

- Pearson correlation criterion (R):

$$R = \frac{\frac{N}{\sum_{t=1}^{N} (y_t - \overline{y})(\hat{y}_t - \overline{\hat{y}})} }{\sqrt{\frac{N}{\sum_{t=1}^{N} (y_t - \overline{y})^2}} \cdot \sqrt{\frac{N}{\sum_{t=1}^{N} (\hat{y}_t - \overline{\hat{y}})^2}}}$$

- Model deviation coefficient $\phi^2$

$$\phi^2 = \frac{\frac{N}{\sum_{t=1}^{N} (y_t - \overline{y})^2}}{\frac{N}{\sum_{n=1}^{N} (y_t - \overline{y})^2}}$$

- Acaike Information Criterion (AIC):

$$AIC = 2,83788771 + \ln(MSE) \cdot N + 2L_p$$

1 The paper describes the results of the research project of the Polish National Center of Science No. N N519 6521 40.
where $y_t$ and $\tilde{y}_t$ mean the measurements data and the model output at the time $t$, $N$ is the number of data in the data series and $L_p$ is the number of the model parameters. To evaluate the models also the results of their simulation runs have been used which have been done using separated data series. In case of neural nets the initial data set was divided into three subsets for learning (L), validating (V) and testing (T) runs and for the simulation the later two subsets have been used. In case of the time series and fuzzy sets modeling the initial data set was divided into two subsets (for learning and testing runs) and the second one was used for the simulation.

**MODELING WITH THE TIME SERIES METHODS**

While modeling with the time series methods the following general model description is used (Nahorski & Studzinski, 1988):

\[
y_n = -A(z^{-1})y_n - \sum_{i=1}^{M} B(z^{-1})x_{in} + v_n
\]

\[
\tilde{y}_n = \hat{A}(z^{-1})y_n - \sum_{i=1}^{M} \hat{B}(z^{-1})x_{in}
\]

with $n = 1,2,\ldots,N$, $N$ - number of measurements data, $M$ - number of model inputs, $A(z^{-1}), B(z^{-1})$ - difference operators for the output $y_n$ and the inputs $x_{in}$ of the process, $\hat{A}(z^{-1}), \hat{B}(z^{-1})$ - difference operators for the model signals, and $v_n$ - the correlated noise. These equations are used directly in the Kalman algorithm while in the algorithms of Clarke and of the maximal likelihood the noise $v_n$ is additionally modeled with the following difference operators, respectively:

\[
(1 + D(z^{-1}))v_n = \varepsilon_n
\]

\[
v_n = (1 + D(z^{-1}))\varepsilon_n
\]

In the case of load modeling for the water net the autoregressive time series models with only one output being the water production (WP) and without any inputs, i.e. without the difference operators $A(z^{-1}), B(z^{-1})$ in (1) and (2), have been investigated. While calculating the time series models their evaluation can be done not only with the use of the above formulated synthetic criteria concerning the whole models but also with the use of standard deviations of their individual parameters. This makes possible to eliminate from the model operators the parameters whose standard deviations are essentially less than the parameter values.

The results obtained for the water net load modeling are shown in Tables 1 and 2 and in Figure 1. As the best model turned out the Kalman model of seventh order with four parameters $a_1, a_2, a_3$, and $a_4$; for which the MSE values for the learning (i.e. modeling) and testing (i.e. simulation) runs are smallest and the respective correlation values $R$ are biggest.

In case of load modeling for the wastewater net the time series models with one output being the raw sewage inflow to the sewage treatment plant and with up to three inputs being the water production for the water net, the rainfalls data for the city and the water level in the river flowing through the city have been tested (Bartkiewicz, 2010).

**Table 1.** Evaluation results for the best time series models of the wastewater net load.

<table>
<thead>
<tr>
<th>Model</th>
<th>$\sigma_2$</th>
<th>$\xi$</th>
<th>$\sqrt{MSE_{L}}$</th>
<th>$R_L$</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>K/6/WP-WL-RF</td>
<td>3164</td>
<td>0.47</td>
<td>100.1</td>
<td>0.87</td>
<td>18,518</td>
</tr>
<tr>
<td>CI/6/WP-WL-RF</td>
<td>3164</td>
<td>0.49</td>
<td>100.1</td>
<td>0.87</td>
<td>18,530</td>
</tr>
<tr>
<td>ML/3/WP-WL-RF</td>
<td>3822</td>
<td>0.60</td>
<td>146.1</td>
<td>0.80</td>
<td>18,858</td>
</tr>
<tr>
<td>K/6/WP-WL</td>
<td>4081</td>
<td>0.64</td>
<td>166.5</td>
<td>0.77</td>
<td>19,000</td>
</tr>
<tr>
<td>CI/6/WP-WL</td>
<td>4081</td>
<td>0.64</td>
<td>166.5</td>
<td>0.77</td>
<td>19,002</td>
</tr>
<tr>
<td>ML/3/WP-WL-RF</td>
<td>4354</td>
<td>0.68</td>
<td>189.6</td>
<td>0.74</td>
<td>19,106</td>
</tr>
<tr>
<td>K/6/WP-RF</td>
<td>3320</td>
<td>0.52</td>
<td>110.2</td>
<td>0.85</td>
<td>18,598</td>
</tr>
<tr>
<td>CI/6/WP-RF</td>
<td>3320</td>
<td>0.52</td>
<td>110.2</td>
<td>0.85</td>
<td>18,600</td>
</tr>
<tr>
<td>ML/3/WP-RF</td>
<td>3902</td>
<td>0.61</td>
<td>152.3</td>
<td>0.79</td>
<td>18,893</td>
</tr>
<tr>
<td>K/6/WP</td>
<td>4272</td>
<td>0.67</td>
<td>182.5</td>
<td>0.74</td>
<td>19,075</td>
</tr>
<tr>
<td>K/6/RF</td>
<td>3624</td>
<td>0.57</td>
<td>131.3</td>
<td>0.83</td>
<td>18,754</td>
</tr>
<tr>
<td>K/6/WL</td>
<td>4333</td>
<td>0.68</td>
<td>187.8</td>
<td>0.75</td>
<td>19,103</td>
</tr>
</tbody>
</table>

**Figure 1.** Modeling results for the best Kalman model of the water net load.

**Table 2.** Evaluation results for the best time series models of the water net load.

<table>
<thead>
<tr>
<th>Model</th>
<th>K</th>
<th>CI</th>
<th>ML</th>
<th>MSE_L</th>
<th>MSE_T</th>
<th>R_f</th>
<th>MSE_L/R_f</th>
</tr>
</thead>
<tbody>
<tr>
<td>5322</td>
<td>0.91</td>
<td>5326</td>
<td>0.90</td>
<td>4480</td>
<td>0.73</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4298</td>
<td>0.92</td>
<td>4307</td>
<td>0.91</td>
<td>4345</td>
<td>0.82</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The results obtained for the wastewater net load modeling are shown in Table 3 and in Figure 2 (an exemplary designation K/6/WP in Table 3 means the Kalman model of sixth order with only 1 input WP). Also in this case as the best model turned out to be the Kalman model of sixth order and with all three inputs considered (WP, WL, RF), for which the values of its statistical criteria are in general better than these ones of other models.

**MODELING WITH NEURAL NETS**

At the calculations of the water net load with the MLP neural nets only 1 neuron was placed on the output layer being the daily water production in day n and on the input layer seven neurons were placed being the water production in the properly previous days n-1, n-2, ..., n-7. In this way in a neural model the measurements data from a whole week are simultaneously considered what corresponds to the adoption of a difference operator of seventh order in an autoregressive time series model (Bogdan & Studzinski, 2010).

The results of modeling with the neural nets of the water net load are shown in Table 4 and Figure 3 (an exemplary designation MLP/7/3/1 in Table 4 means the MLP model with 7, 3 and with 1 neuron on the input, hidden and output layer, respectively). As the best model turned out the neural net with the numbers of neurons 7-10-1 on the subsequent layers. Evaluating the models with respect to the values of their statistical criteria one can see that in principle all computed models are similar. Moreover the model that has been taken as the best one depicts the measurements relatively well concerning its fitting to the measurement values as well as its keeping out with the measurement changes (see Figure 3).

The best results of modeling of the wastewater net load are in Table 5 and Figure 4 presented (an exemplary designation MLP/4/3/12-5-1 in Table 5 means the MLP model with the time shifting of measurements in the input signals about 4 days, with 3 input signals WP, WL and RF and with the numbers of neurons 12-5-1 on the input, hidden and output layers of the network). There is to see there that Model MLP/4/3/12-5-1 with three inputs turned out to be the best considering the correlation values ($R$) for the learning, testing and validation runs and the value of $AIC$ criterion. In this model the measurements in the all input signals are shifted back round 4 days what corresponds to the adoption of difference operators of fourth order for each input in a time series model.

![Figure 2. Modeling results for the best Kalman model K/6/WP-WL-RF of the wastewater net load.](image)

![Figure 3. Modeling results for the best neural model MLP/7/10/1 of the wastewater net load.](image)

![Figure 4. The best neural model MLP/4/3/12-5-1 of the wastewater net load.](image)
MODELING WITH THE FUZZY SETS

The process of modeling with the fuzzy sets method consists of three steps: determining some affiliation functions for fuzzification of the input signals of the model investigated, fuzzification of the model inputs and de-fuzzification of the model output (Studzinski & Stachura, 2011). At first the value ranges of the model signals are divided into zones for which some proper affiliation functions are dedicated. The values range of these functions is between 0 and 1 and they are mostly in form of trapezoids or triangles. Then the signals concerned are properly transformed (fuzzified) using these functions and after that the output of the model is formed. In order to get a dynamic model a linear difference operator is applied as the output signal in which the transformed inputs are appearing as variables. A potential nonlinearity of the process modeled is emulated through the inputs fuzzification by means of the nonlinear affiliation functions. The evaluation of the model parameters included in the output equation occurs with a method of static optimization exactly the same as in the case of time series or neural models. While computing the model the data set used is divided into two subsets for making the modeling (learning) and simulation (testing) runs.

Figure 5. Affiliation functions for the fuzzification of the water production data and the components of the best fuzzy sets model for the water net load.

Table 6. The fuzzy sets model of the water net load.

<table>
<thead>
<tr>
<th>Model</th>
<th>( \varphi^2 \times 100% )</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSK Learning</td>
<td>7,17%</td>
</tr>
<tr>
<td>TSK Testing</td>
<td>7,94%</td>
</tr>
</tbody>
</table>

Figure 6. Testing results of the TSK model of the water net load.

Table 7. The fuzzy sets models of the wastewater net load.

<table>
<thead>
<tr>
<th>Model</th>
<th>( \sigma_y ) (mean value)</th>
<th>( \xi ) (mean value)</th>
<th>( \text{MSE} ) (mean value)</th>
<th>( R ) (mean value)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSK/WP-WL-RF</td>
<td>4290</td>
<td>4034</td>
<td>0,671</td>
<td>0,631</td>
</tr>
<tr>
<td>TSK/WP-WL</td>
<td>4006</td>
<td>5253</td>
<td>0,626</td>
<td>0,821</td>
</tr>
<tr>
<td>TSK/WP-RF</td>
<td>4784</td>
<td>5827</td>
<td>0,748</td>
<td>0,911</td>
</tr>
</tbody>
</table>

Figure 7. Calculation results of the best fuzzy-sets-Model TSK/WP-WL-RF for the learning (on top) and testing data.

The results obtained while modeling the water net load are shown in Figures 5 and 6 and in Table 6. The determined
TSK model is in form of the autoregressive difference operator of seventh order (Fig. 5) that is similar to the time series model of Kalman (Table 1). During the fuzzy sets modeling for each data zone a sub-model is calculated separately and the end model of the process is formed as a sum of all sub-models determined. The evaluation results of the model (Table 6 and Fig. 6) shows its usefulness for forecasting the water net load with the forecast horizon of 1 day.

While modeling the wastewater net load only the models with two and three inputs have been investigated. The models with only 1 input have displayed a bad fitting to the data which came from the calculation with the neural nets (Table 5). The models determined have got an autoregressive structure what makes them structurally similar to the time series and neural models. The results of modeling are shown in Table 7 and Figure 7. There is to see from there that the models reproduce on principle well the daily changes of the raw sewage inflow to the sewage treatment plant although the fitting of the models to the measurements data is generally worse as in case of the time series and neural models. As the best model turned out the model TSK/VP-WL-RF with three inputs whereas the water production WP with two trapezoid functions and the rain falls RF with three trapezoid functions were fuzzified. The third input (water level WL) as well as the model output (sewage inflow SI) have been not fuzzified. The best model obtained is in form of a difference equation with four difference operators of fourth order for all three inputs and for the output.

The results for the best models obtained while modeling the wastewater net load with different methods are shown in Table 8.

**Table 8.** Comparison of the best time series, neural and fuzzy sets models of the wastewater net load.

<table>
<thead>
<tr>
<th>Model</th>
<th>$\xi$</th>
<th>R</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>K/6/VP-WL-RF</td>
<td>0.47</td>
<td>0.87</td>
<td>18.52</td>
</tr>
<tr>
<td>MLP/4/12-5.1</td>
<td>0.63</td>
<td>0.83</td>
<td>17.50</td>
</tr>
<tr>
<td>TSK/VP-WL-RF</td>
<td>0.67</td>
<td>0.60</td>
<td>-</td>
</tr>
</tbody>
</table>

**CONCLUSIONS**

In the paper also the results of mathematical modeling of the wastewater net load are presented. The models have been either calculated with three different modeling methods, i.e. with the time series, neural nets and fuzzy sets. The methods seem to be very different but one can say that all the models are after all nothing else as the big systems of linear and nonlinear algebraic equations of differentiated complexity. At it the time series equations are the simplest description of models and the fuzzy sets ones are the most complicated. The results of modeling show that the simplest and fastest time series method of Kalman creates better, i.e. exact models of the wastewater net load than the other and more complicated times series methods. It was also the case by the modeling of the water net load. The Kalman method makes possible to determine better models than the much more complex methods of neural nets and fuzzy sets. The water net load models obtained are meant as the computer support to forecast the raw sewage inflow to the sewage treatment plant in the city Rzeszow what will facilitate the development of control algorithms for the object.

**REFERENCES**


http://www.statsoft.pl/textbook

Prof. Jan Studzinski, born in 1946 in Warsaw, educated at the Technical University of Warsaw at the Faculty for Electrical Engineering and at the Warsaw University at the Faculty for Mathematics, working at the Systems Research Institute of Polish Academy of Sciences, where he has obtained his PhD. and Sc.D. degrees and where he leads the Center for Applications of Informatics in Environmental Engineering, author of 3 books and more than 220 scientific papers, dealing for many years with mathematical modeling and computer simulation of complex dynamical systems, with development of optimization and control methods and of computer aided decision support systems for management of communal waterworks, awarded for his work with several prices at the Belgian and International Trade Fairs for Technological Innovation held in Brussels, Board chairman of the Foundation for Development of System Sciences of Polish Academy of Sciences.
Modeling and simulation of technological wastewater neutralization system in the brewing industry

Andrzej Urbanik
Institute of Computing Science
Poznań University of Technology
ul. Piotrowo 2, 60-965 Poznań, Poland
Andrzej.Urbaniak@cs.put.poznan.pl

Wojciech Góra
Institute of Environmental Engineering
Poznań University of Technology
ul. Piotrowo 3a, 60-965 Poznań, Poland
Wojciech.Gora@put.poznan.pl

Rafal Brodziak
Institute of Environmental Engineering
Poznań University of Technology
ul. Piotrowo 3a, 60-965 Poznań, Poland
Rafal.Brodziak@put.poznan.pl

KEY WORDS
environmental modelling, water treatment, simulation tools, data acquisition, brewing

ABSTRACT
The paper presents modeling and simulation of the system for averaging and neutralization of wastewater from the brewing industry by using waste carbon dioxide. The goal of this work is to prepare environment for optimization results of minimizing the use of CO₂ in the process of decreasing pH index of wastewater. It was obtained by relevant use of alkaline waste water tank volume and acidic waste water. Performed calculations were based on a mathematical model of a physical object in the MATLAB Simulink environment. To simulate variability of the flow model there was used an pH value of effluent obtained from the measuring and control equipment of the existing system. Also, functions for defining volume of water and pH in each tank were determined on the basis of experiments and general physico-chemical knowledge.

INTRODUCTION
Various technological processes cause different kinds of waste fluids. Usually, this wastewater is directed to the general sewer system. The diversity of pollutants requires individual primary treatment of industrial wastewater up to the accepted level for general sewer system [Dymaczewski and Sozański, 1997]. The accepted levels of particular pollutants are defined by an sewer system operator’s. Usually, running of many treatment processes is based on the set-points assumed in the project stage. During an operation many primary values change, thus the current data must be updated for an optimal treatment process’ conduction [Sozański and Urbanik, 2009]. In brewing plants the wastewater have high variability of pH and flow values while draining off. Furthermore, these effluents are characterized by high load of organic carbon and relatively small load of nutrients, such as nitrogen and phosphorus. The composition of the brewing wastewater helps to improve operation conditions at municipal wastewater treatment plants by improving the ratio of carbon, nitrogen and phosphorus on the biological treatment stage. Therefore the only magnitudes that require regulation and control of the effluent from the industrial plant are instantaneous flow volume and pH. Control of these two magnitudes allows correct operation of the sewer system. Because of the specific character of beer production processes wastewater discharged from the manufacturing plant is mostly alkaline. Furthermore, in the process of fermentation great amount of carbon dioxide is produced. Hence, in accordance with European emission standards for large industrial plants, named as BAT (Best Available Techniques) in the brewing industry, there is recommended the temporary retention averaging composition of treatment and suppressing extreme flows, followed by neutralization with the usage of waste carbon dioxide. Neutralization with carbon dioxide proceeds only from alkaline into acid reaction [Enyi et al. 2007]. In case of short term wastewater retention there is significant likelihood of exceeding the lower limit of pH value during acid wastewater inflow and the need for mineral alkali in order to raise the pH into the normative range [Góra and Jaszczyszyn, 2010].
Such circumstance would raise operating costs of the system, so it is advisable to split the neutral and alkaline wastewater volume from the acidic and use the latter for self-neutralization during alkaline wastewater inflow. Such solution allows to minimize operating costs and negative impacts on the environment thanks to limitation of types and amounts of reagents added to wastewater. Given the carbon dioxide, although it is a byproduct of beer production, is purified, dried and used at the time of bottling beer, and its residual part is sold. [Rao et al. 2007]
Therefore, despite the impossibility of an overdose of carbon dioxide (regardless of the dose, it can not be exceeded to lower normative pH values), it is advisable to carry out the monitoring of both treating wastewater pH value and conducting the neutralization process.
Thus, there is a need for at least two reservoirs before the main process of neutralization: one for its neutral and alkaline wastewater and another for its acidic wastewater tank. Neutralization with carbon dioxide can be done in an integrated circuit with self-neutralization tank or in independent reaction tank. This approach belongs to the class of integrated management and control system [Obozynska et al. 2000].
DESCRIPTION OF THE CONSTRUCTION AND OPERATION OF MODELLLED PLANT

The proposed structure of neutralization system is presented on Fig.1. Wastewater flows by gravity to the point of decision making (D1), which through the snapshot valves (EV1, EV2) is directed to the self-neutralization tank (SNT) or to the acidic wastewater tank (AWT). A decision is taken as a function of flow rate(Q) and pH of the influent wastewater and the status of AWT and SNT tanks. Depending on parameters of the influent wastewater (Q_in, pH_in) and state of reservoirs (V, pH – where V is actual wastewater volume in a tank) there is also provided a periodic acid-water pumping from AWT to SNT tank via pump 2 according to decision D2. The right redirection of influent wastewater and regulation of the pump P2 should enable optimum conditions for the neutralization with carbon dioxide(DNT), i.e. the pH value at the outlet from the SNT tank should be as close as possible to the upper limit of the normative pH range and volume of sewage outflow rendering appropriate reaction time. For pumping wastewater with pH normative value it is routed directly to the sewer system (OUTFLOW1). If pH values are outside the range of normatives, wastewater stream is directed to be neutralized with carbon dioxide and then to be drained (OUTFLOW2) – decision D3.

SIMULATION OF THE pH VARIABILITY

In order to perform computer simulation, described by differential equations, the mathematical model of the object was created. The model is based on theoretical knowledge of the object and general physical laws. Simulation of industrial wastewater system was based on input data obtained from studies carried on a real object. Input variables in the system are the pH_in and the flow Q_in. In the computer implementation of the model it is particularly useful that the software modules can build models in the form of block diagrams. This kind of models, where individual blocks represent and are referred to mathematical operations, calculating the response based on extortions, etc. help reading and take into account dependencies in the mathematical model. The big advantage of it is a visible structure and direction of impact of each of the signals. The model was implemented in Simulink software package that is a part of the numerical MathWorks MATLAB.

For the simulation, the implemented model includes three cases for which the actual pH value and volume of the tank are calculated:
- inflow Q_in> 0, outflow Q_out> 0
- inflow Q_in = 0, outflow Q_out> 0
- inflow Q_in> 0, outflow Q_out = 0

To calculate the real time pH value in the tank, there was an algorithm created for calculating pH_in to the A variable as Alcalinity according to experimental set values:

- pH>8.3
  \[ A = 7E-12pH^{1.624} \]
- pH<=7 and pH<=8.3
  \[ A = 26,705\ln(pH) - 56,511 \]
- pH>=2.24 and pH<=7
  \[ A = 2.8371pH-23.94 \]
- pH>0 and pH<2.24
  \[ A = 56.987pH-142.17 \]

The block diagram of this algorithm is presented on Figure 2.

Figure 1. Block scheme of wastewater neutralization system

In order to obtain the proper model process and variability of tanks states it is necessary to know the dependence of pH values changeability on the influent magnitudes. The capability of an influent to change pH is determined by acidity and alkalinity [val/m3], which may vary depending on the type of treatment for the same pH values. Therefore, the experimentally determined average values of alkalinity and acidity of wastewater for a certain pH within a pH range of 1 to 14 and by means of mathematical equations describe the dependence of acidity or alkalinity on pH for the four pH ranges.

There was numerical quantity introduced defined as "the ability of neutralization" in following forms:
for pH <8.3 is equal to the negative value of acidity, for pH> 8.3 is equal to the alkalinity.
for a pH of 8.3 " the ability of neutralization " is "0". This solution allows direct conversion of the pH in wastewater current value upstream to be changed in the tank.

Figure 2: The block diagram of pH to alkalinity algorithm

Based on the designated value of input "alkalinity" there is calculated actual value of alkalinity for every tank. Finally, next step is based on that determined pH, opposite to the first step using inverse functions A. The block diagram presented below on Fig. 3 is the system that calculates actual pH value and the amount of liquid in a tank where the pH value is the main part of modeling for wastewater neutralization system.
ability for self-neutralization and eventually the increased usage of CO₂.

In the paper, there is suggested the technological solution for treatment of wastewater from brewery industry. The proposed approach is characterized by following features:

1. For achieving satisfactory effects there is applied the simulation model of process' control. The simulation tools allow to introduce nonlinear dependences of alkalinity or acidity from pH index and flow of wastewater.

2. The proposed solution needs current measure of two main magnitudes: pH index and inflows in particular points of the process.

3. The obtained simulation results present the essential changes of process variables and justify assumed approach.

4. The proposed solution belongs to class of the integrated management and control system applied for industrial wastewater treatment plant.

REFERENCES

Góra W., Jaszczyszyn K.; Brewery wastewater neutralization with waste carbon dioxide application; Water Supply and Water Quality – present issues; Collective work edited by M. Sozański, 2010, PZITŚ , Poznań

FINAL REMARKS

Created model allows to determine the status of tanks (the pH value and volume of water) as a function of inflow with separation of acidic wastewater and others. Such a model is the basis for optimization of actions intended to minimize the neutralizing reagent use - carbon dioxide. To achieve this the outflow to the reactor pH should be near 9, which is the upper limit of acceptable scope. Draining wastewater with higher pH index results with increasing demand for CO₂ and on the other hand, draining wastewater with pH lower than 9 (including pH with normative range: 6.5-9) causes the wested
DISASTER MANAGEMENT
MOMENTUM PROBABILITY AND STAFFING CALCULATIONS FOR COMET, METEOR AND ASTEROID DEFENSE

By
William Conley
Austin E. Cofrin School of Business
480B Wood Hall
University of Wisconsin at Green Bay
Green Bay, Wisconsin 54311-7001 U.S.A.
conleyw@uwgb.edu

KEYWORDS
Aerospace, asteroid defense, management, Multi Stage Monte Carlo Optimization Simulation, disaster prevention

ABSTRACT
The world will face many challenges and difficulties plus many opportunities in the 21st century and beyond. It is good to plan for the future and yet people and countries and the world must be flexible and somewhat adaptable, because it is hard to predict the future exactly.

However, an almost certain day (or days) of crisis is looming for the whole human race. That is a significant sized asteroid, comet or meteor approaching the earth and heating up in our atmosphere and exploding with the force of many hundreds of huge bombs creating a possible global catastrophe. However, our science and the world’s space programs have advanced (and continue to develop) to the point where this is a thoroughly preventable problem if we act prudently and quickly to organize a defense against this potentially devastating disaster which should never be allowed to occur in the future, even though it has probably occurred many times in ancient history.

Presented here will be some technical examples and ideas for asteroid defense to make our fragile planet safe forever from this future menace.

INTRODUCTION
The asteroid, Apophis (Stone 2008), may possibly threaten the earth in 2029 and/or again in 2036. Estimates by USA and European space experts vary on the probability of it exploding over the earth. However, it has possibly 200 times the volume of the asteroid that exploded over Siberia in 1908 which destroyed a forest of over 800 square miles. Apophis is about 900 feet in length versus 150 feet for the Siberian one. It is disturbing that Apophis (900/150 = 6 and 6x6x6=216), possibly has a volume of 200 times the 1908 one.

Even though the odds of it hitting the earth are small, no odds are acceptable given the potential magnitude of the disaster. The United States space agency, National Aeronautics and Space Agency (NASA) has a whole department (headed by Donald Yeomans) working on asteroid defense. The Europeans are understandably concerned (Isachenko 2009) and working diligently too.

Other countries’ space programs, such as India, Japan, Brazil and also China, can help to eliminate this problem.

Presented here will be a sixteen rocket deflection attack with the actual momentum calculations necessary to divert an asteroid safely. A second example will revolve around a 1200 rocket attack by an international consortium of space programs within a minimum content setting. This will be followed by a staffing and scheduling example to have enough telescopes and scientists using them to map all of the orbits of any potentially earth-threatening dangerous comets, meteors and asteroids. Also, an example involving actual rocket design with a plan to provide enough back-up systems so that the defense rocket has a high probability of working will be provided.

This is clearly an international management problem as (Conley 2010) recommends and also Anatoly Priminov, the head of Russia’s space agency in (Isachenko 2009) is advocating. Additionally, (Maugh 2010) thinks telescopes like the Hubble one can be very useful in preparing this defense and further understanding this world-wide problem so we are not unprepared when this almost certain future day of crisis arrives. Let us look at our three examples, and additionally discuss the ideas of a permanent defensive asteroid corral.

A SIXTEEN ROCKET ATTACK
At some future time astronomers determine that a rock (asteroid or meteor) may head toward the earth and explode in the earth’s atmosphere causing a catastrophe, because it has a mass of 35 billion kilograms and is
traveling at 2300 meters per second. The estimated maximum dimensions are 1000 by 222 by 100 meters. Researcher (Conley 2010) mentions these mass attacks, but does not do the actual momentum calculations as we do here. Therefore, 16 rockets that had been previously launched from the earth with big empty cargo bays, had collected small asteroids in their cargo bays and were awaiting this type of event up in the outer space asteroid coral. So after their engines are fired up, they head toward the big dangerous asteroid at an average velocity of 2500 meters per second (with their 16 separate masses totaling to 80 million kilograms). If they smash into the big bad asteroid from say a north to south direction and the bad asteroid is heading east to west (so to speak, a perpendicular crash) we have 35,000x.23=35080xVE, so VE=22.9475 in the east direction and 80x25=35080xVS so VS=0.570125 in the south direction. Therefore, the arctan (.0570125/22.9475) equals about one seventh of a degree shift of the collided masses away from the collision path to the earth. If this collision happens 124 million miles before the dangerous asteroid’s potential collision with the earth, it should miss the earth by a little over 300,000 miles (give or take a little, depending on gravitational pulls of planets and moons and other factors, etc.).

The engineers want to know what speeds (in meters per second) to run the sixteen rockets (X_b) so that those numbers times the asteroid laden rocket masses in millions of kilograms of 5(15+i.10i) equals 2000. Therefore we want to solve

\[ \sum 5(15+i.10i)X_b = 2000 \]

for all \( X_b \geq 0 \)

and subject to some side conditions the engineers have set up to compare the different rocket engines. They are:

- \( X_4 > X_8 \) velocities
- \( 2X_{12} > X_{11} \) velocities
- \( X_{15} \) wear coefficients
- \( X_{16} \)

\[ X_1 + X_2 + X_3 + X_4 > 1.5 (X_5 + X_6 + X_7 + X_8) \]

and \( 1.3 (X_9 + X_{10} + X_{11} + X_{12}) < X_{13} + X_{14} + X_{15} + X_{16} \)

An eight stage multi stage Monte Carlo optimization produced the following solution subject to the conditions and prevented the explosion on the earth. The rocket velocities in the so called southerly direction (perpendicular to the dangerous asteroid) are

\[
\begin{align*}
X_1 &= 1.08841 \\
X_2 &= 32.22316 \\
X_3 &= 9.77682 \\
X_4 &= 81.35004 \\
X_5 &= 23.91397 \\
X_6 &= 39.73512 \\
X_7 &= 10.99403 \\
X_8 &= 33.91187 \\
X_9 &= 37.33978 \\
X_{10} &= 4.36870 \\
X_{11} &= 15.73495 \\
X_{12} &= 12.21943 \\
X_{13} &= 12.82255 \\
X_{14} &= 48.84333 \\
X_{15} &= 26.23683 \\
X_{16} &= 18.88152
\end{align*}
\]

where the rocket speeds are in the 100’s meters per second. The reasons the speeds or velocities are all different is because of the engineers’ side conditions constraints. Please see the multi stage Monte Carlo multi stage optimization (MSMCO) section for a further explanation of the general purpose MSMCO simulation optimization technique for nonlinear (and linear) multivariate problems.

**AN INTERNATIONAL MINIMUM CONTENT 1200 ROCKET RENDEZVOUS**

A hypothetical asteroid of over a kilometer in length may explode over the earth in a few years time if the world takes no action. Therefore, the international space consortium of six countries has decided on a mass rocket attack on the big bad asteroid. The plan is for 1200 unmanned rockets (some launched with empty cargo bays) to collect little asteroids to add mass and momentum for the projected crash into the dangerous asteroid. The rockets will then smash into the earth threatening asteroid to deflect it off of line from its impending path toward the earth.

Given the estimated success rates of each of the 18 rocket types (3 each from the 6 countries, total of 18 rocket suppliers), and their costs, how many rockets should be purchased from each of the 18 suppliers to minimize the total cost and keep all countries and suppliers on board with the following additional minimum content conditions. At least 8 rockets will be purchased from each of the 18 suppliers and at least 40 in total from each of the six countries. Also, no more than 250 rockets will be purchased from any one supplier.

<table>
<thead>
<tr>
<th>Rocket Type</th>
<th>Success Rates of Rockets (hits one Km Asteroid)</th>
<th>Cost per Rocket in Millions of Euro dollars</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country A</td>
<td>0.98, 0.97, 0.96</td>
<td>385, 370, 355</td>
</tr>
<tr>
<td>Country B</td>
<td>0.95, 0.94, 0.93</td>
<td>340, 325, 310</td>
</tr>
<tr>
<td>Country C</td>
<td>0.92, 0.91, 0.90</td>
<td>295, 280, 265</td>
</tr>
<tr>
<td>Country D</td>
<td>0.89, 0.88, 0.87</td>
<td>250, 235, 220</td>
</tr>
<tr>
<td>Country E</td>
<td>0.86, 0.85, 0.84</td>
<td>205, 190, 175</td>
</tr>
<tr>
<td>Country F</td>
<td>0.83, 0.82, 0.81</td>
<td>160, 145, 130</td>
</tr>
</tbody>
</table>
The reason for the side conditions including all suppliers and all countries in some kind of “minimum content” contribution is to keep all suppliers in the rocket business for future international joint space ventures and also to keep all countries and allies happy that they are contributing to this and hopefully future asteroid (comet and meteor) defense projects to protect the earth. This is certainly an international problem that should require an international solution. The extra cost involved, including all parties, should be worth it politically and practically for future projects. It should also generate more international goodwill.

An eight stage multi stage Monte Carlo optimization (MSMCO) simulation optimization program drawing three million sets of eighteen random numbers (at each stage) between 8 and 250 (and discarding the ones that do not meet the constraints) funneled into a solution of

\[ X_1=11, \; X_2=8, \; X_3=22, \; X_4=10, \; X_5=14, \; X_6=16, \; X_7=8, \; X_8=21, \; X_9=12, \; X_{10}=10, \; X_{11}=15, \; X_{12}=18, \; X_{13}=13, \; X_{14}=250, \; X_{15}=250, \; X_{16}=249, \; X_{17}=250 \]

The printout indicates that at least 1200 rockets should succeed and get to the target. Also, the total rocket purchase cost is 157.4333 billion (thousand million) Euro dollars. This MSMCO program ran in a few seconds of computer time on a desktop PC computer.

Note that countries E and F are supplying many more rockets than the other four. That is acceptable to all because these two countries have advanced space programs and have agreed to fund more of the project. All parties agreed to this as part of the preliminary negotiations.

The major difficulties in space missions with humans on board are saving the crew and launching heavy rockets that must overcome the earth’s considerable gravity, with escape velocities of 15 to 18 thousand miles per hour.

Note that sending up unmanned rockets (many with big empty cargo bays) makes the rockets lighter and also makes a success rate of .91 or .93 acceptable. Therefore, much work must be done on remote guidance of the rockets for collecting mass (small asteroids) in their cargo compartments and also then sending them into their mass rendezvous, or crash into the dangerous (earth threatening) asteroid, to knock it off line.

**SCHEDULING THE SCIENTISTS**

The international space consortium has decided to build and make operational 17 new telescopes to hopefully assist in mapping the orbits and paths of all potentially dangerous comets, meteors and asteroids that could one day threaten our fragile and beautiful earth. Seven of these are to be launched into outer space (like the current Hubble telescope is). The other ten are to be land-based in areas where more telescopes are needed (in addition to the good ones we already have) so that we can view all directions into outer space where asteroids, comets, and meteors may come from.

The consortium has a 35 million dollar budget to hire astronomers and scientists to staff these telescopes. Let \( X_i \) for \( i=1, \; 3, \; 3 \ldots \; 9 \) be the number of nine different categories of employees they plan to hire, with their yearly salary in thousands in parentheses.

<table>
<thead>
<tr>
<th></th>
<th>Most Experience</th>
<th>Average Experience</th>
<th>Least Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ph.d</td>
<td>( X_1 ) 80</td>
<td>( X_2 ) 75</td>
<td>( X_3 ) 65</td>
</tr>
<tr>
<td>Masters</td>
<td>( X_4 ) 60</td>
<td>( X_5 ) 55</td>
<td>( X_6 ) 50</td>
</tr>
<tr>
<td>Degree M.A.</td>
<td>( X_7 ) 43</td>
<td>( X_8 ) 40</td>
<td>( X_9 ) 35</td>
</tr>
</tbody>
</table>

The consortium has developed the following performance equation that they want to optimize subject to budgetary and other constraints. It gives more productivity weight to extra experience and education. Therefore, we seek to maximize the performance equation.

\[ P=\sum(11-i)X_i(2-.05i) \quad (\text{where} \; - \text{means raise to a power}) \]

subject to \( 1 \leq X_i \leq 9 + 10 \) (available staff)

and all \( X_i \) are whole numbers for \( i=1,2,3 \ldots 9 \)

and

\[ X_1+X_2+X_3 \geq 17 \quad \text{or at least one Ph.d. per telescope} \]

\[ X_4+X_5+X_6 \geq 17 \quad \text{or at least one B.A. per telescope} \]

and

\[ 80X_1+75X_2+65X_3+60X_4+55X_5+50X_6+43X_7+40X_8+35X_9 \]

\[ \leq 35000 \] budget constraint in thousands of dollars

A ten stage multi stage Monte Carlo optimization (MSMCO) simulation drawing 65000 sample answers at each stage produced the solution of:

\[ X_1=105 \; \; \; X_2=115 \; \; \; X_3=125 \]

\[ X_4=135 \; \; \; X_5=5 \; \; \; X_6=3 \]

\[ X_7=0 \; \; \; X_8=17 \; \; \; X_9=0 \]

and \( P=270246.531 \). Therefore, many Ph.d.s should be hired for fairly optimal performance within constraints.

This is, of course, a hypothetical staffing optimization problem that MSMCO solved. However, Richard Stone, the Asian editor of *Science*, in (Stone 2008) points out that
the shortage of telescopes, and people to staff them to map all of these orbits accurately, is a very real problem.

One criticism of the approach of putting engines on small asteroids and using them to smash into the dangerous asteroid or collecting small asteroids in a rocket’s cargo bay and using the rocket to crash into the big bad asteroid (Conley 2010) is that with all that mass the little asteroids or rockets could not catch up to a faster moving dangerous comet, meteor, or asteroid.

However, if we map accurately and know the orbits and routes of these dangerous earth threatening objects, the space agencies can send their little asteroids and rockets millions of miles down the ranges of these bad asteroids and wait for them and block their path to the earth.

Therefore, it is vital that the world knows where these dangerous objects will be and at precisely when they will be there. This way we can stay ahead of them and block their path to the earth regardless of their size and speed. We do not have to destroy these objects. We just need to knock them off their path to the earth.

First prize in this new game of “outer space croquet,” so to speak, is forever protecting the earth. It is a prize all of humanity can share.

AN ASTEROID CORRAL

The defense of the earth now and for all future generations against the catastrophic menace of an asteroid, meteor, or comet exploding in the earth’s atmosphere causing untold devastation (Stone 2008) is a long term international project. Therefore, this is really a management problem even though considerable technical expertise is involved. Therefore, the management team should consider all defensive ideas, the ones advanced here and all others too (Conley 2010).

One possibility is to get ahead of this problem by launching several hundred unmanned rockets with empty storage areas and send them up to the asteroid belt (between Mars and Jupiter) to collect small asteroids to add mass. Then bring them all back to our moon and put them in stationary orbit around the moon. There they will wait until an asteroid deemed to be threatening the earth appears. The international space consortium supervising this defensive corral will then fire their moon escape rockets and send them over to smash into the dangerous asteroid to deflect it from its path toward the earth.

Also, in future years, another asteroid corral up in the asteroid belt could be considered. Many people believed that dinosaurs ruled the earth for so long, because they were the fittest creatures (“survival of the fittest”). However, the conventional scientific view is that two asteroids crashed into each other in the asteroid belt (between Mars and Jupiter) deflecting one toward the earth. A few years later it exploded in the earth’s atmosphere wiping out the dinosaurs (Maugh, 2010). Richard Stone, in his careful overview of the asteroid “problem” (Stone 2008), recommends that humans should try to be more wise than the dinosaurs and plan ahead. The geophysicists (Ryan and Pitman 2000) proved that as an ice age was ending thousands of years ago the oceans rose to the point where a channel was cut through what is now the Dardanelles, the Sea of Marmara and the Bosporous (in present day Turkey) and flooded the Black Sea which was then a fresh water lake half of its present day size and several hundred feet below sea level. Religious scholars are almost unanimous that this was the legendary Noah’s Flood (although Ryan and Pitman make no religious claims, only scientific ones). The point is, if only the world had advanced to the point that the flood control engineers in the Netherlands (and elsewhere) are today, this flooding and destruction of an entire civilization could have been prevented with flood barriers and dykes blocking the entrance to the Dardanelles. Therefore, with comet, meteor and asteroid defense we also must be prepared and keep our space programs strong and advancing to protect our fragile earth. Additionally (Salk 1973), says it best in his view of the challenges we face, implies by the title of his book, that “survival of the fittest” perhaps should be replaced with the view of The Survival of the Wisest.

FIVE ALTERNATIVE SUPPLIERS FOR EACH OF THIRTY ROCKET SYSTEMS

The actual unmanned rockets that are launched from the earth to help with the asteroid defense have 30 systems that must all work for the rocket to be successful. Therefore, five alternative suppliers have been identified (in three countries) for each of the 30 systems. Suppliers 3 and 4 are in the same country, as are suppliers 1 and 2. Supplier 5 is in a third country. Therefore, to keep all suppliers on board, at least one system of each type will be purchased from each of the 150 suppliers and at least 20 percent of the total cost of the systems plus backups (in case of system failure) must be spent on supplier 5 in the third country. Also, at least 20 percent of the total cost must be spent in country two (which has suppliers 3 and 4).

Therefore, given these minimum content international supplier constraints (to keep all suppliers and countries content that this is a joint effort) and the following costs and failure patterns of the 150 rocket systems, how many of each of the 150 systems should be purchased and installed in the asteroid defense rocket to have at least a 95 percent probability that the rocket will not fail, and succeed in its mission?
The component failure probabilities are for suppliers
\[
\begin{align*}
&1 & 2 & 3 \\
&.45 - .008 & .65 - .01 & .50 - .011 \\
&4 & 5 \\
&.40 - .009 & .60 - .012 \\
\end{align*}
\]
and \(i = 1, 2, 3, \ldots, 30\).

The component costs (in thousands of dollars) are for suppliers
\[
\begin{align*}
&1 & 2 & 3 \\
&10 + 5jX_j & 11 + 4(j+1)X_{j+1} & 6 + 2(j+2)X_{j+2} \\
&4 & 5 \\
&7.5 + 1.5(j+3)X_{j+3} & 6.1 + 7(j+4)X_{j+4} \\
\end{align*}
\]
and \(j = 1, 6, 11, 16, 21 \ldots, 146\) by steps of 5.

Therefore, we write a short multi stage Monte Carlo optimization (MSMCO) simulation optimization program that has eleven stages drawing 20,000 sample solutions at each stage which stores the minimum cost solution at each ever improving stage (and still meets all of the constraints). The following printout in Table 2 summarizes the answer.

Table 2: How Many of Each Component

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>15931.10</td>
<td>0.95</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
CONCLUSION

The idea of an asteroid corral or two, in the future, to protect the earth plus four actual multivariate problems for comet, meteor and asteroid defense were presented. A momentous diversion of the dangerous asteroid, a minimum content large scale rocket attack on an earth threatening asteroid, a scheduling example, and probability calculations for number of backup systems needed for the rockets to succeed were worked on.

The MSMCO simulation optimization solution technique was also used (and reviewed) to solve our four multivariate asteroid defense examples. Complacency in dealing with this problem is very unwise. Once again, the title of famed scientist and researcher, Jonas Salk’s book says it best, The Survival of the Wisest. We must take action now to plan for that certain day of crisis. We must be ready and prepared.

REFERENCES


BIOGRAPHY

WILLIAM CONLEY received a B.A. in mathematics (with honors) from Albion College in 1970, an M.A. in mathematics from Western Michigan University in 1971, an M.Sc. in statistics in 1973 and a Ph.D. in mathematics - computer statistics from the University of Windsor in 1976. He has taught mathematics, statistics, and computer programming in universities for over 30 years. He is currently a professor emeritus of Business Administration and Statistics at the University of Wisconsin at Green Bay. The developer of multi stage Monte Carlo optimization and the CTSP multivariate correlation statistics, he is the author of five books and more than 200 publications world-wide. He is a member of the American Chemical Society, a fellow in the Institution of Electronic and Telecommunication Engineers and a senior member of the Society for Computer Simulation. Career highlights include presentation of two papers at National Aeronautics and Space Administration (NASA) conferences in Houston, Texas and Washington, D.C.
SIMULATING MEDICAL DISASTERS WITH REALISTIC VICTIM PROFILES

Christophe Ullrich  
Filip Van Utterbeeck  
Department of Mathematics  
Royal Military Academy  
Brussels  
Belgium

Erwin Dhondt  
Michel Debacker  
Research Group on Emergency and  
Disaster Medicine  
Vrije Universiteit Brussels  
Brussels  
Belgium

Erwin Dhondt  
Queen Astrid Military Hospital  
Neder-Over-Heembeek  
Belgium

KEYWORDS
Health care, Ressource management, Decision support systems, Model design, Discrete simulation

ABSTRACT

We propose a methodology to generate realistic victim profiles for medical disaster simulations based on victims defined in the VictimBase library. We apply these profiles in a medical disaster model where victim entities evolve in parallel through a medical response model and a victim pathway model. These models interact in correspondence with the time triggers and intervention triggers from VictimBase. We show how such a model can be used to assess the impact of asset availability and implemented victim prioritization rule on the clinical condition of the victims.

INTRODUCTION

Medical disaster management is a young medical discipline which requires a solid scientific foundation. However, it is obviously impossible and/or ethically inappropriate to identify the experimental and control groups essential for the hypothesis testing which is required to conduct scientific randomized controlled clinical trials. There are currently no defined (validated) performance outcome measures as to what constitutes a “good” disaster response or not.

An improved methodology is clearly needed to judge and evaluate the effectiveness and adequacy of health and relief services provided during disaster medical response. Computer simulation has already been widely applied in health care studies. For example, Dugay and Chetouane (2007) describe a discrete event simulation study of another emergency department. Günal and Pidd (2009) present a literature review about discrete event simulation for performance modeling in health care. Su (2003) uses an object oriented simulation software to improve the emergency medical service of Taiwan. Brailsford and Hilton (2000) compare system dynamics and discrete event simulation to see which method should be applied in specific circumstances. McGuire (1998) dedicates a chapter to the application of simulation tools in health care.

The research presented in this paper is the development of a stochastic discrete event simulation model which will be used to evaluate applicable methodologies and identify rules of best practice for medical disaster and military battlefield management in different large-scale event scenarios. The simulation model for these scenarios will be based on three principal components: the victim pathway model, the medical response model and a set of outcome performance measures. This paper focuses on the use of realistic victim profiles for medical disaster simulation.

The paper is organized as follows. In the next two sections, we introduce the VictimBase project and the victim profiles used in our simulation model. The fourth section details the implementation of the medical response mode and its interaction with the victim pathway model. In the fifth section, results of a small test case are presented. Finally, in the last two sections, we conclude and discuss future work.

VICTIMBASE

VictimBase is an on-line tool designed to support the medical community in creating dynamic victims for use in disaster medicine exercises. The tool is specifically focused on creating a database of disaster victims to be used in medical disaster management exercises, research, simulation and training.

Each victim includes three primary components: the general victim data, a set of clinical conditions (which include primary survey, triage and diagnostic test data) and a set of transitions, which are either time trigger based or intervention trigger based.

The general victim data provides an overview of the victim as well as the victim parameters that do not change over time or as the result of a clinical intervention (e.g. name, gender, anthropometric data, injury or illness and medical history). A clinical condition (CC) refers to a specific victim state. A complete victim includes multiple CCs. Each CC consists of primary survey data, triage data and diagnostic
test data. We refer to Van Utterbeek et al (2011) for a more elaborate description of VictimBase.

VICTIM PROFILES

Each victim profile is generated using a victim from the VictimBase library. We will only use profiles with maximum one intervention trigger per CC. As such, each victim’s current medical status is uniquely identified by his clinical condition and there are two possibilities to evolve towards a different clinical condition: either the time interval defined by the time trigger (TT) elapses or the victim receives the medical attention defined by the medical intervention trigger (MIT) before said time trigger elapses. The application of a medical intervention trigger requires the presence of specific assets. The local emergency medical services system is organized as a three-tiered system. The first tier comprises an ambulance car and two ambulance men, skilled according to the Basic Life Support-Defibrillation (BLS-D) level. The second tier consists of an ambulance man driving the ambulance accompanied by an emergency registered nurse, delivering limited Advanced Life Support (ALS), according to standing orders. The third tier finally consists of a staffed intervention car with both an emergency nurse and an emergency physician, with full ALS capabilities (ALS+). Each skill level implies mastery of the skills associated with the inferior levels. The list of assets required for a medical intervention trigger is predefined in the model.

We will consider only two types of victims of a road traffic accident in an urban setting involving a bus and a car to illustrate the workings of the model and to obtain the results presented. Every medical intervention requires a broad array of materials and consumables. To avoid going into too much detail, we will mention in the table 1 only the number of different materials and consumables required.

Table 1: Assets required for the medical interventions

<table>
<thead>
<tr>
<th>MIT</th>
<th>Skill Level</th>
<th>Time (min)</th>
<th>Number of materials</th>
<th>Number of consumables</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIT₁</td>
<td>ALS</td>
<td>16</td>
<td>4</td>
<td>19</td>
</tr>
<tr>
<td>MIT₂</td>
<td>ALS+</td>
<td>19</td>
<td>4</td>
<td>26</td>
</tr>
<tr>
<td>MIT₃</td>
<td>ALS</td>
<td>3</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>MIT₄</td>
<td>ALS+</td>
<td>5</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>MIT₅</td>
<td>BLS-D</td>
<td>9</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>MIT₆</td>
<td>ALS</td>
<td>14</td>
<td>4</td>
<td>16</td>
</tr>
</tbody>
</table>

Victim ID 1

The victim ID1 is a middle aged male, having a front seat in the bus and suffering blunt chest trauma. Basically, three different pathways of clinical evolution are programmed for this single victim as represented in figure 1.

![Pathway 1: blunt chest trauma](image)

After 15, 20 and 15 minutes respectively without any medical treatment intervention, his clinical condition is progressively worsening from CC0, over CC1 and CC4 to CC7; CC7 being his worst outcome end clinical condition leading to death.

In a second pathway, on the other hand, this victim transitions from CC0 to CC2, after a set of 9 treatments has been applied.

His condition generally improves, having better respiratory parameters while being asleep in an artificial coma (CC2), but his circulatory condition still does not satisfy. That is why a second intravenous (IV) access is put in place and more crystalloids are added in order to further stabilize the victim. This intervention results in CC6, however without the expected effect.

After another 20 minutes without further stabilizing therapeutic intervention, the victim transitions to CC9, which is an intermediate outcome end clinical condition, since not optimal and possibly still life threatening.

If on the contrary, at CC6, a pleural drainage is applied, in order to resolve the life threatening tension pneumothorax, accompanied by some additional analgesics, the hemodynamics of the victim definitely improve, saving him from further deterioration and leading to the best outcome end clinical condition, CC8.

The third pathway which is starting at CC1, is characterized by a very complete, although delayed, first treatment intervention comprising the same 9 treatments as initially applied in the second pathway, plus the live saving pleural drainage.

As such, the victim transitions immediately to a more stable CC3, however requiring in a second step some more crystalloids via a second IV line, may be in order to optimize circulation after a too generous analgesedation.

This results in CC5 which further improves, 10 minutes later and even without further intervention, to CC8, the best outcome end clinical condition.
Victim ID 2

Type 2 is a young male (approximately 17 years old) ejected from the car and having a fractured right arm, multiple face lacerations and a cerebral commotion due to blunt head trauma. The evolution of clinical condition is presented by the figure 2.

![Diagram of Pathway 2: multiple face lacerations and a cerebral commotion](image)

The CC0 → CC1 → CC4 pathway is the no treatment pathway, leading to death, the worst outcome end clinical condition.

An alternative pathway consists of a delayed rescue of the victim, namely at CC1, i.e. after a 10 minutes therapy free interval.

Despite the full treatment at CC1, the victims clinical condition is further worsening CC3 probably because of neglecting the airway and concomitant increasing oxygen depth in the crucial first ten minutes. Because of full respiratory support the victim will not die on the scene; although his final outcome is bad due to irreversible traumatic brain injury. CC3 is the intermediate outcome end clinical condition.

The best outcome end clinical condition CC5 is obviously achieved after performing as soon as possible a small set of simple treatments mainly focused on securing the airway.

The victims clinical condition especially the respiratory function is immediately stabilized and not deteriorating, even after a further 30 minutes time frame of no therapeutic intervention.

MEDICAL DISASTER MODEL

A disaster scenario is composed of incidents associated to a point in space and time, expressed in terms of material damage and victims, on which the disaster manager has no control. The decision making process of the disaster manager is expressed in terms of material and human resources made available and their assignment to tasks in space and time. The consequences of these decisions are assessed and this gives rise to rules of best practice.

Our medical disaster model contains two major components: a medical response model (where the victim interacts with the environment and with the resources at the disposition of the disaster manager) and a victim pathway model (where the current clinical condition of every victim is monitored). The specificity of our simulation model is the fact that the victim entities will evolve through both the medical response model and the victim pathway model in parallel, while the interaction between both models is ensured through triggers (figure 3). In this section we focus on the implementation of these two models and their interaction, and we discuss an implementation using the Arena software. References for Arena include (Altink and Melamed 2007), (Kelton et al 2010) and (Rossetti 2010).

![Diagram of Disaster model](image)

The *victim create* module creates all the victims of the scenario (in most disaster scenarios all the victims will exist at the simulation start, immediately after the disaster event which marks the start of the scenario). For every victim we define the specific victim profile which corresponds with the nature of his injuries in an attribute. The number of victims generated and the mapping of the victims to victim profiles is of course scenario-specific.

The *victim duplicate* module creates a copy of the victim entity. Subsequently, both copies are assigned an attribute “Entity ID” which contains the identifier of the corresponding duplicate. This attribute is crucial for the communication between the medical response model and the victim pathway model.

The *medical response* model describes the environment (victims, places, time related factors,...), the resources at the disposition of the disaster manager (medical personnel, equipment, means of transportation,...) as well as the operational procedures and decision making models used (triage procedure, assignment of victims to hospitals logic, ambulance dispatch logic,...).

The *victim pathway* model contains the victim profiles and all the information found in the corresponding VictimBase victim descriptions. This includes for example the required parameters for triage procedures and diagnosis and all information concerning the clinical conditions. The final victim pathway model will be a modular library containing a large number of victims which can be used to generate the required random victims for the scenario under consideration. The two victim profiles used as an example throughout this paper have been described above.
The implementation of the four modules has been described in detail in Van Utterbeek et al (2011) We propose an improved implementation of the victim pathway module in order to manage more easily all the different kinds of victim pathways defined in VictimBase. The pathway logic for each victim type was previously hard coded in Arena. We improved this implementation and introduce a model logic which can handle all kinds of pathway without any modification of the Arena model. This logic is represented in figure 4 below.

**Figures 4: Clinical condition logic**

Each clinical condition (CC) can evolve by transitions, which are: time trigger (TT) and medical intervention trigger (MIT), only TT and only MIT. An end CC is a CC with no triggers. For the first two cases, the victim entity enters through a station (Time Trigger), passes the end CC test and goes to a delay module where it will be held until the time trigger delay for this CC elapses. The value of the time trigger delay is stored in an attribute of the victim entity. If no timely medical intervention is initiated, then the victim triggers a VBA module to update the victim's CC parameters. The parameters corresponding with the next CC are again read by the VBA module from the VictimBase database which contains all the victim pathway data. Finally, a route module sends the victim entity to the entry station corresponding to the type of the new CC. If the clinical condition has no time trigger, the victim entity is sent to the intervention trigger station and it will be held until the next medical intervention trigger. If this is an end clinical condition, the victim entity fires a VBA module to update a log file containing all the victims, their end clinical condition and pathway ID, then it is immediately disposed of.

It is however possible that a medical intervention is initiated in the medical response model before the time trigger delay elapses or while the entity is being held until next intervention. The entity is then sent to the "Being treated" logic branch. The victim in question will be held there until the Medical Process logic sends the victim entity onwards to the next CC.

**RESULTS**

The test case results presented in this section serve as proof of concept for the medical disaster model presented in the previous section. We use the two example victim profiles from section “victim profiles” and a simplified medical response model with a single medical process (at the disaster site zone). We consider 10 scenarios: 2 different settings for the assignment of the priority in the medical process queue combined with 5 different settings for the assets present in the zone. The 5 settings for the assets are shown in table 2 below. Each ambulance carries materials and consumables to achieve twice each of the medical interventions of their skill level and below, and also a medical team with skills related to the skill level of the ambulance. In each scenario we generate 10 victims (5 of each profile ID) and we note the number of victims in each end CC for each scenario. Priority rule 1 assigns a higher priority to victim ID1 than to victim ID2. Priority rule 2 is identical but assigns the highest priority to victim ID2. Within each victim pathway the classification from the highest priority to the lowest priority CC is the following for victim ID1: CC1 > CC0 > CC2 > CC3 > CC6 > CC5 > CC4 = CC7 = CC8 = CC9, and for victim ID2: CC1 > CC0 > CC2 > CC3 = CC4 = CC5.

**Table 2: Assets present in the zone**

<table>
<thead>
<tr>
<th>Number of ambulance</th>
<th>Number of Ambulance</th>
<th>Number of Ambulance</th>
</tr>
</thead>
<tbody>
<tr>
<td>S BLS-D</td>
<td>S ALS</td>
<td>S ALS+</td>
</tr>
<tr>
<td>Assets 1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Assets 2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Assets 3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Assets 4</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Assets 5</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

Tables 3 and 4 show the results obtained using priority rule 1 for the 5 different assets settings. When the number of assets is low (assets 1, 2, 3), then 3 (60%) victims with profile ID1 end with a clinical condition with only MIT as further transitions. There are not enough consumables in the system which causes them to remain stuck in these CC's. When the assets are higher the victims receive more treatment. For victim ID2 only one victim receives treatment late and finishes in CC3.

**Table 3: Results with priority rule 1, patway ID 1**

<table>
<thead>
<tr>
<th>CC</th>
<th>CC</th>
<th>CC</th>
<th>CC</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets 1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Assets 2</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Assets 3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Assets 4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Assets 5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

**Table 4: Results with priority rule 1, patway ID 2**

<table>
<thead>
<tr>
<th>CC</th>
<th>CC</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Assets 2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Assets 3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Assets 4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Assets 5</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Tables 5 and 6 show the results obtained using priority rule 2 for the 5 different assets settings. With all the assets, all the victims ID2 finish in the best end CC (CC5). For the victims ID1, we obtain the same results as with priority rule 1 with the exception of assets 5: for this assets setting one victim did not receive the MIT.

Table 5: Results with priority rule 2, patway ID 1

<table>
<thead>
<tr>
<th></th>
<th>CC 2</th>
<th>CC 3</th>
<th>CC 7</th>
<th>CC 8</th>
<th>CC 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets 1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Assets 2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Assets 3</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Assets 4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Assets 5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6: Results with priority rule 2, patway ID 2

<table>
<thead>
<tr>
<th></th>
<th>CC 3</th>
<th>CC 4</th>
<th>CC 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets 1</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Assets 2</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Assets 3</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Assets 4</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Assets 5</td>
<td>0</td>
<td>0</td>
<td>5</td>
</tr>
</tbody>
</table>

These results clearly show the influence of both the priority rule and the available assets on the end CC of the victims. It is interesting to note that optimal priority rule selection and asset availability are not necessarily independent.

CONCLUSIONS

We propose a victim profile for medical disaster simulations based on victims from the VictimBase library. These victims are highly realistic and their validity is assured by the VictimBase account system: only medical professionals who are granted authorship by the European Master in Disaster Management Academy can create these victims. This victim profile is applied in a medical disaster model where victim entities evolve in parallel through a medical response model and a victim pathway model. We show the implementation in Arena of the interaction between a medical process in the medical disaster model and a victim profile in the victim pathway model. These interactions correspond with the time triggers and intervention triggers from VictimBase. We provide a simple example that shows how such a model can be used to assess the impact of asset availability and the decision rule used to prioritize the victim being treated on the clinical condition of the victims.

FUTURE WORK

In future work a more complex and realistic medical response model will be developed. The victim pathway model will be expanded into a library containing a large number of victim profiles. These components will then be used to investigate several scenarios: a pilot case study describing a major road traffic accident will be studied for verification and validation of the implemented medical response and victim pathway models and for performance and outcome measures evaluation. Afterwards, four scenarios (an aeronautical catastrophe, a CBRNE (Chemical, Biological, Radiological, Nuclear, and Explosives) incident, mass gatherings and hospital catastrophes) will be developed. The expected outcomes of this project are evidence based recommendations and rules of best practices for optimal disaster management and medical battlefield management in different large-scale event scenarios, as well as evidence based recommendations for teaching, training and research in medical disaster management.

REFERENCES


Su S. 2003. “Modeling an emergency medical service system using computer simulation”. International Journal of Medical Informatics, N°1-3, 57-72

CLIMATE CHANGE VULNERABILITY ASSESSMENT WITH CONSTRAINED DESIGN OF EXPERIMENTS, USING A MODEL DRIVEN ENGINEERING APPROACH

Romain Lardy(1,3,4), Gianni Belloccchi(1), Bruno Bachelet(2,3,4), David R.C. Hill(2,3,4)

(1) Grassland Ecosystem Research Unit, French National Institute for Agricultural Research (INRA), 234 Avenue du Brézet, 63100 Clermont-Ferrand, France
(2) Clermont Université, BP 10448, F-63000 CLERMONT-FERRAND
(3) CNRS, UMR 6158, Université Blaise Pascal, LIMOS, F-63173
(4) ISIMA, Computer Science & Modelling Institute, BP 10125, F-63177

KEYWORDS
Climate Change Vulnerability, Experimental Design, Grassland Ecosystems, Model Driven Engineering

ABSTRACT

Vulnerability is the degree to which human and environmental systems are likely to experience harm due to a perturbation or stress. Vulnerability assessment under climate change needs a huge amount of simulations. The set of simulations needed depends on the uncertainties hypothesis. In this study, we propose an approach for vulnerability assessment with different types of design of experiments. We build up model of these designs, to identify a common pattern seen as metamodel, which models can be conformed to. In order to build up dedicated distribution platform for vulnerability analysis, we use a Model Driven Engineering approach.

INTRODUCTION

Vulnerability is the degree to which human and environmental systems are likely to experience harm due to a perturbation or stress (Kasperson et al. 2005). It has become in recent years a central focus of the global change (including climate change) and sustainability science research communities (Füssel 2007).

The purpose of vulnerability assessment is not only to determine which systems are the most vulnerable, but also to understand why they are so (Luers et al. 2003). This information is crucial in the process of determining strategies of adaptation to, or mitigation of the effects of change. In the context of climate change, this remains not enough studied (Easterling et al. 2007).

The literature contains many definitions of vulnerability, going from the notion of sensitivity (Dowing and Patwardhan 2005) to more complex ideas, yet tacking into account the exposure history of the system (Turner et al. 2003) up to residual impacts of climate change after adaptation (IPCC 2001). Due to the great number of sensitive parameters influencing climate change vulnerability, and because of the high level uncertainties in climate change impact studies (e.g. emission scenario, climate modelling, downscaling and initialization, and modelling of the impacts on a target system), vulnerability assessment (which is partly correlated to sensitivity analysis) will necessary need many simulations. Moreover, taking into account potential adaptations means more assumptions and more simulations. It is clear that a pertinent Design of Experiment (DOE) will be absolutely necessary to reduce the time required for simulations. Such an amount of simulations also needs high performance computing, at least for the distribution of simulations. In this context, our paper presents an approach for climate change vulnerability assessment with constrained DOE, using a model driven engineering approach. This work is meant as a preliminary step towards software framework to deal with the distribution of experimental designs under vulnerability constraint.

In order to present our approach, we use ModVege (Jouven et al. 2006a), a mechanistic model for the dynamics of production, structure and digestibility of managed permanent pastures. This model has been retained to study the feasibility of our approach, because on the one hand, it is complex enough to reproduce climate variability impacts on a pasture (Jouven et al. 2006b) but on the other hand, it requires relatively few inputs and limited computational time. Our approach will then be applied to larger classical biogeochemical models typically used for this kind of studies (e.g. PaSiM, Graux et al. 2011).

This work is the continuity of the model-driven, reverse-engineering applied to design of experiment by Lardy et al. (2011). The remainder of the paper is structured as follows. The next section presents the ModVege model. The three following sections present the methodology used, with special focus on DOE and Model Driven Engineering (MDE). Then, we detail our vulnerability assessment methodology with a special spotlight on DOE and associated models. Our MDE approach is completed by the proposition of a metamodel of agro-ecological models with its associated DOE. This approach aims at proposing abstractions that will enable the distribution of experimental designs on high performance computing platforms.

MODEL DESCRIPTION

ModVege (Jouven et al. 2006a) is a multi-year, mechanistic model simulating the dynamics of production, structure and
digestibility of managed permanent pastures. It is designed to respond to various defoliation regimes, based on five assumptions. Firstly, the average value of the biological attributes of vegetation (functional traits) can explain the functioning of a permanent pasture (Louault et al. 2005). Secondly, sward heterogeneity can be modelled by the relative abundance of the structural plant components (Carrère et al. 2002) (i.e. green leaves and sheath, dead leaves and sheath, green stems and flowers, and dead stems and flowers). Thirdly, like many grassland dynamic models (e.g. HPM, Thornley 1998), senescence, growth and abscission can be modelled by continuous fluxes, calculated on daily-time step. Fourthly, due to reserves storage and mobilization in plant organs, shoot growth, based on a light-utilization efficiency approach, is modulated by a seasonal pattern (e.g. Volenc et al. 1996). The latter is considered as a functional trait. The last assumption is that the quality of green compartment, abscission and senescence are influenced by compartment ageing. The model was evaluated for upland grasslands in the Auvergne region of France (Jouven et al. 2006b).

**DESIGN OF EXPERIMENTS**

Design of experiments (DOE) has a rich history, with many theoretical developments and practical applications in a variety of fields. Since the beginning of computer simulation, DOE has been an active research field (Kempthorne 1952; Ambiard et al. 2003). In the modelling field, DOE is a needed tool for efficient testing and analysing the behaviour of a model (Kleijnjen 1987). Most of model simulations aim at exploring and/or testing the behaviour of the model. A parameter or an input is called a factor in the DOE terminology, it could be either qualitative or quantitative (Kleijnjen et al. 2005). Each factor can take two or more values, called levels of factors. An experimental design is a combination of factor levels.

Whatsoever for verification and validation or for the different models’ usage, a huge number of simulation runs are necessary. In particular, for environmental dynamics modelling, models have become increasingly more complex at the pace of computer power. Due to the high number of parameters required by the model and to the computation time of a single run, the needed time for a full factorial DOE is usually too expensive for a sequential machine. This implies that, first of all we use smart but less complete DOE and, on top of that, we have to use distributed computing. The use of a proper DOE will help to get, firstly, all the information we are looking for. For example, in the case of sensitivity analysis, the DOE is important to get relevant sensitivity to all parameters without neglecting their interactions. The second point is to have the smallest number of simulations for the best quality results, thus implying optimization of the total computation time. Computation time is then considerably reduced thanks to the distribution of processes on parallel architectures.

**SIMULATION WITH MODVEGE**

The ModVege model, initially developed in Python, has recently been re-implemented in Java for the purpose of integration into a modelling platform. As most field-scale management decisions are taken on daily basis, the model runs on daily-time step.

To sum up, model inputs can be grouped into four classes (Figure 1):

- **Vegetation** data, i.e. average functional traits of vegetation community (characteristics of functioning of the plant) and initial sward status (biomass and age of the four compartments),
- **Site** properties, i.e. soil nutrition index and soil water-holding capacity,
- **Environment** data (photosynthetically active radiation, air temperature, precipitations and potential evapotranspiration),
- The **management strategy**, which is composed of a number of mowing events.

![Figure 1. UML metamodel of the ModVege input model](image)

Few outputs generally are used, either for direct use or for coupling with other models (e.g. animal model). The most used are the standing and the digestibility of the grass. Figure 2 provides an example of result on upland grassland in the Auvergne region of France.

**MODEL DRIVEN ENGINEERING**

Model Driven Engineering (MDE) is part of software engineering that studies, since more than a decade, software development, maintenance and evolution with a unifying modelling approach (Favre et al. 2006). The Model Driven
Architecture (MDA) is a set of industry standard promoted by the Object Management Group (OMG). The separation between the descriptions of the platform independent system part (PIM, Platform Independent Model) and of the platform specific one (PSM Platform Specific Model) characterizes the MDA, whereas the MDE is a global integrative approach (Favre et al. 2006) for various technological spaces. MDE relies on three fundamental concepts: the “model”, the “metamodel” and the “transformation procedure”.

![Graph A: Standing biomass](image1)

**Figure 2.** A) Standing biomass and B) digestibility of the grass on intensive permanent grassland at Marcenat, in the Auvergne region of France.

A model is a simplified representation of a system. The system is an entity modelled in order to study it, to understand it, and to predict in a mastered context other than reality. The model could be defined by the relation “is a representation of” between itself and the studied system (Hill 1996; Atkinson and Kuhne 2003; Seidewitz and Technologies, 2003; Bézivin 2004). Nevertheless, in the MDE context, this definition is not sufficient because it does not allow the model to become “productive” (i.e. interpretable and exploitable by a machine). That is why many authors have preferred the following definition (Kleppe et al. 2003): “A model is a description of (part of) a system written in a well-defined language”, since it is more adapted to transformation procedures that enable models to become productive.

The notion of well-defined language indirectly points to the second MDE principle, i.e. the “metamodel”. Different definitions can be found in the literature: “a metamodel is a model that defines the language for expressing a model” (OMG 2002); “a metamodel is a specification model for a class of SUS (System Under Study) where each SUS in the class is itself a valid model expressed in a certain modelling language” (Kleppe et al. 2003). Unlike to the popular opinion, a metamodel is not a model of model, since it is better defined as a model of modelling language. This definition is based on the following relation: a model “is conform to” a metamodel. For instance, in the context of Object-Oriented Programming, if we consider the object as a model of reality, then the class is a metamodel and the object “is conform to” its class. However, metamodels can have specific forms depending on the technical domain such as:

- XML technologies: an XML file is conform to a Document Type Definition (DTD) or to an XML schema,

- language theory and compilation: a source code is conform to its grammar,

- in cartography, if our system is France, our model could be an IGN (French National Geographic Institute) map and its metamodel its legend: a map is conform to its legend,

- Standard Template Library (STL): a `vector<int>` is conform to `Vector<T>` metamodel.

MDE principles are relevant for all types of models, either object-oriented or not. MDE is not restrained to a technical domain.

Nevertheless, to get a productive model, it is necessary to describe how to transform it. This aspect corresponds to the third MDE concept: “transformation of model”. Unlike the two other notions, there is no consensus for its definition (Rahim and Mansoor 2008; Lano and Clark 2008; Iacob et al. 2008). According to Favre (2004), the relation could be defined as “is transformed in”. As for the metamodel, the transformation can take different forms under the technical domain (Favre et al. 2006), for example:

- eXtensible Stylesheet Language (XSLT) into XML language,

- compilation and code generation for language theory

- Model to Model: Atlas Transformation Language (ATL)
The Model Driven Engineering approach is in our case a good way to tackle the problem of designing a framework dealing with many types of experimental design and different ways to assess vulnerability.

**VULNERABILITY ASSESSMENT METHODOLOGY**

**A two-stage approach**

Climate change vulnerability literature is rich on definitions. For example, Füssel and Klein (2006) identified three categories of vulnerability definitions:

- The “risk-hazard framework” definition. In this case, vulnerability is defined as response relation between an exogenous stress and its negative effects (Doking and Patwardhan, 2003). This definition is rather similar to the sensitivity definition in the IPCC (2001).

- The social constructivist model. Here, vulnerability is seen as the set of socio-economic causes explaining the difference between sensitivity and exposure.

- The IPCC 2001 definition: “vulnerability is defined as the extent to which a natural or social system is susceptible to sustaining damage from climate change. Vulnerability is a function of the sensitivity of a system to changes in climate (the degree to which a system will respond to a given change in climate, including beneficial and harmful effects), adaptive capacity”. This definition is only centred on climate change, but could be easily extended: Vulnerability is the degree to which a human or environmental system is likely to experience harm before being damaged (Kasperon et al. 2003; Turner et al. 2003).

A few indicators exist in the literature on vulnerability assessment. In this publication, we will narrow their use to what is proposed by Luers et al. (2003) in their method. This vulnerability evaluation is based on four concepts: the state of the system relatively to a damage threshold, its sensitivity, its exposure history, and its adaptation capacity.

As we wish to evaluate different vulnerability assessments in a comparative fashion, depending on definitions, outputs of interest and tested hypotheses, we will need to be able to deal with vulnerability with or without adaptation. This is why we propose a two-step approach (Figure 3):

1. Firstly, a sensitivity analysis step, whose aim will be to estimate vulnerability without adaptation and to calculate response surfaces. A response surface is a model or approximation of this implicit Input/Output (I/O) function that characterizes the relationship between inputs and outputs in much simpler terms than the full simulation (Kleijnen et al. 2005).

2. In the next step, we will try to minimize vulnerability under constraints of actual adaptation capacity.

In the first stage, the approach will consist in building up a suitable experimental design for sensitivity analysis, then to achieve simulations thanks to the impact model. The analysis of the results will give us an estimation of vulnerability without adaptation (i.e. climate change potential impacts). It will also allow us to build up response surfaces (one per considered output). If the quality of the sensitivity analysis and the response surface is considered good enough, then we go throw the next step, otherwise we complete the experimental design for sensitivity analysis.

![Figure 3. Proposed approach for vulnerability assessment](image)

In the second stage, we start by a step of vulnerability minimization thanks to a metaheuristic applied on the previously calculated response surface, while taking into account the real adaptation capacity. Then, we build up an experimental design to evaluate the vulnerability minimum and its robustness. Robustness is defined, here, as the inverse of sensitivity to uncertainties (e.g. climate, management). A local sensitivity analysis, at the vulnerability minimum, should be suitable. Once performed these simulations, we evaluate the results and update the response surfaces thanks to these new simulations. If we consider that the vulnerability minimum found is suitable, then we can stop and we have obtained a vulnerability assessment taking into account adaptation; otherwise, we restart this stage by searching a new vulnerability minimum.

The proposed approach is generic for all kinds of impact model. However, biogeochemical models used for climate change impact projections (for example, PaSim (Graux et al. 2011)) are usually time-consuming and require lots of inputs. That is why we previously explained that we will test and illustrate this approach with a “simpler and faster” model (i.e. the ModVege model (Jouven et al. 2006a,b)).
This model, relatively simple (about 20 equations) and fast (in computing time), is thus suitable for the purpose of assessing our approach.

**Vulnerability assessment without adaptation**

Different approaches can be chosen, depending on what is taken into account for sensitivity analysis. For instance, we could suppose that all forcing parameters (like vegetation traits) and variables (as climate data) are known with enough accuracy. Otherwise, we could account for uncertainties of one or many model-forcing components. In fact, we could define a whole set of approaches depending on released degrees of freedom. All these approaches are summed up into Table 1. To illustrate the different approaches, we will consider a simple case: a permanent pasture with a well known climate (30 years of weather data) and a known (and similar in time) management. We will only consider possible uncertainties on climate and management.

<table>
<thead>
<tr>
<th>Evaluation</th>
<th>Description</th>
<th>Design of Experiment</th>
</tr>
</thead>
<tbody>
<tr>
<td>“simple”</td>
<td>The coefficient of variability gave us sensitivity of vulnerability assessment.  No account for uncertainties.</td>
<td>1 site × 1 vegetation × 1 Environment × 1 Strategy of management</td>
</tr>
<tr>
<td>“climate uncertainties”</td>
<td>The coefficient of variability gave us sensitivity of vulnerability assessment. Climate uncertainties: correlation of years order is null or can be neglect in front of the appurtenance of one year to the considered period.</td>
<td>1 site × 1 vegetation × n2 Environments × Management (Bootstrapping without replacement on couple Management x Environment Year)</td>
</tr>
<tr>
<td>“uncertainties on climate extreme events frequency”</td>
<td>The sensitivity, in the vulnerability assessment with account for exposure (i.e. perturbations to which a system is exposed), is weighted by extreme events occurrence probabilities. Account for extreme events frequency uncertainties.</td>
<td>1 site × 1 vegetation × n3 Environments × Management, made of 30 Management x Environment years with eventually extreme years repetitions</td>
</tr>
<tr>
<td>“uncertainties on management”</td>
<td>The sensitivity, in the vulnerability assessment with account for exposure, is weighted by deviation form reference management occurrence probabilities. Account for management uncertainties.</td>
<td>1 site × 1 vegetation × 1 Environment × n4 Management strategies (composed of 30 management years, similar to reference with eventually a “delta”)</td>
</tr>
<tr>
<td>“uncertainties on management, climate and extreme events”</td>
<td>The sensitivity, in the vulnerability assessment with account for exposure, is weighted by occurrences probabilities for deviation to reference management and extreme events. Account for management, climate and extreme events frequency uncertainties.</td>
<td>1 site × 1 vegetation × n5 Environments × Management, composed of n5 Strategies of management and n5 Environments</td>
</tr>
</tbody>
</table>

**“Simple” evaluation**

The first case consists in simply simulating the system over 30 years. The coefficient of variability (CV) will be taken as measure of sensitivity, needed for vulnerability assessment. As every year is considered as equiprobable, CV will allow us to calculate vulnerability while also accounting for exposure to climate variability. This first approach does not account for any kind of uncertainties. The DOE model is simple (Figure 4), as each factor (element of the ModVege metamodel input) only appears once. This approach allows a “simple” evaluation of the climate change impacts and its associated vulnerability.

**Evaluation with account for climatic uncertainties**

In this case, we consider that climatic years are representative of a period, and that correlation between realizations at year N and year N-1 are negligible compared to the correlation of theses years and their membership to the period. So, we could consider, that the order of occurrence of individual years is random. This is without any consideration for the fact that atmospheric CO₂ increases over year. We could generate them by bootstrapping (sampling without replacement, e.g. Efron 1993), thus keeping climate realizations paired with management practices. As before, coefficient of variability can be used to evaluate sensitivity in the vulnerability assessment.

The resulting DOE (Figure 5) is, in this case, composed of fixed factors (Site and Vegetation) and of many Management strategies and Environments, which are composed of 30 realizations (without repetitions) of respectively Management Year and Environment Year. Due to the uniqueness of the constraint defined by the pair Management x Environment year, the latter has only 30 occurrences. The interest of this design is that it only takes into account part of the climate uncertainties.
Indeed, Management strategies and Environments are still composed of 30 years but some years can be repeated. In this DOE, we must differentiate “Normal” years from “Extreme” years, which can replace normal years. The interest of this DOE is that it accounts for uncertainties on EE frequency.

Evaluation with account for management uncertainties

Another possibility could be to consider that the management could have been slightly different (for example, mowing dates slightly altered). In this case, we will use an experimental design testing the gap between new and initial dates. Thus, the first cutting date of the first year can be moved one week earlier, whereas the first cutting date of the second year is three days later, and so on. For the vulnerability assessment, we can estimate vulnerability, while accounting for exposure, by considering that the gap of days follows a known distribution (Gaussian for example). The DOE (Figure 7) is this time made of three fixed components (Site, Vegetation, Environment) and a set of Management strategies, which are in turn made of 30 Management years in which we can distinguish years similar to observation (Base) and those with a modification (With Delta). This vulnerability assessment DOE allows accounting for management uncertainties.

Figure 6. DOE model for vulnerability assessment under climate change when accounting for uncertainties on extreme climate event frequency

The DOE (Figure 6) is made of two fixed elements: Site and Vegetation. The main differences with the previous DOE (Figure 5) are the cardinalities of yearly realizations.

Evaluation with uncertainties on management and climate (including climate extremes)

To take into account all uncertainties and having a sensitivity evaluation, we could combine the three previous approaches. The main issue is the DOE size. A Latin Hypercube Design could be a suitable way to reduce the number of needed simulations (McKay et al. 1979).

The aim of this DOE (Figure 8) is to account for the different uncertainties sources (only the most probable, we have excluded Site and Vegetation) and to bring a good a priori knowledge for the response surface and for vulnerability assessment. In this DOE, we find a combination of Management Strategy and Environment. These are made of 30 elements, possibly with repetitions.
Indeed, all Agro-ecological models use a set of Inputs, in which we can distinguish two types. On one hand, the Decision inputs are those who help applying a management strategy impacting results, and which must be optimised in function of the study criteria. On the other hand, the Environment inputs only bring uncertainties.

This distinction between these two types of inputs corresponds to Taguchi’s approach (Taguchi 1987). This classification is helpful to propose a metamodel in the MDE context (Bézivin 2005), which all agro-ecological model inputs would be conform to. We could also notice that there is a relationship linking the Inputs. This information is contained in the Constraint class, and it could also be specified mathematically:

“Let \( f_T : D \rightarrow D' \) be the characteristic function of a type of experimental design \( T \), where \( D = D_1 \times D_2 \times \ldots \times D_N \) is the domain of definition of the \( N \) inputs and \( D' = D'_1 \times D'_2 \times \ldots \times D'_{N'} \) the target domain of the function \( f_T \). \( D' \) contains the set of factor level combinations ranged by the experimental design \( T \).”

Figure 8. UML DOE Model for vulnerability assessment under climate change, accounting for uncertainties on management, climate including extreme events

**Evaluation of vulnerability minimum**

Once response surfaces will be fitted, a vulnerability minimum will be searched for. But, in fact, what we are really looking for is more a robust solution to reduce vulnerability than an optimal solution by a mathematically-suited fit. Indeed, due to the high environment uncertainties level, an optimal solution could completely fail (Kleijnen et al. 2005). Robust solutions should be in fact satisfactory with respect to vulnerability. For example, a robust solution should provide high grassland yield with low sensitivity to environment (that is, high stability); it should have by the way low vulnerability. An example of robust design, inspired by Taguchi (1987) can be found in Sanchez et al. (1996) and Kleijen and Gaury (2003). The evaluation of the found vulnerability will have two aims: firstly, we have to check that we really found a minimum of vulnerability and, secondly, we need to perform sensitivity analysis to evaluate the robustness of the solution. Finally, the DOE needed for this step will be similar to the DOE previously described (Figure 8).

**METAMODEL OF IMPACT MODELS WITH ASSOCIATED DOE FOR VULNERABILITY ASSESSMENT**

Figure 9. UML metamodel of agro-ecological model

Considering the DOE models we have previously proposed, we can identify a common pattern (Figure 9 and Figure 10).

Figure 10. Instance diagram corresponding to a UML metamodel of DOE associated to an agro-ecological model. All DOE are compositions of Input instances, forming itself a set (class Input \( 1 \times 2 \times \ldots \times N \)).

Moreover, all DOE are the instance composition of Inputs (element 1, element 2 ... element \( N \)). Those inputs can be combined (depending on the type of experimental design \( T \)) to create a set of inputs (Input \( 1 \times 2 \times \ldots \times N \)). All these inputs are constrained by instances of Constraint classes. They could be specified more accurately thanks to the OCL (“Object Constraint Language”, OMG 2010). OCL is a formal language to specify constraints between elements of UML graphs. The resulting cardinality (\( \text{card}_T \)) of the set of inputs is a direct function of the DOE type. For example, if the design is a full factorial design then:

\[
\text{\text{card}}_T = \prod_{i=1}^{N} (\text{\text{card}}_i)
\]

Where \( \text{\text{card}}_i \) is the cardinality of input \( i \) instance definition domain; we could notice that, in this special case, \( f_T \) is in fact the Identity function. If the design is a “One factor At a Time” (Kleijnen 1987), then:

\[
\text{\text{card}}_T = 1 + \sum_{i=1}^{N} (\text{\text{card}}_i, -1)
\]
All DOE are conform to that representation.

All the examples of DOE we have presented are suitable for deterministic models. In the case of stochastic models, the approach will stay almost the same. A slight difference exists to add the replications of simulations in the DoE. It will also be necessary to account for noise into the response surface. Still in the case of stochastic DoE, it is interesting to specify the pseudo-random number generator in use and also to vary this generator among a set of reliable ones (Mersenne Twister series, WELLS and the latest MRGs from L’Ecuyer) (Hill 2010). The initialization and parallelization techniques of the random number generator do also have to be specified in high quality DOE.

DISCUSSION AND POSSIBLE SOLUTIONS

Vulnerability assessment is based on sensitivity analysis. Depending on the chosen definition, it can include the notion of exposure (by, for example, weighting years with extreme events). But it could also account for adaptations (limited by its real capacity). Identification and evaluation of adaptation options need a lot of simulations. In order to reduce this number (and mainly the computation time), we intend to use response surfaces instead of real simulations, for vulnerability minimum searching. Indeed, response surface can be used as model of simulations with a rather good accuracy and a faster computation time. Whatever the chosen definition and method, this requires a large number of simulations. Facing this issue, the choice of pertinent experimental designs appears as much critical as the simulation distribution question. Indeed, a suitable DOE will allow us to reduce the number of experiments as much as possible, whereas distribution to multiple computing platforms (i.e. clusters, grid, cloud computing) will reduce the waiting time to get all the results. We have already tackled the design of such software tools, which provide distributed computation and platform independent DOE (Amblard et al, 2003; Reuillon et al, 2008; Reuillon et al, 2010). However, theses tools do not take into account vulnerability assessments, in their actual state. To adapt them, we will need metamodels of the agro-ecological model and of the associated design of experiment.

CONCLUSION

Many simulations are necessary to perform a vulnerability assessment. We have defined different designs depending on the uncertainty levels that we wish to account for. For each of these designs, we have proposed a model. The study of the obtained models enabled us to go one step further with the proposition of a metamodel of agro-ecological models with their associated DOE. Each of the previously proposed models is conform to this metamodel. This work, following a reverse engineering experience (Lardy et al. 2011), can be seen as preliminary to build up a dedicated software framework for vulnerability assessment under climate change scenarios. This framework will tackle distribution of constrained experimental designs. We consider that a Model Driven Engineering approach will help us in the design and production of our future framework. The different models, according to the released degrees of freedom, and the associated metamodels presented in this paper enable establishing the first step towards the design of a generic tool for climate change vulnerability assessment.

ACKNOWLEDGEMENTS

This work was jointly supported by the EC FP7 ‘GHG-Europe IP’ and the FP7 EU ‘CARBO-Extreme’ project.

REFERENCES


Jacob, M. Steen, M. Heerink, L. 2008 “Reusable Model Transformation Patterns”. Workshop on Models and

**BIOGRAFIE**

**ROMAIN LARDY** is a PhD student jointly at the LIMOS (ISIMA) - UMR CNRS 6158 of Blaise Pascal University of Clermont Ferrand (France), and at the UREP in the French National Institute for Agricultural Research (INRA). His work is focused on High Performance Vulnerability Assessment with constrained Design of Experiments using a model driven engineering approach, applied to climate change vulnerability of grassland / livestock systems.

**GIANNI BELLOCCHI** is a senior scientist at the Grassland Ecosystem Research Unit (UREP), of the French National Institute for Agricultural Research (INRA) at Clermont-Ferrand. He has co-authored many scientific papers related to agronomy, plant physiology and climatology. His current research aims at enhancing the use of modelling tools to analyze grassland/livestock systems under climate change.

**BRUNO BACHELET** is Associate Professor at Blaise Pascal University. He received a Ph.D. degree from Blaise Pascal University in 2003. His current work is focused on software engineering techniques for scientific computing, mainly for operations research applications and simulation models.

**DAVID HILL** is currently Vice President and Professor at Blaise Pascal University. Since 1990, Professor Hill has authored or co-authored many technical papers and journal papers and he has published various text books including recent free e-books from Blaise Pascal University Press (http://www.isima.fr/~hil).
MEDICAL SIMULATION
SPATIAL SIMULATION OF A(H1N1) INFLUENZA VIRUS SPREADING
COMPARISON WITH REGIONAL DATA

Luc Touraille1, Amandine Letagneaux3, Damien Mouly3, Alain Leroy4, Delphine Bas12, Julien Keruzec12, Claude Mazel12, Alexandre Muzy5, David R.C. Hill12

1 Clermont Université, Université Blaise Pascal, ISIMA / LIMOS, BP 10448, F-63000 CLERMONT-FERRAND
2 CNRS, UMR 6158, ISIMA / LIMOS, F-63173 AUBIERE
3 InVS – CIRE Auvergne, 60 avenue de l’Union Soviétique, 63057 Clermont-Ferrand cedex
4 ARS Auvergne, 60 avenue de l’union soviétique, 63057 Clermont Ferrand Cedex, France
5 LISA, CNRS, Università di Corsica Pasquale Paoli, Campus Grossetti, 20250 Corti, France

luc.touraille@isima.fr, amandine.letagneaux@ars.sante.fr, damien.mouly@ars.sante.fr, david.hill@isima.fr

KEYWORDS
Pandemic, Influenza, Epidemic model, H1N1, Simulation

ABSTRACT

An influenza epidemic/pandemic model and the corresponding simulation tool were developed by a team of computer scientists and epidemiologists. The aim is to obtain a set of indicators for regional authorities, such as the number of infected individuals or the number of hospitalizations. The evolution of the epidemic/pandemic is modelled week by week and area by area. The resulting software can be used by decision-makers to experiment different scenarios and to observe their consequences on the health system. Based on the outcomes of the model, they can take the necessary measures to reallocate medical resources in the different areas to avoid the overload of the system. To validate the model, we compared its results against the data gathered in Auvergne (France) during the 2009 A(H1N1) pandemic.

INTRODUCTION

Every year, seasonal influenza epidemics provoke 250 000 to 500 000 deaths and up to 5 million cases of severe illness, according to the World Health Organization. In France, surveillance systems estimate that nearly 2.1 million cases of clinical influenza are affected every winter during epidemic wave (InVS 2008). Punctually, influenza pandemics occur, causing yet more damages – three occurred during the 20th century. In 2009, the outbreak of the A(H1N1) flu showed that mutating virus can quickly become transmissible between humans, provoking unexpected waves of infections. Fortunately the 2009 A(H1N1) epidemic was not severe, but in such cases of emergency, decision-makers need tools to help them choose the appropriate response. In these situations where real experiments are difficult if not impossible to perform, simulation is a powerful tool to explore possible scenarios (Kleijnen and Groenendaal 1992; Hill 1996).

Since the 2004 spread of the avian influenza strain H5N1 in Asia, many models have been developed in regard to this virus (Longini et al. 2004, 2005) (Ferguson et al. 2005) (Colizza et al. 2007) (Das et al. 2008) (Iwami et al. 2009). The contribution of this type of simulation models to public health decisions have been studied in (Bonmarin and Levy-Bruhl 2007).

In 2008, we started the implementation of simulation software to model the spread of H5N1 in Corsica (Hill et al. 2008). We gathered a multi-disciplinary team composed of epidemiologists and computer scientists to design an individual-based model that could produce interesting indicators for the health system. The result was an application that combined a Geographical Information System and a Multi-Agent System using a small-world like communication network (Hill et al. 2009). Other agent-based models have been proposed in the past. In (Deguchi et al. 2007), the authors model the social activities of agents to see their impact on the epidemic dynamics. In (Carrat et al. 2006), contacts among a community are modelled through randomly generated graphs, to test the effect of several measures such as treatment or vaccination. More recently, (Kleczkowski and Maharaj 2010) presented some work in progress where they study the effectiveness of two controls (washing hands and staying at home) to limit the spread of a virus, using a cellular automata model.

During the 2009 H1N1 pandemic, epidemiologists from the French Institute for Public health Surveillance (InVS) proposed to the French Ministry of Health a set of parameters that could be used to model an influenza epidemic/pandemic (InVS–CIRE 2009). Our team, composed of epidemiologists from the CIRE Auvergne (Regional Office of the InVS) and computer scientists from the LIMOS CNRS Laboratory (Laboratoire d’Informatique, de Modélisation et d’Optimisation des Systèmes – System Modelling and Optimization Computer Science Laboratory), implemented this model in simulation software to provide an easy to use tool for scenarios exploration.

The model we present in this paper is a deterministic model based on simple equations. Its aim is not to perform exploratory research on epidemics dynamics but rather to produce quite predictive results accurate enough to serve as a basis for decision-making. In case of an outbreak, the authorities can use our simulator to test different scenarios and observe their consequences on a regional health system, week by week and area by area; with the indicators produced by the model, they can take measures to prevent the lack of medical resources in the areas where they are too scarce to handle correctly the apparition of new cases.
MODELS IN EPIDEMIOLOGY

Compartmental models were first studied in (Kermack 1927). In such models, the population under study is divided into groups. Groups are called compartments which are determined according to their disease sensitivity. The most common compartmental model is called SIR (Susceptible, Infectious, Recovered). The population is divided in three compartments: people who are susceptible to be infected, people who are infected, and people who recovered and are immunized. Dynamically, people from one compartment can die or become part of the next compartment (e.g., a person achieving all states will start from susceptible, then become infected and finally turn recovered). Usually, the number of people in each compartment is calculated using differential equations (∑, with S the number of people in susceptible state, etc.). Compartmental models are often studied mathematically (Li et al. 2011).

Many extensions to the basic compartmental model have been proposed:

- to include temporary immunity (Kyrychko 2005),
- to include vaccination effects (Brauer 2008),
- to add as many parameters as required (René 2007).

The main limit of compartmental model (Bowong 2010) concerns the addition of parameters. Because of their differential-equation structure, adding parameters leads to intractable equations.

Other approaches based on discrete models can be categorized into computational models. Various kinds of models compose this category. One kind concerns cellular automata. Interactions between people and their characteristics are considered to be homogeneous. A comparative study of SIR-type models including usual mathematical models and cellular automata can be found in (Schneckenreither 2008). Another type of computational models concerns Multi-agent Systems (Dion 2011). Interactions and characteristics of people need not to be homogeneous. Using an agent-based description, each person can be described through his own characteristics (age, vaccinated, etc.) and interactions between different persons can be different in time.

Our model is discrete; therefore, it is well suited for simulation and can easily integrate as many parameters as required (contrarily to differential equation models, which could become intractable).

When compared to usual compartmental and computational models, our model presents several differences. First of all, our goal here is not to define an exhaustive system for decision support in case of epidemics. Complete methodologies and software are rarely used by authorities because they are too specific and/or too complex and/or too dependent on experimental data availability. This is why our goal is to define a simple abstract model, which aims to be as close as possible to the usual parameters used by authorities to face and anticipate epidemics.

Secondly, interactions with authorities are not abstracted here into a methodology. However, due to the simplicity and closeness of the model to real cases, it is expected through this model to facilitate further discussions between modellers and authorities.

SOFTWARE SUITE ARCHITECTURE

In order to enhance the flexibility and reusability of our simulation tool, we designed it with a very low coupling between its different components, as shown in Figure 1. In fact, we developed a suite of three applications completely independent, whose compatibility relies only on a common format for representing simulation results.

The main application is the simulator itself, named JGrippeSim (for Java Flu Simulator in French), which is parameterized either through an XML input file or through the simple GUI presented in Figure 3. These parameters,
described in the next section, model the main characteristics of an epidemic/pandemic disease, e.g. the attack rate or the hospitalization factor. The simulator also takes as input a population file in CSV format; this file contains the list of the areas to be considered in the simulation, along with their population and their number of general practitioners. The type of areas depends on the scale selected for the simulation: it could be rural districts, regions or even states.

The simulator produces a result file in CSV format that can be directly exploited by spreadsheets or used as input in the two proposed analysis tools:

- a map generator (JGrippeCarte), which creates a visual representation of the results through several coloured maps;
- a report generator (JGrippeRapport), which compiles the results in a well-presented document with tables and figures, in a variety of formats (PDF, HTML, ODT, etc.).

The design of these two additional applications is beyond the scope of this paper. However, it is interesting to detail the simulator slightly more in depth since it is the core of our work.

The classes of the simulation software are divided in three packages. The data package contains all the classes responsible for storing the data needed and generated by the simulator, as well as all the I/O facilities such as the serialization of the parameters in XML or the writing of the results in CSV. The simulator package contains the simulator itself, representing our flu model, and a simulation manager making the connection between the simulator and the other packages. Finally, the gui package contains the code for displaying the maps and the user interface shown in Figure 2 and Figure 3.

In Figure 3, we see that the GUI lets the user fill in or load the parameters of the model (“Configuration loaded” checkbox) and load the population file (“Population loaded” checkbox). The application also supports command-line execution in order to facilitate the creation of batch simulations.

**OVERVIEW OF THE EPIDEMIC DISEASE MODEL**

As we said previously, the model used in our simulator is a rather simple deterministic model aiming at exhibiting some predictive capabilities to help decision-makers in emergency situations. It is based on a proposition made by InVS where they identified a set of parameters for estimating the impact of the 2009 A(H1N1) pandemic on the health system (InVS-CIRE 2009). Parameters values have been fixed from both regional surveillance data and bibliographical sources.

We made the hypothesis of a two-wave pandemic, each wave lasting eight weeks (it is possible to simulate a single wave pandemic by fixing the second wave attack rate to zero). The parameters we retained are the following:

![Figure 2: Map of the Auvergne Region with a Hundred Different Medical Areas](image)

![Figure 3: Graphical User Interface Used to Input the Simulator Parameters](image)
Two attack rates, one for each wave. The attack rate represents the ratio between the number of infected individuals and the number of exposed individuals (in our case, the entire population).

The distribution of the cases between four age groups. Population is divided according to four age ranges: 0 to 4 years old, 5 to 15 years old, 16 to 64 years old and over 65 years old. The distribution of infected individuals in these age groups is usually not homogeneous – due to the weaker resistance of some people to certain diseases.

The hospitalization rate. It represents the number of hospitalized individuals for 100,000 inhabitants. Even if the number of hospitalizations related to the number of infections is more intuitive, this rate depends on the quality of surveillance systems for cases of infections which can be different by country. Surveillance is probably more accurate for hospitalizations cases than for infections cases. The rate of hospitalizations related to inhabitant is independent of these possible variations and is also easier to obtain than the rate among infected individuals.

The distribution of the hospitalizations by age groups.

The rate of acceptance in Intensive Care Unit (ICU), i.e., the proportion of hospitalized individuals that need to be transferred to ICU.

The distribution of the individual in ICU by age groups.

Besides these parameters related to the disease virulence, we also included the possibility to model a vaccination campaign through a vaccination rate (ratio of individuals being vaccinated among the population) and a probability of success (probability that the vaccine prevents an individual from contracting the disease). Figure 4 below presents a possible design of experiments, showing a response surface for 36 different simulations where the vaccination rate is varying from 10% to 90% (9 levels for this factor with a 10% step) and four different attack rates (2nd factor).

Finally, in order to study the stress applied on the health system by the epidemic, the model is parameterized by a consultation rate (ratio of infected individuals consulting a general practitioner) and a mean number of achievable consultations by a practitioner during a week.

In addition to the parameters described above, the simulator takes as input a population data file. This file contains a list of the areas of interest with, for each of them, its population (with age distribution) and its number of general practitioners.

Based on these inputs, the simulator computes a set of results for each area and each week of the pandemic. First, the number of infected individuals is computed from the attack rate and the population of each area. In Figure 5, we see a comparison of two scenarios on the Auvergne region with 2 attack rates: 3.8% and 8%.

Figure 5: Number of Infected Individuals. Comparison of 2 Attack Rates over 8 Weeks

The evolution of an epidemic wave usually follows the same trend: a rapid increase of the cases to a peak followed by a more gradual decrease until the end of the epidemic. To model this evolution, we distribute the apparition of new cases according to a normal distribution, resulting in a Gaussian shape that gives a satisfactory approximation. The infected individuals are then distributed in the different age groups according to the distribution parameters.

When contracting a disease, some people do not consult a doctor while others consult several times. To take this into account, we weight the number of infected individuals by a consultation factor, which can be greater than 1 if people tend to consult more than once. Doing so, we obtain the number of consultations in each age group, area, and week (Figure 6). This quantity is then compared to the number of consultations practitioners usually perform in one week. The resulting consultation margin gives important information to the decision-makers regarding the stress applied to the health system: a consultation margin below zero means that there will not be enough doctors to achieve the necessary number of consultations (at least without doing extra hours). Based on that, the authorities could decide a reorganization of the medical system to better distribute the medical human resources.
A third type of information is computed by the simulator, regarding the activity of hospitals. Each week, a fraction of the population becomes hospitalized according to the hospitalization rate. The cases are distributed in the different age groups. Among the hospitalized individuals, a portion of them (determined by the rate of acceptance in ICU) needs more intensive care and is transferred to ICU.

To sum up, here is the list of the outputs computed by the model:

- Number of infections per area, week and age group.
- Number of consultations per area, week and age group.
- Number of hospitalizations per area, week and age group.
- Number of transfers in ICU per area, week and age group.
- Consultation margin per area and week.

The results also include several sums to facilitate their analysis, e.g. the totals of the presented quantities over all the areas.

MODEL VALIDATION WITH AUVERGNE DATA

Now that the epidemiological characteristics of the A(H1N1)_{2009} influenza are known, it is possible to compare the model results with real data. To do so, we determined a set of parameters based on bibliographical sources and data collected by surveillance organizations in Auvergne. In addition to this, the Health Regional Agency of Auvergne area (ARS Auvergne) provided us data about the allocation of the medical resources in this region.

Parameters determination

The attack rate for the first wave in Auvergne was estimated in (Letagneaux et al. 2010) to be 3.8% of the population. This estimation is based on general practitioners consultations. Though in this real case there was no second wave, in our study the second attack rate was fixed arbitrarily in the model to 8% (approximately the double of the first wave). The repartition of the cases among the different age groups was calculated from data emanating from general practitioners (Letagneaux et al. 2010). On average, over the A(H1N1) epidemic, 8% of the infected individuals were between 0 and 4 years old, 28% between 5 and 15 years old, 61% between 16 and 64 years old, and 3% were 65 and over.

The hospitalization rate is based on data collected from 13 general hospitals of Auvergne. During the epidemic period, response rate of hospitals was satisfactory (94% of indicators were completed every day). In our study, 230 hospitalizations due to A(H1N1) infection were reported, which represent a proportion of 17.2 per 100 000 inhabitants of the Auvergne population. The repartition of the hospitalizations among the age groups was fixed as follows, according to the estimation proposed by the French CIREs (InVS–CIRE 2009): 15% between 0 and 4, 15% between 5 and 15, 60% between 16 and 64 and 10% over 65.

The rate of acceptance in ICU is 9% of the hospitalized individuals, once again according to data reported by general hospitals. The repartition among age groups according to cases observed during epidemic period in Auvergne area (InVS–CIRE 2009) was 7%-9%-77%-7%.

As the attack rate was determined based on the outcome of the epidemic, we did not include any vaccination in the simulation. Indeed, in our case study, the vaccination rate was very low (below 10% in 2009) and the effect of the vaccination campaign has been taken into account in the attack rate; consequently, adding another reduction of the cases would have led to an underestimation of the infections. The consultation factor was fixed to 0.98. Finally, the number of consultations achievable by general practitioners was set to 97 consultations per week and per doctor, according to (Evraud 2002).

Comparison Between Model Results and Actual Data

The number of consultations provided by the application (52,626) is very close to the one actually observed during the A(H1N1)_{2009} epidemic (53,075) (a small error of 0.85%). Figure 7 shows the number of consultations estimated by the model and actually observed.

![Number of Consultations Estimated by the Model and Actually Observed (Total and per Age Groups)](image-url)
However, the weekly evolution of the consultations per age groups differs significantly from the reality. Indeed, the evolution in the model is very homogeneous, whereas the actual data exhibits great variations.

Regarding the temporal evolution of the total number of consultations, Figure 8 shows that both the model and the observed data demonstrate an epidemic peak at the fourth week. On the one hand, the model curve is a Gaussian centred on week four and distributed over 8 weeks. On the other hand, we see on the actual data curve that the number of consultations decreases after the peak more slowly than in the model. Differences between predicted and observed cases could partly be explained by several factors such as human behaviours (i.e individual hygienic protections) or environmental influence, which are not taken into account in model.

![Figure 8: Temporal Evolution of the Number of Consultations](image)

The number of hospitalizations estimated by the model adds up to 224 cases; the data reported by the general hospitals mentions 230 cases (2.68% error). The temporal repartition of the hospitalizations by the model exhibits the discrepancy we previously observed with consultations: during the first weeks, the model overestimates the number of cases, and underestimates them after the peak, as can be seen in Figure 9.

![Figure 9: Temporal Evolution of the Number of Hospitalizations](image)

Finally, regarding ICU cases, the model once again provides a satisfactory estimation of the total number of cases (20 estimated versus 21 observed), but fails to approximate the temporal evolution. Indeed, as Figure 10 shows, the shape of the distribution of ICU cases per weeks in Auvergne during the 2009 pandemic is unlike the wave-like shape characteristic of epidemics. This is easily explained by the small size of the sample (about twenty individuals).

![Figure 10: Temporal Evolution of the Number of Acceptances in Intensive Care Unit](image)

**CONCLUSION**

In this paper we have presented decision support software dedicated to predict the regional impact of an influenza pandemic. In addition to sophisticated models that we previously proposed for research around pandemic (Hill et al. 2008, 2009), we have based this decision support system on simple deterministic equations using the main parameters which influence the propagation of an epidemic. The model simplicity is linked to predictive capacities enabling local authorities to estimate the main figures of a pandemic with regional data spatially distributed. Indeed, the model is able to give, weekly, for a set of geographic areas and by age group: the number of geographic areas and by age group: the number of infected individuals, the number of consultations, the number of hospitalizations per area, the number of transfers in ICU and the consultation margin per area and per weeks according to the local medical resources. The scale of geographic information has to be adapted to health facilities suitable for the medical management of cases. Thus, estimated number of consultations will need to be accurate at the municipality or intra-municipality level, while a county resolution seems sufficient for the number of severe cases that require treatment by a intensive care unit.

The current model is inspired by an InVS proposition based on a rough Gaussian model to represent epidemic waves over 8 weeks. This simple approach has been satisfactorily tested again the 2009-2010 data of A(H1N1) pandemic in the Auvergne (France) region. Though this type of model is limited, overall trends of the temporal evolution are overall correct. To obtain more statistical relevance, more data would have been required and the underlying model distribution should have been more flexible. Empirical distribution of epidemic waves during more than 8 weeks could be obtained by gathering and collecting data from historical epidemics and pandemics. Moreover, the epidemic parameters implemented in the software are estimated from observed data. They depend on types of virus responsible for the epidemic, of barrier
measures implemented to fight against the epidemic, and of the way monitoring data is collected. The sources of information available when an epidemic starts are often situated in the southern hemisphere, where collection data may be different from the one in France. Observations after the epidemic are therefore more or less offset by the forecast. In the case of the pandemic 2009-2010, health impact has been finally lighter than the announced forecasts (InVS 2010). However, the modelling tools, even if imprecise, have proved useful in preparing the response to this type of event.

Predictive capacities of the model are satisfactory during the first 4 weeks of the epidemic (i.e. from start to the peak). Consequently, this tool should allow Health Authorities to implement preventive actions for the spread of the epidemic and the potential impact on the organization of medical care. To better take into account the spatial dimension of the epidemic, the cartographic output module will need to be further developed.

REFERENCES


Carrat F., Luong J., Lao H., Sallé A., Lajaunie C., Wackernagel H., 2006, “A small-world-like model for comparing interventions aimed at preventing and controlling influenza pandemics”. BMC Medicine, 4, n°26. An electronic version of this article can be found online at: www.biomedcentral.com/1741-7015/4/26


Latest advances in the simulation of the removal of blood clots by a newly developed mechanical thrombectomy device.

Romero, G.\textsuperscript{a}; Moreno, I.\textsuperscript{a}; Martínez, M.L.\textsuperscript{a}; Pearce, G.\textsuperscript{b}; Perkinson, N.D.\textsuperscript{b}

\textsuperscript{a} Graphic Engineering and Simulation Group, ETSI Industriales, Universidad Politécnica de Madrid, Madrid, Spain
e-mail: \{gregorio.romero, isabel.moreno, luisa.mtzmuneta\}@upm.es

\textsuperscript{b} Biomedical Sciences Division, Wolverhampton University, Wolverhampton, United Kingdom
e-mail: gillpearce@gmail.com, neil@perkinson.org.uk

KEYWORDS
Biomedical engineering, Thrombus Aspiration Device, Blood clot, Adhesion forces, Simulation techniques, Bond Graph, Strokes.

ABSTRACT

In this work, we present the analysis, design and optimization of one experimental device recently developed in the UK, called the “GP” Thrombus Aspiration Device (GPTAD), which has been designed to remove blood clots without the need to make contact with the clot itself, thereby potentially reducing the risk of problems such as downstream embolisation. To obtain the minimum pressure necessary to extract the clot and to optimize the device, it has been necessary to simulate the performance of the GPTAD analyzing the resistances, compliances and inertances effects. We model a range of diameters for the GPTAD considering different lengths and percentages of occlusion. In each case we determine the optimum pressure required to extract the blood clot from the artery using the GPTAD, which is attached at its proximal end to a suction pump.

INTRODUCTION

The World Health Organization reports that 15 million people worldwide suffer stroke; and of these, 5 million die and a further 5 million are left permanently disabled, many severely impaired. Consequently stroke is a major cause of mortality world-wide. Most strokes are caused by a blood clot that occludes an artery in the cerebral circulation. Thrombolytic agents such as Alteplase are used to dissolve blood clots that arise in the cerebral arteries of the brain but there are limitations on the use. Recently screening for patients at risk of strokes and TIA’s has come into being. If such plaques are detected in the carotid arteries (by Ultrasound), a Carotid endarterectomy (CEA) - a surgical operation - may be performed to remove the occlusive plaque. Additionally, stenting has been investigated for partially occluded carotid arteries. In the SAPPHIRE study [O’Riordan 2004], it was concluded that carotid stenting lowered the incidences of major stroke and myocardial infarction, in patients at high risk who needed surgery.

Over the past decade, other methods of treatment have been developed which include Thrombectomy Devices e.g. the ‘GP’ Thrombus Aspiration Device (‘GP’ TAD). Such devices have the potential to be used as an alternative to thrombolytic agents or in conjunction with them to extract clots in the different arteries e.g. in the middle cerebral artery of the brain, carotid, popliteal artery, etc. Alternatively a clot of blood may become attached to the plaque and subsequently become detached and pass into the cerebral circulation giving rise to a stroke. In the case of 100% occlusion, it causes total blockage of the artery.

In this work, we present an analysis and modelling of the GPTAD and show it to be a highly effective method of simulating the device under a variety of conditions of percentage occlusions. We take into account factors such as the resistances, compliances and inertances effects in the modelling. Such modelling is useful in optimizing the GPTAD and predicting the result of clot extraction under a variety of conditions (e.g. different extents of arterial occlusion). The model includes the systolic and diastolic blood pressure variations too. The aim of this simulation model is to obtain the minimum and the maximum pressure necessary to extract the clot and to verify that, both the pressure and the time required to complete the clot extraction are realistic for use in clinical situations, and are consistent with any experimentally obtained data. Previous models have been analyzed by the authors in other papers, but this model includes a more realistic blood clot – wall adhesion force, introducing and analyzing the different and necessary parameters and changes into the model to make it more realistic.

MODEL DESCRIPTION

In recent times the mechanical thrombectomy devices have become increasingly involved in blood clot removal. The “GP” Thrombus Aspiration Device, is under development and was invented by Pearce and Perkinson (Pearce et al. 2007).

Figure 1. ‘GP’ Device.
It has potential advantages in clot removal procedures in that it does not need to touch the clot to facilitate clot removal (therefore there is less risk of downstream embolisation), and it has no moving parts (therefore less risk of breaking). It also has low forces on the periphery of the device, so that there is less risk of arterial wall collapse.

A pump attached to the other end of the catheter that contains the GPTAD provides the necessary suction pressure for the operation, joined to a very long catheter. The ‘GP’ is located at the end of this catheter. It is a hollow cylinder with the same diameter as the catheter. Its interior is the place where the vortex is created to carry out the extraction. This device is introduced into the cerebral artery at the place where the clot is, and is positioned at a distance of 3mm from it. Then the suction begins until the clot is extracted. The clot crosses the 3mm that separates it from the ‘GP’ TAD and the clot is captured in the ‘GP’ TAD. Once this capture of the clot has occurred the device containing the clot is removed from the body.

**MODELLING THE GPTAD DEVICE**

The objective of this study is to introduce a new model that can be used investigate and assist in the final design of the ‘GPTAD’. We investigate the potential performance of the device under different conditions of blood flow, size of blood clot and systolic-diastolic pressure, in a given vessel. The method chosen for the representation and simulation in this instance is the Bond Graph technique (Karnopp et al. 1990, Zadpoor et al. 2005), which allows assimilating the model to an electric circuit made up of inductances, capacitances and resistances. Therefore, it is possible to obtain the results in a simple way by evaluating flows and efforts that join and connect the components of the model.

Consequently, we needed to analyze the inertances, compressibilities and resistances that the system is subjected to.

Previous models developed by the authors about the ‘GPTAD’ device (Romero et al. 2010, Romero et al. 2010, Romero et al. 2011) have shown good results but some simplifications concerning the interaction inside the blood clot and with the artery wall was considered.

One of the most important improvements in this model is related to the friction between the clot and the arterial wall, which creates another resistance factor. The value of this parameter must be variable depending on whether the clot has begun its movement (dynamic friction) or before it has begun to move (static friction) during the clot extraction procedure. In the presented work, this value is obtained from the interaction platelet - artery wall for the static friction and with Stokes for the dynamic friction (which decreases considerably); the transition between both values marks the beginning or the end of the clot movement.

In our work, the modelling of the 70% occlusion is compared to the results at 80% and 100% occlusions to highlight differences between these states of occlusion. In those cases the remainder of the occlusion is made by the atherosclerotic plaque.

Finally, into the model developed in this work we have included the systolic and diastolic blood pressure (120/80 mmHg) as additional positive pressure adding a variable pressure source that pushes the clot in sympathy with to the rhythm of the heart.

**GPTAD MODEL**

To generate the correct model as for the case of the previous models, firstly it is necessary to analyze the inertances due to the mass of fluid, the compressibility that the blood and artery are subjected to, and finally the resistances that appear when fluid and blood clot flow into the catheter (Margolis 1979).

Firstly, the pump, which is the component that creates the necessary pressure to carry out the extraction, is positioned. It can be represented by a variable pressure source (Se) whose value will increase from zero to a non-determined value suitable for carrying out this operation. The time taken to reach the maximum value of pressure has been considered to be 1 sec., after which time the pressure provided by the pump remains constant.

The pump links to the catheter, a 110 cm long, hollow cylindrical tube which is joined to the ‘GPTAD’ device of length 20 mm, and both with the same diameter. In order to represent both elements, they are considered as several pipe sections bearing in mind the different phenomena that take place in their interior: load and inertia loss, and fluid compressibility (Margolis 1979).

Firstly, the flow inertia to be overcome in its movement is taken into account and considering a section with circular geometry it must be represented by a type ‘I’ port and a type ‘I’ junction and can be modelled with this expression:

$$I = \frac{\rho \cdot L}{\pi \cdot \left(\frac{D}{2}\right)^2}$$

(1)

where ‘ρ’ is the blood density, ‘L’ the length of the pipe section and ‘D’ its diameter.

The blood has been deemed to be an incompressible fluid and this eliminates the blood compressibility effect.

Due to the friction between the liquid particles and the pipe walls, load losses appear. Since we are dealing with straight, horizontal and constant cross section pipes in our model, only linear load losses are taken into account. So the load loss is reduced to being progressive and proportional to the length of the pipe. This pressure loss can be represented by a resistance and a type ‘1’ junction. If we assume that when an artery is 100% occluded there is no flow, and since the Reynolds number is only significant if turbulent flow is involved, the equation that governs its behaviour can be determined from expression (2), where ‘η’ is the dynamic viscosity of the blood flow, ‘L’ the length of the pipe section and ‘D’ its diameter.
\[ R = \frac{128\eta L}{\pi D^4} \]  

(2)

The flow of blood may well be turbulent at an arterial bifurcation for example. In this case it would be necessary to determine the load losses for turbulent flow and taking into account the Reynolds number.

Following on from the tube corresponding with the catheter, the ‘GPTAD’ device (see figure 1) must be positioned such that it can be represented with its corresponding values.

Due to the artery being located at the end of the ‘GPTAD’ device, it is necessary to consider the transition between both elements as a secondary load loss caused by the difference in diameter of the ‘GPTAD’ probe and the artery respectively and the subsequent variations in flow (deflecting nozzle in figure 2). These load losses can be represented as a resistance and can be calculated with the following expression:

\[ R = 8 \rho \xi \frac{Q}{\pi^2 D^4} \]  

(3)

where ‘\( \rho \)’ is the blood density, ‘\( Q \)’ is the flow which circulates in the section between the end of the ‘GPTAD’ probe and the artery, and the diameter ‘\( D \)’ is the mean value between the probe (\( D_{\text{w}} \)) and the artery (\( D_{\text{artery}} \). The load loss coefficient ‘\( \xi \)’ is a dimensionless parameter that quantifies the loss produced and depends on the geometry of the conical aperture. This value will be actually 0.4 in the best case.

The artery can be included in the model as another section of a pipe, similar to the catheter and the ‘GPTAD’ device and it must be defined by the loss of linear load ‘\( R \)’ and the inertia ‘\( I \)’ (figure 2). In addition, it is necessary to insert a parameter that represents the compressibility of the artery, in line with its Young’s modulus:

\[ K = \frac{E \cdot h}{V_0 \cdot 2r_0} \]

(4)

where ‘\( E \)’ is its Young’s modulus, ‘\( h \)’ is the thickness of the artery, ‘\( V_0 \)’ is the artery initial volume and ‘\( r_0 \)’ is the artery initial radius.

Once the elements corresponding to pump, catheter, ‘GPTAD’ device and artery have been defined by fluid mechanics, it is necessary to change from the domain of hydraulics to mechanics, to be able to evaluate the movements and efforts in clot removal, as well as to define the physical friction between the clot and the artery. This domain change is carried out by a Transformer (TF) element. To calculate the value of the coefficient defining this element, the change in the definition of the flow before and after this element is evaluated and, we can conclude that it is equal to the inverse of the effective area (it depends if the occlusion is 20%, 40% or 50% clot adjacent to the plaque).

Accurately defining the clot model in order to model it is the most complex part of the modelling. In previous models the clot has been approximated to a cylindrically-shaped element of [0.5-5] cm long, and of a mass that falls between [0.5-0.1] gram, connected to the artery wall by using an equivalent spring, with the same adhesion force every cases. The model has a spring-damper system in parallel to simulate the elastoplastic behaviour of the clot to withstand traction. This representation of the internal conditions of the clot will be maintained in all subsequent models, although they change their name. Additionally, the clot inertia also supports the bond strength to the artery. To know when the force will be reached in the equivalent spring and, therefore, when the clot movement will begin, we calculated the displacement of the spring when it is subjected to 0.01 N (and other values depending of various parameters) via a typical spring equation. Therefore, only when the spring underwent this displacement did clot movement begin. No clot movement occurred before this value was achieved.
BLOOD CLOT AND FRICTION RE-MODELING

This previous model has some difficulties that we should redefine. It focused on the way we simulated the adhesion of the clot to the wall. We think that the simulation of the complete catheter was quite realistic, so we kept it, focusing only on a new model for studying the clot and its adherence to the arterial wall.

Consideration of the adhesion force is relatively complex, and it is challenging to find the spring-damper rate that represents the junction with the wall and we only considered the in-vitro maximum adhesion force. The value of the constants in both the spring and the damper must be extremely high to simulate a firm anchor to the point of release. Simulating the moment when the clot breaks loose from the wall in the previous model method was very difficult, because the junction to the wall had to allow, clot movement, according to pressure, even if such pressure was not enough to move it in totality.

Due to these problems we look for another solution to simulate the clot and its behaviour under pressure. As we have seen before, we kept the partitions represented by inertia and joined by spring-damper system (Runion, Kunion), to simulate the elastic and plastic behaviour of the clot. However, we need to add a new inertia to solve some problems with the simulation; this inertia (Inertance1) means a punctual mass that represents the force application point.

To model the junction with the artery, the point of release and the static and dynamic friction, we decided to add to each inertia an effort source that varies depending on the moment of the simulation. We can observe the new configuration in figure 4. Each inertia will suffer a force due to suction, which should be compensated in the model with a force of friction to annul it, while the clot is in the position of static friction. Once we have calculated the flow-effort table of the system, we apply the condition that the stress on the inertia must be zero.

The effort source, as we have said before, varies. That is, when the clot begins its movement, the static friction disappears and the dynamic friction acts in the system. It is much lower that the static friction. We have calculated it by means of the Stokes law for a cylindrical solid:

$$F_{\text{dynamic friction}} = \frac{C}{8} \cdot \rho \cdot \pi \cdot D^2 \cdot V^2 \quad (5)$$

where ‘C’ is the form coefficient for a cylinder, ‘p’ is the blood’s density, ‘D’ the clot’s diameter and ‘V’ the velocity of the first partition.

The condition to determine if the clot is attached to the surface is based on the force that the spring suffers between partitions (Kunion). Hence when Kunion/Displacement(Kunion) is higher than the adherence force, the clot releases from the surface. The value of the adherence force was analyzed by C. J. Flannery (Flannery 2005) and we made use of all the necessary data from that. To calculate the adhesion strength C.J Flannery consider a not cylindrical clot that narrows down the middle due to stenosis or atherosclerotic plaque. But this geometry is only valid for the condition of static friction, because, once the clot releases, it recovers a part of its form and again becomes like a cylinder. So, we assume this model and we divided this clot into 5 parts:

![Fig. 3. Blood clot 5 parts division.](image)

We decided to make these partitions because each one has the same contact area with the artery and the same volume, in order to simplify the calculation of the adhesion force and the mass of each partition.

![Fig. 4 New developed model of the ‘GPTAD’](image)
As we have defined our clot, we then calculated the adhesion force by means of the platelet adhesion force. And we have obtained from C.J. Flannery the next equation:

\[ N^\circ \text{ platelets per area} = fp \cdot \frac{SA}{MPA} \]  \hspace{1cm} (6)

where ‘fp’ is the % of platelets in the clot, ‘SA’ is the surface in contact with the artery and ‘MPA’ the Mean Platelet Area.

Once we have the number of platelets in contact with the artery and the force/platelet, we can calculate the adhesion force of each partition and obviously of the entire clot, in this way we would have different adhesion forces depending on the form and size of the clot, and we can reference those to the artery diameter and the percentage of occlusion.

The systolic and diastolic blood pressure were introduced by using a variable pressure. If we consider that the blood pressure varies from about 120mmHg to 80mmHg (16 kPa to 11 kPa) in systolic to diastolic pressure variation in the normal cardiac cycle, we impose a rate of 1 cycle per second, we can approximate mathematically the pressure (kPa) in two parts (0.00-0.32 sec. and 0.32-1.00 sec.) by using next two polynomial expressions:

\[ P_1 = 13415 \cdot t^4 - 8508.6 \cdot t^3 + 986.58 \cdot t^2 + 177.3 \cdot t + 4.99 \cdot 1 + 11.01 \]  \hspace{1cm} (7)

\[ P_2 = -1488.6 \cdot t^6 + 6237.6 \cdot t^5 - 10700 \cdot t^4 + 9999.3 \cdot t^3 - 4719.5 \cdot t^2 + 1188.3 \cdot t - 102.8 \]  \hspace{1cm} (8)

Representing both equations we can obtain the necessary pressure values to introduce it into the 90% and 70% occlusion model, such as we can see in following figure.

![Graph showing systolic/diastolic pressure approximation](image)

**Fig. 5. Systolic / Diastolic pressure approximation**

**MODEL VALIDATION**

The object of this study involves determining the minimum pressure required for the extraction of a blood clot. To do this, by varying the values of the pressure source, the movement of the clot and the time required for its extraction are measured, thereby obtaining the optimum minimum pressure.

To carry out the model validation, the values of the parameters used in the simulation are listed in the following table.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pressure</strong></td>
<td>[0, -100] kPa</td>
</tr>
<tr>
<td><strong>Blood Density (ρ)</strong></td>
<td>1060 Kg/m3</td>
</tr>
<tr>
<td><strong>‘GP’ length (L)</strong></td>
<td>0.02 m</td>
</tr>
<tr>
<td><strong>Artery Young modulus (E)</strong></td>
<td>2.810^7 N/m</td>
</tr>
<tr>
<td><strong>Artery thickness (h)</strong></td>
<td>0.0001 m</td>
</tr>
<tr>
<td><strong>Artery length (La)</strong></td>
<td>0.003 m</td>
</tr>
<tr>
<td><strong>Artery diameter</strong></td>
<td>2.5, 3.2, 5.510^-3 m</td>
</tr>
<tr>
<td><strong>Blood Viscosity (η)</strong></td>
<td>0.0035 m</td>
</tr>
<tr>
<td><strong>Kuni</strong></td>
<td>1.91 N/m</td>
</tr>
<tr>
<td><strong>Runi</strong></td>
<td>0.035 N/s/m</td>
</tr>
<tr>
<td><strong>Inertance</strong></td>
<td>0.0000001 Kg</td>
</tr>
<tr>
<td><strong>Catheter length</strong></td>
<td>1.1 m</td>
</tr>
<tr>
<td><strong>Catheter diameter</strong></td>
<td>1.0, 2.0, 4.010^-3 m</td>
</tr>
<tr>
<td><strong>Clot length</strong></td>
<td>[1.0, 5.0]10^-3 m</td>
</tr>
<tr>
<td><strong>‘GP’ diameter</strong></td>
<td>1.0, 2.0, 4.010^-3 m</td>
</tr>
<tr>
<td><strong>fp</strong></td>
<td>0.96</td>
</tr>
<tr>
<td><strong>MPA</strong></td>
<td>5.3110^-12 m^2</td>
</tr>
<tr>
<td><strong>Fadhesion_platelet</strong></td>
<td>32.10^9 N</td>
</tr>
<tr>
<td><strong>Occlusion</strong></td>
<td>[70, 100] %</td>
</tr>
</tbody>
</table>

**Table I. Parameter values**

The parameters that define almost completely elastic-plastic behaviour of the clot and its resistance to breakage are the constants Kuni and Runi of the spring-damper systems in parallel that are among the partitions of the clot and that characterize the clot in the stretch, when it suffers the suction before it is detached from the wall.

To find the value of the Kuni parameter, Savushkin (Savushkin 2003) analyzes the stiffness of the clot and the breaking strength. We considered that the values are valid, since the parameters of the experiments described fall within our range, and therefore we can assume that Kuni = 3.41 ± 1.5 N/m. We took the minimum value 1.91 N/m to study the system in the worst possible conditions. We have also find in the referred work the rupture force necessary to break the clot, so we included it also in our simulation (Frupture = 209 ± 73 ·10^-3 N).

The Runi value Pennati (Pennati et al. 2010) was considered to be a useful parameter. This work described values for the viscosity of the blood and the clot that they use in their model; due to the viscosity of the clot, we can assume that Runi = 0.035 kg/m·s.

**RESULTS**

The aim of this simulation is to compare extraction of a clot in different arteries (5.5 mm, 3.2 mm and 2.5 mm) considering different clot lengths (1 cm up to 5.0 cm each case) and 70, 80, 90 and 100% of occlusion.

The size of the artery determines the size of the GP TAD used to remove the clot. In the study the sizes of GPTAD used was in the range 1 mm, 2mm and 4 mm of diameter for the catheter and the GP.

The existence of different clot lengths will affect the mass of the clot being removed (up to 1 gr.) and increase the time taken for clot removal as the value of the clot length increases.
It has been found, that the greater the rigidity of the clot, the shorter the extraction time. This factor is also related to the viscosity and composition of the clot that will vary in each case. Therefore, for this study we have determined the critical values possible, making the assumption that there are 96% platelets in the clot, which yields a fairly high bond strength, which gives us an idea of the maximum pressure needed.

The breaking strength of the clot, is considered, and is taken to an accuracy of an order of magnitude. With this data, it is found that in cases of $D = 5.5$ mm and $L_{clot} \geq 3$ cm and for 90 and 100% of occlusion with $D \geq 3$ mm and $L_{clot} = 5$ cm, there is a danger of rupture prior to complete clot removal, which would mean the failure of the process. Because when the clot breaks up, flow channels appear, and the blood begins to circulate through them. Consequently it is no longer possible to extract the clot. Therefore, it is taken as a benchmark, to call attention to the danger of rupture cases.

The following figures show the time it takes to move the clot 3 mm, which is the distance that is maintained during the extraction to the 'GPTAD'. In figure 6 we can observe the different times to extract a blood clot in a 100% occlusion case, with a diameter of $D = 2.5$ mm and for different lengths such as $0.01 \div 0.05$ m respectively. As we can see, the lower the length the lower is the time needed to extract the clot, because we have less adherence force, and less inertia due to the mass.

![Fig. 6. Required times (2.5mm diameter - 100% occlusion).](image)

Comparing the different occlusion percentages, in the figure the lines corresponding with the 70% occlusion are always below the ones for 100% occlusion. That is reasonable because we have more adherence area for a 100% occlusion; hence more adherence force and therefore we need more time to extract the blood clot. However we can observe that for $Lc=1cm$ both lines are very similar, that is because for such small clots the difference of adherence force due to the surface is not so significant as in the other cases.

Finally, figure 8 shows the results for different diameters (2.5, 3.2 and 5.5mm), same length ($Lc=2cm$) and for an 80% of occlusion.

![Fig. 8. Required times (80% occlusion – Lc=2 cm).](image)

We can see in previous figure that as the diameter of the artery increases, the time required to the extraction decreases. That arises because a larger diameter is associated with a larger effective area in relation to pressure. Consequently a larger force is produced by the same suction pressure. Thus, despite increasing the adherence force, the increasing of the suction force is larger, and therefore the extraction time decreases.

**CONCLUSIONS**

For this model we have focused primarily on the configuration and simulation of the clot, which was still in an early phase of study. We have studied the formation, composition and shape of different blood clots in different cases, trying to find some general parameters that could define their behaviour reliably. In particular we have studied the influence of composition, form and some parameters that directly affect the adhesive force that holds the clot against the arterial wall.

But the most significant inclusion in the model, and therefore the one that provides greater reliability to the model has been the change in the approach regarding the internal structure of the clot and its adherence to the wall. The clot is broken down into smaller partitions that are joined together by a spring-damper system, which represents the elastoplastic behaviour of the clot, and whose parameters are the stiffness and viscosity.

To develop the condition of adherence to the wall, detachment and subsequent movement of the clot with viscous friction, it was decided to approach the action-reaction, so that the clot would remain at rest by a frictional
force of equal value, but in the opposite direction, to the one associated with each partition until the condition for clot movement is fulfilled. For clot movement to occur, the frictional force changes to the corresponding value for the simulation of dynamic friction. This friction is calculated by Stokes law, and considers the motion of a solid cylinder within a viscous fluid.

With this approach, conditions and parameters, we can obtain a fairly accurate simulation of the behaviour of a blood clot attached to an artery wall. This is evidenced by various simulations and we can demonstrate how each partition is involved in the model.

The influence of the composition, stiffness, viscosity and geometry of the clot on the clot extraction time can be seen. The effect of increasing the extraction time factors such as the percentage of platelets, the viscosity, the length and decreasing of the rigidity of the clot can be modelled. The results obtained from modelling are consistent with those observed in in-vitro measurements.

So we considered that the model is valid and can be used for different cases now and in the future, as more experimental data become available.

It is therefore concluded that the model fits in reality and can be extrapolated to different models and cases, due to the flexibility and approach of the model under various conditions. This model can be used more generally as opposed to the previous models we developed which could only be used in particular conditions.

Our results enabled us to draw the following conclusions:

- In general, the greater the contact area between the clot and the artery, the greater the force of adherence and therefore higher extraction time. That occurs when we increase the length of the clot or the percentage of occlusion (also the percentage of platelets), but not when we increase the diameter of the artery, because we increase the effective pressure area more than the adherence force, so we increase the force, and we reduce the time needed for clot extraction for the same suction pressure.

- Systolic blood pressure helps the extraction, although not as much as other factors.

- Care should be taken in the removal of long clots as these are in danger of reaching the rupture force and of breaking up, so they cannot be extracted.

- The stiffness and viscosity of the clot influence the process, because they determine uniquely the elastoplastic characteristic, and so to what extent the clot will deform before release and how it will recover once detached from the wall. So the higher the stiffness, the lower the time of extraction. The main objective of the development of this simulation model was to obtain the range of pressure required to perform the clot extraction and to check that this pressure together with the time required to complete the operation are reasonable and within acceptable clinical boundaries for eventual potential use of the GPTAD on a patient. We also confirm that the results are similar to experimentally obtained data in vitro. These studies allow optimization of the device to assist in possible future use of this device in patients with thrombosis.

**FUTURE WORKS**

The presented mathematical modelling is now being used to simulate particular cases in the human body and in the future will include a more precise rupture force, and will be compared to other thrombectomy catheters in order to determine the best device to use in each patient under given conditions.

**REFERENCES**


BIographies

GREGORIO ROMERO received his Mechanical Engineering degree from the UNED (Spain) in 2000. He got his PhD Degree from the Technical University of Madrid in Spain in 2005 working on simulation and virtual reality, optimizing equations systems. He started as Assistant Professor at the Technical University of Madrid in Spain (UPM) in 2001 and became Associated Professor in 2008. He is developing his research in the field of simulation and virtual reality including simulation techniques based on bond graph methodology and virtual reality techniques to simulation in real time. He has published more than 50 technical papers and has been actively involved in over 30 research and development projects.

ISABEL MORENO is studying Mechanical Engineering in Technical University of Madrid (Spain) and she will obtain the degree in 2011. She is working in simulation models by using the Bond Graph technique and she has developed the modified model presented in this paper. Lastly she has been involved in one educational project and pilot activities promoted by the European Commission and other Spanish institutions.

M. LUISA MARTÍNEZ received her Mechanical Engineering and Doctoral degrees from the Technical University of Madrid (Spain) in 1990. She got her PhD Degree in 1997 working on variational geometry. In 1990 she started to work as Associate Professor at the Technical University of Madrid in Spain (UPM). Her thesis was focused on variational geometry. She usually works in the field of computer graphics, virtual reality and CAD. During this time she has been involved in different educational projects and pilot activities promoted by the European Commission and other Spanish institutions. She has published over 55 technical papers and has been actively involved in over 40 research and development projects and different educational projects.

GILIANN PEARCE is a Senior lecturer in Clinical Physiology and Anatomy at the University of Wolverhampton, UK and is an Honorary research fellow at the RJAH Hospital Oswestry, UK. She is the co-inventor of the ‘GP’ Mechanical Thrombectomy Device. She graduated with degrees in Physics; Biomedical sciences; a Ph.D in Astrophysics and a B.M. B.Ch in Medicine. She practised as a medical doctor in the NHS before taking up research in the field of medicine. She has several publications in Astrophysics, medicine and has patents granted and published for medical devices.

NEIL D. PERKINSON was born in New Zealand and has lived in the UK since the 1970’s. He has an established career in the field of electronics before becoming an ordained priest in the Church of England. He has been a priest in the Church of England for several years in parishes in the North and South of England. He is currently Vicar of All Saints, Darlaston, West Midlands, UK. He is the co-inventor of the ‘GP’ Mechanical Thrombectomy Device.
SIMULATION APPROACH FOR SYSTEMS RADIATION BIOLOGY

David Sarramia, Lydia Maigne, Vincent Breton, Clermont Université, Université Blaise Pascal, CNRS/IN2P3, UMR 6533 Laboratoire de Physique Corpusculaire, BP 80026 F-63171 CLERMONT-FERRAND, France {sarramia, maigne, breton}@clermont.in2p3.fr

Laurent Deroussi, Benjamin Louvet Clermont Université, Université Blaise Pascal, CNRS, UMR 6158 Laboratoire d’Informatique Modélisation et Optimisation des Systèmes, BP 10448, F- 63173 CLERMONT-FERRAND, France {louvet, deroussi}@moniut.univ-bpclermont.fr

KEYWORDS
Individual based model, Cellular automata, Bacteria growth.

ABSTRACT
Systems radiation biology is a new field that addresses classics problems in radiation biology by integrating the impact of DNA damages into multi-scale models of biological systems to provide a better understanding of health risks such as cancer. We propose a global framework combining *in-vivo* and *in-silico* experiments. Several main steps are needed: designing a platform, building a multi-scale cell model, coupling the platform and Geant4 DNA software, and validate the *in-silico* and *in-vivo* processes. In this paper, after having presented the global framework, we present the platform design and its validation.

INTRODUCTION
Understanding the impact of radiation on living organisms is of crucial importance to both biology and healthcare. Living organisms are constantly exposed to ionizing particles through sunlight and radioactive materials in environment. This exposure can damage cellular genome and can kill the cell or can result in mutations which lead in initiator for evolution or in causes of diseases as cancers. Radiation is also used in radiotherapy. Radiobiology aims to deeper understand the interaction of radiation with living organisms by integrating the biology of DNA damage with extra-cellular signaling pathways and transcriptional regulatory networks. These models require simulating the living organism, the interaction of ionizing particles inside it, and the damage and repair of DNA. Having a complete model opens interesting avenues for healthcare as it would allow personalized treatment taking into account the patient’s genomic data.

The framework we propose (see Figure 1) is a sequential *in-vivo* and *in-silico* process having three steps:
- First step is to produce a population of biological cells by *in-vivo* experiments and *in-silico* modeling.
- Second step is to irradiate cell population by *in-vivo* experiments (using radiation therapy beams) and by *in-silico* modeling (using the GEANT4 DNA software (Agostinelli et al. 2003)).
- Third step is to examine the growth of the irradiated cell population by *in-vivo* experiments and to model it with *in-silico* computing.

The cross comparison between *in-vivo* and *in-silico* results will provide a through validation of the framework. It is innovative because it couples a multi-scale biological model to a full description of the interaction of the ionizing particles with the irradiated cell using Geant4-DNA. The models used to describe the phenomena at the different relevant scales are constrained by experimental data.

![Figure 1: General framework with *in-vivo* and *in-silico* processes](image)

Radiobiological cell survival model will be managed using validation and parameter evaluation. From the simulation part, the organism model will be built from full organism to the DNA structure. Next we simulate ionizing radiation on model organism using the Geant4-DNA toolkit to produce direct and indirect (by interacting with chemical species produced by the radiolysis of water) interaction probabilities causing strand breaks. A lot of work has to be done to propose an aggregated radiobiological cell survival model at cell level wich could take into account the different possible outcomes after radiation and the three categories of radiation damages (lethal, sublethal and potentially lethal), whereas metabolomic level is left out for now.

For these purposes, a comprehensive set of growth curves for each managed scale is needed as the impact on cell cycle and cell death must be evaluated. As shown before, the study of the multi-scale effect of radiations on organisms needs a lot of input data, parameter identification methods and biological phenomenon understanding in order to avoid bias in results analysis. That's why, in order to validate our approach, we will start the validation of our framework by studying bacterial population of *Escherichia coli* bacteria. For this, we need a software platform that can model a population of bacteria individually. The next section proposes a review of platforms and a classification is proposed in order to distinguish platforms according to biological features and the scales.
BACTERIA GROWTH PLATFORMS

In the literature, many platforms use individual based model (IBM). (Hellwegher and Bucci 2009) have reviewed several IBM frameworks for bacteria growth modeling. But they haven't provided a true classification according to the biological features. Among platforms, some are relevant one to our objectives.

We propose to distinguish platforms according to the way the model is managed and how cell population evolves. The first criterion distinguishes dedicated and generic platforms. A dedicated platform can be resumed as a parameterized model whereas a generic platform can read models written in a standard generic language such as SBML (System Biology Markup Language; Hucka et al. 2010). The second criterion separates the evolutionary from the adaptive platforms. Cells can react to quick environment changes by adapting themselves to the new conditions. However, generation after generation, cells are evolving by an undergoing modification process. When it deals with evolution, a platform will be called evolutionary platform whereas one dealing with adaptation will be called adaptive. A quick review is shown in Table 1. Another set of platforms only dedicated to sub cellular processes (SimCell (Wishart et al. 2005), Voell (Schaff et al. 1997) has been identified. As this level is not currently addressed, we won't detail further such platforms.

Two generic platforms (CompuCell3D, CellDesigner), one evolutionary platform (Avida) and an adaptive one (Bacsim) are briefly detailed in this review.

CompuCell3D (since 2004; Izaguirre et al. 2004) is based on the Glazier-Graner-Hogeweg (GGH) approach. Multi-scale simulations are easy to make as spatial set of cells can represent from a single cell to a cluster of cells and even non-cellular material. It uses CC3DML (CompuCell3D Markup Language) to describe models. Dynamics are modeled as a combination of several equations and a cellular Potts model.

CellDesigner (since 2008; Funahashi et al. 2008) is a editor for gene-regulatory and biochemical networks. It is compliant with SBML and can simulate models using SBML tools and SBW (System Biology Workbench; Hucka, 2002).

It can also connect to species databases.

Avida (since 2003; Pennock. 2007) is simulating a digital organism population, by modeling only a genotype and the corresponding phenotype. It aims at studying Darwinian evolution principles. A circular chromosome is constituted by elementary elements. The reading of the chromosome leads to replication. So it does not represent a reality but enables the understanding of some processes. Death, growth and division are taken into account.

Bacsim (since 1998; Kreft et al. 1998) simulates bacteria population. It takes into account between other things diffusion process, death, and growth. The main objective is the emergence of a biofilm and the interindividual relations.

The different platforms deal with one scale at the time and the hypothesis of evolution or adaptation has been done at their design. This is not relevant with our objectives that are to be able to study the adaptation of biological cells to their environmental conditions but also to study the evolution of a cell population under a radiative environment. As the second point contains multi-scale processes that cannot be tackled in their whole directly, an aggregation model should be designed as a “black box” for the metabolomic functioning modified by the radiation impact. But, the above state of the art implies that a combination of various platforms will be needed in order to achieve our objectives. The next section details our platform.

<table>
<thead>
<tr>
<th></th>
<th>Avida</th>
<th>BacSim</th>
<th>CellDesigner</th>
<th>CompuCell3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edition</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Simulation</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>SBML</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Database</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Intra-cellular models</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Inter-cellular models</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Diffusion</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Quick review of platforms features

PROPOSED PLATFORM

This platform has been designed to manage any kind of models. In its current version, the platform can run models based on a discrete approach such as biological cellular automata (CA) and their extensions. The covered scales are the cell, the cell population and the environment. A biological cell is modeled by a CA cell. In the biological context, the data used are supposed to be available or measurable. This will lead to write simple models with few parameters as BacSim authors are advocating for (Kreft et al. 1998). We have planned three steps to achieve this goal: validate/verify the platform functioning, get good growing curves, and validate a full cell model. Here we present the validation part.

The platform working has been validated on common CA ranging from classical CA (game of life; (a; Gardner. 1970), wireworld (b; Dewdney. 1990)) to more biological CA (daisyworld (c; Watson and Lovelock. 1983)). We have verified that for those examples, we got the awaited behavior. These examples were taken as they enable us to explore some CA characteristics, all having a regular lattice:

- Space dimension: 1D (b) and 2D (a,b,c)
- State set : 2 (a), 3 (c) and 4(b)
- Transition function: deterministic (a,b), probabilistic (c)

Game of Life is a one of the most famous 2D cellular automaton, with 2 possible states (active or not), and a deterministic transition function defined by two rules. This model has been successfully tested both with random and given known initial conditions such as “the clown”.

Wireworld is a 2D cellular automaton, ruled by a deterministic transition function and with 4 possible states (empty cell, copper wire, electron head or electron tail). Four rules are implied in the transition function. As an electron would run along a simple copper wire, this transition function could be applied on 1D cellular automaton. The platform displayed a good behavior of the CA in several tested configurations from simple wire to more complex circuits with any types of logical gates.

Daisyworld; we consider a simplified version of the Lovelock’s model (this is a homeostasis model). We use a 2D CA with 3 states (grass, black or white flower), and a global temperature, evaluated by flowers ratio, bringing out a balance between the two kinds of flowers. Expected results have been obtained.

The proposed individual-based model for bacteria

The model we need to develop should be “able to predict”
the effect of the DNA structural or chemical modification by cell irradiation on the population growth.

The proposed CA is a 2D/2 states deterministic one, and its transition function models that an empty cell remains empty unless one neighborhood cell is ready to divide. Each living cell will run the algorithm ruling the cell’s cycle independently. This cell cycle depends on the bacterium’s parameters, on environmental (other cells, temperature, pH) and nutrient conditions (sugar, growth factor, ions…). The pseudo-algorithm of the cell cycle is shown in Figure 2. The symbol “ϕ” indicates that some biological functions are embedded in the corresponding boxes (initialization, extraction and maintenance).

Figure 2: Proposition of a bacterial cell cycle model.

The cell cycle begins when a cell is generated. It ends either when the cell dies or when it divides. The metabolism of the cell is modeled thanks to the concepts of internal state. If the internal state of a cell is too damaged, the cell dies. On the contrary, it draws from the environment the nutrients it needs and progresses in its cell cycle. When this one is completed, the cell is ready to divide.

The “initialization” box allows to define the own parameters of each bacterium. It is a way to introduce intra-population variability, and mutations induced by DNA damage. The “extraction” box models the interactions with the extra-cellular environment. The extracted compounds amount is evaluated according to the considered growth factors. The Fick’s laws of diffusion are used to model the movement of chemical species in the environment.

The biological functions of the cell are updated thanks to the “maintenance” box. If the environment provides a sufficient amount of energy, the bacterium progresses in its cell cycle. Else, the internal state of the cell deteriorates.

The first step of our work consists in validating our framework. We develop a first version of the model in which we assume that bacteria have no nutrient and space restriction for growing. The obtained growth curves show the expected exponential phase. In future work we will experimentally study the growth of the bacterium E.coli in a batch culture to adjust the parameters of our model. Two reasons motivate the choice of E.coli for these experiments, its fast growth rate, and the good knowledge of its genome.

CONCLUSION

In this paper, we propose to address the area of systems radiation biology. To achieve the main problem of measuring radiation deposits and its impact on life, a global framework has been defined. This framework coupled an in-silico process and an in-vivo process that are able to exchange data at any time. One main point is to be able to simulate the growth of biological cells in a given environment before and after radiation. This implies multi-scale capabilities for the simulation tools and the possibility to run easily new models.

A review of existing simulation platform has shown that none of them fulfill all the requirements. This is the reason of the design of our platform. In this paper, we mainly address the validation of the designed platform by implementing and running models from the literature. The platform has displayed in all the case the good behavior. Finally, as the biology is our main aim, we have presented a simplified model for E.coli. With no diffusion process, the model simulates correctly the exponential phase of growth. We are now implementing a diffusion process following Fick’s law in order to obtain the full growth curve, and in particular the stationary phase. Experimental studies in LPC laboratory are also currently managed to obtain growth data in various environment for the temperature and the nutrients.

ACKNOWLEDGEMENTS

We acknowledge fruitful discussions with the Geant4-DNA collaboration, especially Sébastien Incerti and Hervé Seznec. We also thank Sylvie Norre and Michel Gourgand from the LIMOS laboratory for their helpful collaboration.

REFERENCES


MEDICAL
DECISION
SUPPORT
SYSTEMS
A Moral Decision Support System in Medicine

Gabriel Pontes, Ana Duarte, David Cuevas, Maria Salazar, Miguel Miranda, António Abelha and José Machado
Universidade do Minho, CTC, Departamento de Informática, Braga, Portugal
gabrielpontes@di.uminho.pt {pg17654,pg18566}@alunos.uminho.pt mcsalazar@chporto.min-saude.pt
{miranda,abelha,jmac}@di.uminho.pt

KEYWORDS
Moral Agents, Decision Support Systems, Reasoning

ABSTRACT

Intensive Care Units are, in hospitals, special units where the use of ethics is common. Usually, there are few available beds and financial costs are huge. In this paper, it is presented a model for the simulation of the allocation of resources, in an Intensive Care Unit. Since it is a problem that deals with human life, decisions must be supported by a sound reasoning process in order to cause the minimum damage. Therefore, it is important to introduce the concept of ethics and moral reasoning, in particular taking the maximum advantage of moral agents, which are entities capable of making intelligent decisions based on moral guidelines. These entities have also an intelligent behaviour, simulating a physician conduct when there is an overcrowding of patients in the Unit. The decision process is carried out based on the computation of some critical factors, including a death rate, the survival quality and financial costs. The death rate is achieved using SAPS3 algorithm.

INTRODUCTION

An Intensive Care Unit (ICU) is a specialized section of a hospital which provides intensive medicine and treatments to people in critical health conditions. If a patient, in critical conditions, needs to benefit from the ICU services, which patient should be transferred in order to turn a bed available? Since it is a problem that deals with human life, decisions must cause the minimum damage. On the other hand, usually decision support systems are based on rational or emotional intelligence (Rady and Johnson 2004). In ICUs, moral and legal judgements must also be taken under consideration to take decisions on the fly (Andrade 2008). Most of the patients in the ICU are unable to help in making a decision despite of the information that might be presented by the physician, being either incompetent due to the nature of the critical illness or because of the sedative drugs widely used in ICU (Danbury and Waldman 2006). Furthermore, the medical team needs to consider other constraints such as the rights of relatives in the decision process, futility of treatment and resources management in very dynamic situations where time is a major constraint. The notion of virtual entity is here used to differentiate entities with higher levels of autonomy, learning, prediction and decision from a mainly reactive and controlled machine. Moreover, considering developments in the area of informatics and Artificial Intelligence (AI) in particular, it must be considered that many of these entities can exist within a single physical machine or even that a single one can be distributed within limitless machines. Therefore the notion of a virtual entity in this case is similar to the concept of an agent in the area of Multi-Agent Systems (Wooldridge and Jennings 1995, Rao and Georgeff 1995).

As virtual entities become more complex and hold critical functions, a justified doubt and concern regarding the impact of actions performed by these entities arises. From the numerous scenarios where they can interact with their surrounding environment, some carry moral consequences and describe ethically intricate actions from a human point of view. From the need to prevent immoral decisions and ensure confidence regarding these virtual entities, further understanding of the capacity of moral agency, moral modelling and the complexity of human moral ethics (Torrance 2007, Machado et al. 2008). Also, it will make sure that the affective component that typically characterizes human beings will be put aside, thus preventing decisions less exempt. The notion of teams is not limited to the daily cooperation in each service of an healthcare unit. These teams include elements from different scientific domains and backgrounds. Moreover, considering the moral complications and the complexity of medical practice, argumentation through the use of auxiliary indicators or a multidisciplinary team to support the decision making process, presents itself as a mean towards the validation and perhaps improvement of such processes.

Ethics pursues in supporting the proper way of living by human thought, and it can be defined as the science of human conduct. And it is this kind of reasoning that is required by the system as in mentioned situations it is intended an ethical behaviour, namely, that behaviour that is considered good (Anderson et al. 2006, Machado et al. 2010, Pereira and Saptawijaya 2007). There exists no definite solution for modeling ethical virtual entities,
and presently several approaches are being presented and some compared against one another. Looking at the present and the ongoing research in this area, different methodologies for modeling moral capabilities using AI techniques can be segmented according to their main characteristics (Tonkens 2009, Himma 2008). Thus, the aim of this paper is focused on developing a Decision Support System, which allows an assistance to doctors in decision making, in view of the impossibility of directly allocate beds to patients in ICU. That is, it is intended to carry out a simulation of doctor conduct as a decision agent.

Finally, another aim is also to take advantage of some learning techniques used in AI, such as Rules-based Reasoning and Case-based Reasoning, thereby building up over time, a large and complete database of cases and rules.

**QUALITY OF INFORMATION - A LOGIC PROGRAMMING APPROACH**

It is more and more important to have a decision making process with quality (Machado et al. 2008). For reaching this goal, ones have to evaluate the completeness and the soundness of knowledge that can be extracted from medical databases. Intelligence is frequently rational but emotions and morality have also to be attended. In medicine, moral values are sometimes more important than economical ones and some attributes must be evaluated besides their subjective levels.

A program in Extended Logic Programming (ELP) is a finite collection of rules and invariants, defined by its extensions, in the form:

\[ p \leftarrow p_1 \land \ldots \land p_n \land \neg q_1 \land \ldots \land \neg q_m; \text{ and} \]

\[ \neg p_1 \land \ldots \land \neg p_n \land \neg q_1 \land \ldots \land \neg q_m (n, m \geq 0) \]

where \( \neg \) is an atom denoting falsity, \( p_i \), \( q_j \), and \( p \) are literals, i.e. positive atoms or atoms preceded by the strong negation symbol \( \neg \).

Let \( \Gamma \) a program in ELP and \( g(X) \) a query where \( X \) contains variables \( X_1 \land \ldots \land X_n (n \geq 0) \):

- the answer of \( \Gamma \) to \( g(X) \) is **true** iff \( g(X) \rightarrow \text{demo}(\Gamma, g(X), \text{true}) \)

- the answer of \( \Gamma \) to \( g(X) \) is **false** iff \( \neg g(X) \rightarrow \text{demo}(\Gamma, g(X), \text{false}) \)

- the answer of \( \Gamma \) to \( g(X) \) is **unknown** iff \( \neg \neg g(X) \land \neg g(X) \rightarrow \text{demo}(\Gamma, g(X), \text{unknown}) \)

\( \neg \text{survival}_{rate}(X, Y) \rightarrow \neg \text{survival}_{rate}(X, Y) \land \neg \text{exception}(\text{survival}_{rate}(X, Y)) \land \neg \text{survival}_{rate}(X, \text{unknown}) \land \neg \text{survival}_{rate}(X, Y) \land \neg \text{survival}_{rate}(X, Z) \land \neg \text{exception}(\text{survival}_{rate}(X, Y)) \land \neg \text{exception}(\text{survival}_{rate}(X, Z)) \land

\text{ag}_{\text{survival}_{rate}}(gb, R, V)

\text{ag}_{\text{survival}_{rate}}(pd, R, V)

With this methodology it is possible to define scenarios for each patient, to generate hypothesis for each patient and to measure the quality of information for each hypothesis.
METHODOLOGY

First, it was important getting familiarity with some concepts like the operating of ICU and its management tools. It was necessary to get into the subject related to the evaluation of illness severity and study some scoring systems. After the whole study and investigation, it was necessary to define a methodology for the development of the application. For the increase of flexibility and to allow a reduction of system maintenance costs, it was used a Three Layer Model for the system development (Presentation Layer, Business Layer and Data Layer). After analyzing the concerned problem, and adapting the system to the architecture model mentioned above, it is concluded the importance of a module capable of storing all the information related to the cases and rules; a module capable of conciliating the data received with the information stored in the database; and a module capable of displaying/requesting what it is required in the aims. So, the domain of the application boils down to the following scheme:

There is a moral agent with the ability to make decisions based on ethics, and that uses for that purpose, any information it needs from the database. Apart from making use of the database, it also helps it becoming more complete, allowing it to add all the fresh information derived from agent decisions. At the time of making decisions, it will be produced several hypotheses containing potential solutions. These components will be discussed more further in the following chapters.

PATIENT RECORDS

First of all, we need to record the information of the patients. This task is performed using AIDA-EHR, an Electronic Health Record Application used in Centro Hospitalar do Porto (CHP), one of the Portuguese major hospital and which is a product of a research partnership between the University of Minho and CHP. Whenever a new patient admittance, the database stores historical and actual information using interoperability services available in AIDA-EHR. A patient also has associated the information of his presence (or not) in the ICU. Besides, it is included in his patient form, a range of measurements that have been made or monitored since his entry in the unit. These measurements consist in a set of parameters used for the calculation of the rates required to obtain the final decision. Either the probability of death, the rate of life quality, or the rate of the treatment costs, are dependent on a particular number of factors. This will be explored in the moral agent section.

RULE BASE

In order to complement these applications, some models have been developed to evaluate the patients with respect to their survival/death rates, such as, for instance, SAPS3 (Simplified Acute Physiology Score 3) (Metnitz et al. 2008). Other scoring models where also taken into account, such as APACHE II or SAPS2. We chose SAPS3 since it is a more recent and more complete model than SAPS2, and is intrinsically related to the evaluation of the patients received in Intensive Care Units. To calculate the probability of death is then used the SAPS3 algorithm. After a set of entries of values in a form, it is generated a final score. This score is then converted into a rate by the moral agent. Thus, the attributes of SAPS3 match those that take part of the measurements of each patient, already mentioned. With the support of the documentation provided by SAPS3, we could obtain the scoring values for each attribute value (Metnitz et al. 2008). Therefore, and knowing that the final score arises from the different scorings found, it was necessary to create rules that associate an attribute value to a score. Thus, not only the a measurement is associated with its values, but also with the scores that arise from them, facilitating the final calculation. It was also necessary to establish rules that would support the calculation of life quality and the total treatment costs. Unlike what was done for the death rate, in these two cases, we do not apply known algorithms, being then necessary an imaginative creation of those rules.

CASE BASE

In order to increase the potential of the Case-based Reasoning (CBR) developed for this application, it is important to record all the cases that doctors will be facing. Each case consists in two measurements: one that belongs to a patient already occupying an ICU bed; other belonging to the patient that just arrived and waits for a bed. Upon insertion of the recent case, it is also expected further information, including the solution adopted, and the respective justification. Each time the CBR module is executed, it can easily access the information related to the cases, being then able to analyze and compare all the characteristics that compose a measurement (from each patient).

MORAL AGENT

The moral agent is the one who takes the responsibility for the decisions displayed in a suggestive way to the user. It uses ethical reasoning in order to attempt to address its behaviour to what is considered to be a behaviour, and thus meeting the best way of living and coexisting in the society. To be able to solve any ethical dilemma, it is necessary the fulfillment of a variety of tasks. So, the methodology followed by this entity shall be the one described below. In particular, it may attend to the particularism and the classification and reclassification of moral cases (Guarini 2006).
THE PROBLEM

This is the first task present in the decision process. This phase illustrates the gathering of all possible and relevant information, that will be used in a subsequent phase. In this particular case, this information corresponds to the measurement data of the patients. Namely, identifying the problem is to make a detailed description of the case in hands.

RESPONSE ACTIONS

After the problem identification, it is necessary a comprehensive analysis of all possible solutions. That is, it takes into account the possibility of withdrawing any of the patients of the ICU, or even none. After analyzing all the potential solutions, it is important to consider all of their consequences.

It is analyzed the consequences through the measurements results. These are reached by calculating the three weighting factors, that were mentioned along the document. The death rate is considered the most severe consequence. Before removing a patient from the ICU, it is necessary to take into consideration this result, and attempt to understand if he is going to die or not. This calculation is performed on the basis of the rules base, which in turn is supported by SAPS3. So, after summed up all the points from the measurements, the death rate can be calculated by the moral agent, based on the following receipts:

\[ \text{Logit} = -32.6659 + \ln(SAPS3\text{score} + 20.5958) \times 7.3068 \]  
(1)

\[ \text{Probability of death} = \frac{\text{e}^{\text{logit}}}{1 + \text{e}^{\text{logit}}} \]  
(2)

The first equation relates the SAPS3 score with the vital status at hospital discharge. The second illustrates the probability of mortality. With regard to the life quality rate, this calculation is performed in the basis of the rules stored in the database. Although not as crucial as the death rate, it is also a factor that shall be taken into account.

For example, knowing that a patient has a high probability of dying, but otherwise could stay into a vegetative state, would it compensate the loss of other patient at the price of this decision, that would possibly have a better quality of life than the previous one? This is an example of an ethical dilemma, very hard to handle. But it will be solved by the moral agent, based on the consideration of several factors. We assume a major importance to the factor age, namely, the older, lower the quality of life. Since in the death rate, the age participates in the total scoring, and since the older patients stay in advantage with regard to getting a bed because of the lower value of the death rate when compared to younger ones, the rule implemented for life quality is an attempt to overcome this fact. Also, it is important to take into account the possibility of getting critical diseases, or even to stay in vegetative state.

Treatment costs are the least important factor with respect to human life. However they do exist, and sometimes cannot be supported: either due to monetary issues, whether to temporal affairs. These two characteristics are present in the rules of costs, in the database. After calculating these three rates, we are prepared to the final rate, which is a consequence of the balance of all the factors at play. The death rate is worth 70% of the total rate; the life quality rate is worth 20%; and the costs rate takes the other 10% of the total percentage. In the final rate, the higher the score is, the most conditions the patient meets to abandon the Intensive Care Unit.

THE lesser EVIL

After analysis of the previous points, the agent is prepared to make a decision. The solutions chosen are those which lead to a lesser evil, having less bad consequences. Although if it is found the solution most advisable, others are also displayed to the user, giving the doctor, choice assumption. After the selection of an item, the moral agent implements the solution chosen, updating the whole system.

However, when many cases of this sort begin to be solved, it also starts creating the database of cases, that will assist the Case-based Reasoning module. Thus, even before performing the actions that will lead to the allocation of the patient, the agent will confirm if perhaps there will be a similar case in the current case base. If so, it is applied the same solution as the most similar case. If not, the moral agent will proceed with the normal execution of the allocation, explained in the previous scheme.

CASE BASE REASONING

One of the most natural ways of reasoning is know nowadays as the Case-based Reasoning (CBR) (Nugent and Cunningham 2005). CBR allows reasoning by remembering past events. This is a problem-solving methodology that consists in identifying and justifying the solution to a given problem, by the rebuilding of similar situations, reusing and adapting knowledge of such situations.

CONCLUSIONS

From a large dataset and from the estimates of weights of all alternatives to solve a problem, it is possible the development of a generic model of decision making capable of quickly assist the user. The resources used during the project were the Java language and Oracle database, which revealed to be very useful for the creation of three-layer applications. It was developed and validated a database capable of storing
all the information needed and supporting the required queries. It was also developed a moral agent that is able to analyze and consider the several hypothesis, picking the most advisable ones. A major difficulty that was found was related to the measurement and rules information, and consequently the calculation of the rates, due to the limited knowledge in medicine area. A disadvantage of this fact was the use of rules that were not real, but even so revealed to be useful in the context of the simulation. It was found that the application is able to do what has been objectified. Nevertheless, it is still necessary some further features, in order to make the application more robust, strong and complete.

There were also certain implementations based on different approaches, but they were abandoned.

**BIOGRAPHIES**

Gabriel Pontes, Maria Salazar and Miguel Miranda are PhD students in Biomedical Engineering (Medical Informatics), in University of Minho, Portugal.

Ana Duarte and David Cuevas are MsC students of the departamento de Informática of Universidade do Minho, in Braga, Portugal.

António Abella and José Machado are professors at the Department of Informatics, Universidade do Minho, Braga, and researchers of CCTC (Scientific Center for Computation Technologies).

**REFERENCES**


Himna K.E., 2008. *Artificial agency, consciousness, and the criteria for moral agency: what properties must an artificial agent have to be a moral agent?* In *Ethics and Information Technology*. Springer.

DIGITAL CLINICAL GUIDELINES MODELLING

Tiago Oliveira¹, Ângelo Costa², João Neves³, Paulo Novais⁴

¹,²,⁴ Department of Informatics
University of Minho
Braga, Portugal
¹tiago.jose.martins.oliveira@gmail.com
²acosta@di.uminho.pt
³ Centro Hospitalar de Vila Nova de Gaia/Espinho
EPE
Portugal
joaocpneves@gmail.com

KEYWORDS
Digital, Clinical, Guidelines, Modeling, CIG, CG, Medical Aid, Knowledge Representation

ABSTRACT
Healthcare environments are very demanding, because practitioners are required to consult many patients in a short period of time, increasing the levels of stress which usually harms the outcome of healthcare processes. The short time practitioners have with their patients does not facilitate informed decision making and checking all possibilities. A possible solution is the use of guideline-based applications, because they have the potential of being an effective means of both changing the process of healthcare and improving its outcomes. However, current Clinical Guidelines are available in text format as long documents, which render them difficult to consult and to integrate in clinical Decision Support Systems. With this paper we present a new model for guideline interpretation, in order to facilitate the development of guideline-based Decision Support Systems and to increase the availability of Clinical Guidelines at the moment of the clinical process. This model will also provide mechanisms to comply with cases where incomplete and uncertain information is present. The development and implementation of this model will be presented in the following pages.

INTRODUCTION
The health care sector is very demanding on health care professionals. They work several hours in stressful environments and make important decisions every second. It is only normal that sometimes psychological and physical fatigue may lead to a decline in healthcare quality. This leads to an increase of stress levels of the professionals and the management office, which is reflected in the quality of medical treatment, forming a vicious cycle. Although healthcare professionals are human beings, errors in their area of expertise are not well tolerated, since their errors can cost a human life. A study conducted in the Mayo Clinic (United States) (West et al. 2009) provides a useful insight of the levels of stress the physicians are submitted to, from the real registered errors and from the physician perspective. The number of hours and patient per hour that the physician attends and the levels of stress are strongly correlated. The typical spawn of such stress leads to factors as fatigue, burnout and depression.

Hence it is necessary to lessen their burden by providing them the means to make better informed decisions. Clinical Guidelines (CGs) and their computer implementation may hold the key to address this issue.

CGs are evidence-based statements that provide recommendations to patients and healthcare professionals about a medical procedure. These guidelines are good practice manuals that focus in the effectiveness, efficiency and transparency of medical practice (Woof 2000).

Computer-based guideline systems may also give a solution to other current problems in hospitals and healthcare facilities. For instance the use of complementary means of diagnostic in situations where they are not really required is a great problem for hospital management, because it greatly increases the expenses. The recommendations of CGs assure that these means are only used when necessary and can also reduce the variability in medical practice, as well as the occurrence of medical errors. As a result the gap between ideal medical practice and actual medical practice, shown in recent research, will be reduced and healthcare quality will improve (Kawamoto et al. 2005)(Langley et al. 2009)(Sackett 1997).

CLINICAL GUIDELINES

During the last two decades, many countries started paying attention to the development of Clinical Guidelines (CG), recognizing their importance in the improvement of healthcare quality. CGs are documents based on scientific evidence and expert opinion, developed to assist practitioner and patient decisions about appropriate healthcare (Rosenfeld and Shiffman 2006).

Computer Science progressed greatly in the last years, expanding its reach to multiple areas, including medical science. The adoption of evidence-based procedures helped CGs to become a standard among medical personnel. Despite the fact that doctors have the specificities about the most common diseases memorized, CGs can still provide their contribution in difficult cases, where they can compensate for any knowledge gaps.

Although CGs are important sources of information, since they keep updated data about several symptoms, diseases and medical procedures, they are quite long documents with a high degree of complexity, which makes them inadequate for real life situations. This inadequacy is also caused by the speed at which information and guidance are required in a real life clinical situation. It is possible to address these issues through Computer Science, with computer representations of CGs, called Computer-Interpretable Guidelines (CIGs), the development of which started in the middle of the 1980’s (Hommerson et al. 2004)(Hommerson et al. 2008).

Studies conducted (Sánchez et al. 2009) (Cannon et al., 2000) demonstrated that with the adoption of current
applications of CIG’s have helped to minimize the clinical errors and provide a better overall efficiency of patients attendances, avoiding also hospital spending in unnecessary complementary exams.
This also affects positively the service provided to the user, by minimizing the time spent in the hospital and having a more precise diagnostic, scientifically justified. However, nowadays hospitals are reluctant to use CIG based applications, because the available solutions show some functional issues, such as time of execution and the learning curve. These facts disuade the use of the computerized solutions, where the medical environment of operation has time as a key factor.

RELATED WORKS

To implement guidelines within a computer-based clinical decision support system, guideline modeling is a critical issue. A good depiction model for guideline modeling will provide the user a good understanding of the clinical process, thus making it more transparent and consistent and less ambiguous and redundant. There are a number of existing formalisms for expressing CIGs, for instance Arden Syntax, Guideline Interchange Format (GLIF), Asbru and SAGE, among others.

The oldest depiction model is Arden Syntax (Kim et al. 2008). It was created in 1989 and its approach sees guidelines as independent modules, little pieces of knowledge for very specific situations that are easy to share and integrate in different Decision Support Systems (DSSs). Each independent module is called Medical Logic Module (MLM) and contains knowledge for a specific decision in his Knowledge Compartment. A MLM has two more compartments called Maintenance and Library. The current version of Arden Syntax is Arden Syntax 2.0 and it is now a standard of Health Level 7 (HL7).

In 1998, the group Intermed Collaboratory presented the GLIF (Peleg et al. 2000) depiction model, which focuses in modeling workflow, relying for this purpose on the Task Network Model (TNM). This depiction model consists of a set of steps that represent different moments in the clinical process. These steps are Decision steps, Patient State steps, Branch steps, Synchronization Steps or Action steps. For Decision steps it is used a subset of Arden Syntax logical expressions. GLIF also uses a subset of Asbru elaborated temporal language (Shahara et al. 1998) to represent temporal constraints. The current version of GLIF is GLIF3, which is available in XML format and has a medical data model based on the HL7 Reference Information Model (RIM).

The SAGE (Standards-Based Sharable Active Guideline Environment) (Ram et al. 2004) depiction model sees guidelines as Recommendation Sets built as graphs of Context Nodes, which can be Action Nodes, Decision Nodes and Routing Nodes. This approach is the product of a collaboration of six research groups, namely IDX Systems, University of Nebraska Medical Center, Intermountain Health Care, Apelon, Inc., Stanford Medical Informatics and the Mayo Clinic, and is an effort towards a sharable format of Clinical Guidelines.

The main goal of the present work is to combine general models of human task execution (present in the several available approaches) with formal decision making models and at the same time address the issue of incomplete information in decision making.

GUIDELINE MODELING

CGs are a good way of representing medical procedures, given their clear and concise nature. They compile all facts, terms and procedures about a given disease and make logical connections between them. CGs are subjected to a thorough review before being published are updated on a regular basis. The goal is to adapt textual CGs to a digital format, where they are represented according to a depiction model based on the best features of previous models. This depiction model represents guidelines as flowcharts to achieve a better communication with the user, making it easier to globally analyze the problem at hand. This flow-chart act will as a blueprint that guides healthcare professionals through the clinical process. The automated execution of CG requires the
The development of an intelligent guideline engine, sensitive to the input of medical terms and symptoms. This guideline engine provides healthcare professionals a real-time suggestion system, maximizing the efficiency and the quality of healthcare.

The formalism proposed by this paper draws its inspiration from PROforma, one of the available depiction models for CGs. It presents an abstract view of decision making processes and task management during a clinical procedure. CGs are represented as oriented graphs where each node represents a task. Tasks are the basic unit of this model and every procedure in a clinical process is viewed as a task. The types of tasks presented in this approach are Decisions, Actions, Enquiries and Plans, which are inserted in a more general task called Root task, as in Figure 1. A CG is modeled as a Plan, which is a collection of tasks. The CG elements are the Root task, Plan, Action, Decision and Enquiry. The description of these elements is:

- **Root task** is a class of tasks that contains different Plans, according to their main orientation, namely diagnosis, treatment and clinical examinations, among others. The Root task has the following set of attributes: Name, Actions, Decisions and Initial processing and variables definition.

- **Actions** are tasks that involve a choice of any kind. For this purpose they contain different options as well as the rules that dictate which option to choose.

- **Decisions** are medical procedures that can only take place outside the system and they are executed by an external agent. The specific attributes for Action tasks are Method, Confirmation and Special conditions. The Method is the detailed description of the means required to carry out the task. In Confirmation it is specified if the Action needs permission from the user to be performed or not. Special conditions are only used when the task at hand can only be executed under a set of conditions, which are textually specified in this attribute.

- **Enquiries** are tasks with the objective of acquiring information, through questions. They are entry points for data concerning the patient state. The specific attributes of this class of tasks are Enquiry name and Data Definition. In Enquiry Name we represent the name of the parameter to be obtained and in Data Definition we define how this parameter is input in the system.

**ARCHITECTURE**

![Architecture Diagram](image)

Figure 2: The architecture of the system.

Caption, Description and Goals.

- **A Plan** comprises a set of tasks that should be executed in order to achieve a certain goal. Besides Name, Caption, Description and Goals, a Plan has attributes like Trigger Conditions, Components, Scheduling Constraints and Abort Conditions. Trigger Conditions contain the events that trigger the execution of the Plan. When these Trigger conditions hold, the tasks that have their Name in the attribute Components start executing. The order by which these tasks are executed is defined in Scheduling Constraints. This attribute is also used to define which tasks are mutually exclusive and which tasks should be executed simultaneously. If there are events that imply canceling the guideline execution, they should be put in Abort conditions. Another important feature of Plans is that they can be nested inside another Plan.

This project works with several different programming paradigms, being heavily based on logic inference paradigms. The system relies mostly in four modules: Database Modulation, User Input Parsing and the Logic Inference Matcher (Figure 2). The final goal is the deployment of a system that was undergone through a process of modeling and simulation, the final product being a polished and fully functional suggestion system, having a very low-to-none rate of errors. This is a model on which a physician/nurse has to rely, and the medical condition of a patient is at the line, thus a thorough simulation process is demanded to test all the variables for possible problems.

The Database Modulation (DM) parses the guidelines introduced in the system and generates the associations between the several elements that the guideline is composed of. It is also responsible to deliver the correct guideline when
the model requires it. This is done by an initial interactive questionnaire that is presented in the form of a triage environment, and it serves as a first filter to select the appropriate guideline(s) to load, in order to follow on to the next step, the Logic Inference Matcher. The databases used are tagged textual data with medical knowledge that is a perfect transposition of the original guideline to the computer format. The resultant archives will constitute a well-defined guideline that is constituted by procedures, actions, questions, among other information. The knowledge resultant should be able to be machine and human readable, so it can be processed by computers and understood by man (Novais et al. 2010). To withstand time and different computer architectures where the model should run, JSON and XML were adopted (Nursetiov et al. 2009). These prototypes of data interchange files suit better than the standard databases, being lightweight and very portable, having the characteristic that an internal parser can easily read the content without any external tool or application, being also human-readable.

The Logic Inference Matcher (LIM) consists of a processing engine that has rules defined in a logic programming language. The LIM is highly dependent of what the user introduces in the interface, thus it depends on the User Input Parsing module. The data delivered by the User Input Parsing is processed and matched against the factual knowledge retrieved from the database and follows a hierarchy of processes that follows the tree of decisions that emulate the decision tree of the original guideline. The LIM is based in logic paradigms, embracing Java and Prolog as main constructing languages. The guideline interpreter is developed in Prolog language, to follow the conjunctures of the required logic and to better support the implementation of Extended Logic Programming. The link between the User Input Parsing and the guideline interpreter is done in Java, it supports the Prolog connection and the connection to the interfaces, adding the fact that nearly every computer can execute it (Wielemaker, Costa 2011).

Unlike temperature or blood pressure, some symptoms are difficult to measure and can only be described through subjective values, like pain. These measures are vague and the values fluctuate given the perception of the patient towards the comprised symptom.

Having this in mind, we have adopted the Quality of Information methodology to relate the incomplete or inaccurate information with the formal information and produce answers with a confidence index associated to them (Gomes et al. 2010)(Costa et al. 2011).

The visual representation has the uttermost importance; the typical users are not much computer literate and being time and time of response a key component in a ER the interface should be as simple as possible, but, at the same time, provide all the options and inputs. The approach of this module development was towards the minimalism and follows a process of execution that emulates the human decision process. The emulation of the human process of logic elimination is, in our view, the best way to achieve the goal proposed. This process consists in having an array of possibilities, and by a process of elimination trough comparison and approximation leave only the possible items and processes. Translating to the visual form, it is a map of all the items, displaying the next item connected to the previous ones, providing an oriented graph, were the user can clearly see what was the path of the procedure and the selection that he has made.

The User Input Parsing (UIP) consists in a flowchart-like interface that has the ability of receiving various types user input and parses them in order to deliver them to the LIM (Figure 3). The interface of the system has three steps. The initial board, where the physician can see the user medical chart and general information, when he activates the Digital Clinical Guideline, a triage-like questionnaire form will be suggested for him/her to fill in, so the system can more easily select the appropriate guideline, or it can be done manually.

From there on the physician access a visual representation of the steps and actions taken, as also the data he input in the forms. This flowchart shows a time-frame of all the diagnostic processes that have been done so far and the future options that the physician has (Marcus et al. 2000) (Guyatt et al. 2000).

![Figure 3: The interface of the system.](image-url)
Figure 4: Representation of the detection of Metabolic Syndrome, a fragment from the ATPIII guideline for Hypercholesterolemia
KNOWLEDGE REPRESENTATION

Our model provides the means of dealing with incomplete information. The common problem is when a diagnosis is made and, because of insufficient data input or uncertain values, the output of the system has a weak confidence value or even provides no results. To deal with this situation an implementation of Quality of Information (Novais et al. 2010) was made to this project to provide a strong result and to manage the input provided by the user.

Knowledge representation is denoted by using Logic Programming, with a sense of strong negation. Extended Logic Programming (ELP) (Analide et al. 2006) uses classic negation as an explicit way of representing negative information, which is useful when the available information about a predicate p is incomplete and its absence does not necessarily mean its falsity. An Extended Logic Program is a finite collection of rules of the form:

\[ q \leftarrow p_1 \land \ldots \land p_n \land \neg p_{n+1} \land \ldots \land \neg p_m \]  
\[ ? p_1 \land \ldots \land p_n \land \neg p_{n+1} \land \ldots \land \neg p_m \]  

where ? is a domain atom that denotes falsity and p and q are literals. In ELP, default negation, not A (where A is a literal), represents an extension to literal A, while if it is classic negation, \( \neg A \) and A are just simple literals.

ELP programs are associated with abducible sets, which represent hypotheses for possible solutions to queries. These abducible sets are represented as exceptions to the extension of the predicates and provide a way of dealing with incomplete information. Thus it is introduced a new way of classifying information, adding unknown as a viable classification, besides true and false.

This is very useful to cases where empirical data is collected from the patient. For instance, pain is, at least, a very relative value and is almost impossible to measure with accuracy a specific value; it depends from person to person and their tolerance to pain. The fact is that the confidence that this field implies is very weak, but in some cases the only method available to classify a symptom, so Quality of Information process works to supplant the missing values, so the guideline can continue to be executed with a strong value of confidence even if a required field has dubious values.

CASE STUDY

The system starts with a triage process that consists of a set of general questions whose objective is to make an initial assessment of the health of the patient. One of those questions is about the level of Low Density Lipoprotein Cholesterol (LDL-C) of the patient. The result from this triage process shows that the patient has elevated LDL-C (about 170 mg/dL). Then, the system searches the treatment Root task for a Plan that has this Trigger Conditions. If a Plan has this Trigger Condition, its execution is suggested to the user. The Plan P1 is the one considered most suitable and is specialized in the treatment of Hypercholesterolemia.

Hypercholesterolemia is a metabolic derangement characterized by the presence of high levels of cholesterol in the blood. Elevated cholesterol in the blood is due to abnormalities in the levels of lipoproteins, the carriers of cholesterol in the bloodstream. There are three major types of lipoproteins: Low Density Lipoprotein (LDL), High Density Lipoprotein (HDL) and Very Low Density Lipoprotein (VLDL). The detection of high cholesterol is done by measuring de levels of LDL Cholesterol (LDL-C), or bad cholesterol, in the bloodstream, since this is the lipoprotein responsible for the formation of atherosclerotic plaques. Atherosclerosis may cause Coronary Heart Disease (CHD), which is the leading cause of death in many developed countries, such as the United States (National Heart, Lung and Blood Institute 2001).

Figure 4 is a fragment from ATPIII Hypercholesterolemia guideline. The fragment refers to the detection of Metabolic Syndrome, a combination of metabolic disorders that when occurring together increase the risk of cardiovascular disease. This detection is done after the patient undergoes Therapy Lifestyle Change, which includes a low fat diet, weight management and physical activity, in order to reduce LDL-C levels (National Heart, Lung and Blood Institute 2001).

The detection of Metabolic Syndrome starts with three consecutive Enquiries: E11, E12 and E13. These Enquiries are data entry points in the execution of the guideline, for the metabolic disorders that feature Metabolic Syndrome. Enquiry E11 has the purpose of obtaining the value for the waist circumference of the patient, in order to detect if he suffers from obesity. Central adiposity is a key feature of the syndrome. The following Enquiry, E12, has the aim of discovering if the patient suffers from hypertension. The last Enquiry, E13, refers to the level of fasting glucose in the bloodstream.

The next task is Decision D4, where the objective is to determine if the patient suffers or not from Metabolic Syndrome, these being the two possible options. If the patient has a waist circumference superior to 102 cm, combined with a level of fasting glucose superior to 110 mg/dL and hypertension, then the patient has Metabolic Syndrome. In this case the workflow is redirected to Plan P8, the Treatment of Metabolic Syndrome, since it is the task that has a Trigger Condition that meets the conclusion of Decision D4. Plan P8 includes Actions A8 and A9 that should be executed simultaneously. Action A8 states that the patient must undergo weight management and diet, combined with increased physical activity. Action A9 states that hypertension must be treated and the patient should use aspirin to reduce prothrombotic state, when suffering from CHD. However, if the patient does not have one of the symptoms that are essential to the diagnosis of Metabolic Syndrome, he does not have this health condition and the workflow is redirected to Action A7 that recommends the assessment of the level of triglycerides.

CONCLUSION AND FUTURE WORK

There is a growing need in healthcare to offer support in decision making processes, because the knowledge used in it, namely in diagnosis situations, can often be incoherent, incomplete and subject to error which brings out the
necessity of modeling a set of tools that enable the manipulation of this knowledge. PLE can be a valuable asset in the solution of this issue. The Quality of Information methodology enables the achievement of a measure of how this information is reliable.

The work so far focused on the organizational aspects of a medical procedure and tried to model it in an intuitive way, through classes of tasks that humans use in their everyday activities, so that it would feel as natural as possible to practitioners. This approach was developed keeping in mind integration with a logic based decision model described above. Future work includes improving the issue of uncertain information and Quality of Information.

REFERENCES


AASYS - APPOINTMENT ALERT SYSTEM: AN OPEN-SOURCE-BASED SOFTWARE TO IMPROVE SHOW RATES IN A HEALTH CARE UNIT

Hugo Peixoto
José Machado
António Abelha,
José Neves
CCTC,
Universidade do Minho, Portugal
email: \{hpeixoto,jmac,abelha,jneves\}@di.uminho.pt

António Correia
Serviços Partilhados,
Ministério da Saúde, Portugal
email: acorreia@spms.pt

Manuel F. Santos
Algoritmi Center,
Universidade do Minho, Portugal,
email: mfs@dsi.uminho.pt

KEYWORDS

ABSTRACT

In this paper we show a practical approach to deal with government impositions correlated with on time appointments in major healthcare facilities in Portugal. Reduce waiting time and improve quality on patient communication are two of the main objectives in this project. Information Systems can improve patient communication with hospitals and build a trust and better patient-caregiver relationship. New possibilities upraise with new smart phones and it is up to us to use it according to national standards and laws. We intend to establish a platform named Appointment Alert System (AASYS) that aids healthcare facilities to decrease costs correlated with calls for appointments, which indeed one of the major sources of appointment department costs. Schedule administration is taken in consideration and an user interface web page was also built. At the end we prove that an alert system based on open source software leading to short message service and email can help decrease costs and increase show rates.

INTRODUCTION

Handel & Hackman in 2010 indicate in their study that Information Communication Technology interventions can improve the efficiency of healthcare services (Handel and Hackman 2010). Hospitals could significantly improve in patient appraisal by providing a fast and effective alert and notification system by all means available and with this increase patient attendance. Improve patient assiduity cannot only reduce number of hours spent with doctors with no patients, but also increase cost effectiveness workflows. Schedules can be redesigned to improve attendance times and besides improving appointment times, human resources can be better used to benefit the institution. Studies show that there are distinct reasons for missed appointments (Defi e et al. 2010), and several misses are common in Portuguese hospitals (Montalvão et al. 2000). Centro Hospitalar do Tâmega e Sousa (CHTS) is a major hospital located in the north of Portugal and it covers over 500.000 patients from several regions near Vale do Sousa. Indeed this population led to 247.003 appointments in 2010 at a pace of almost one thousand every day, over 70 physicians per day as well as 30 administrative helpers. Therefore a rigid schedule plan is taken in consideration to avoid unnecessary losses. Nowadays Health Information Systems (HIS) capabilities are increasingly exposed and exploited. A Health Unit is computationally represented by a heterogeneous set of applications that speak different languages and are customized by different customers. So a practical and effective communication platform between information systems is paramount taking into consideration the quality of information. A study published in 2010 by ANACOM, the National Communication Authority in Portugal, reveals that are 15.929 million cell-phones were operating in Portugal for a total of 10 million population size in the last trimester in 2009 (ANACOM 2010). Adding Internet to this equation is simple to deduce what are the best means to achieve less misses, and they are: Short Message Service (SMS) and e-mail. We present the use of Mirth Connect @ to empower interoperability and integration (MIRTHCOP 2011). The impact of open source in healthcare is still limited due to corporate licenses, although several other areas are now using them (Murray et al. 2009). Jana- manchi and colleagues (Janaamanchi et al. 2009) managed to publish a study on the state and profile of open source software projects in health care and medical informatics and they state that more and more healthcare organizations tend to be sponsors of such projects and
not only pure users. In this paper we first present Information Systems and how essential they are to healthcare facilities. Next, we show how Integration and Interoperability can be the way to promote communication between information silos within a hospital. Before the workflow, results and conclusion we present how open source and web services can be implemented in healthcare and the best features of these to meet the work goals.

INFORMATION SYSTEMS

Healthcare is turning into a science based on information (Hersh 2002). In the last decade, HIS have gained great importance and have grown in quality and in quantity. As Sujansky refers the fast growth of Information Systems is due to quick expansion of Biomedical Knowledge, reduction in computing costs and spread of internet access have created an ocean of electronic data and this opens a lot of study opportunities (Sujansky 2001). Demands of information handling within the healthcare sector range from clinically valuable patient-specific information to a variety of aggregation levels for follow-up and statistical and/or quantifiable reporting. HIS have been for some years a very attractive domain for Computer Science (CS) researchers. Even more, such systems have great potential for information integration and automation, and this is an issue of study in which medicine and web service technologies and methodologies for problem solving may overlap (Hersh 2002). Furthermore, healthcare systems have to be addressed in terms of a wide variety of heterogeneous, distributed and ubiquitous systems speaking different languages, integrating medical equipment and customized by different entities, which in turn were set by different people aiming at different goals (Hersh 2002).

Integration and Interoperability

Communication, management and commerce via the Internet became a central focus for businesses, consumers and government. The actual time marks the convergence of the computer and communications industries and their associated services and products (Peixoto et al. 2010). A HIS is a set of software applications that often do not interact and create silos of information. Currently, despite the new technological advances, there is still a long way to achieve full interoperability in a healthcare unit. The various sources of information and the unique patient identification were major problems in the project development (Peixoto et al. 2010). Current implementations still lack information quality and access due to silos of information created by proprietary software and old equipment that doesn’t integrate with HIS among other known problems. Despite all the efforts full interoperability is still to come and until that point a lot has to be done and there is still a lot of road to travel. The lack of integration between the different HIS is not only an obstacle for a more effective clinical practice, but it may lead to a suboptimal care for the patient. In this project the main goal was to run into governmental laws and integrate all medical and chirurgical appointments in one single alert system, which reduces costs to the institution and time for administrative tasks. Controlling every appointment implies a huge effort, and several variables should be taken in to consideration. Healthcare Information Systems have become an important helper in these tasks and flawless systems are easy to build, preventing bad procedures and amending potential mistakes. Fig. 1 shows the integration model adopted to solve this task. Integration between medical schedules, chirurgical schedules and free spaces is displayed on user friendly web page. A final structured database that results from user interaction with the system is used to feed the main open source software Mirth Connect and alert patients. The alert system enables one to create, send and retrieve messages online. It may be very useful for handling data, images or even file exchange. Encryption and the right protocols of trading are also paramount (Machado et al. 2008). Alert systems are extremely important not only for the internal workflow in a healthcare institution, but as well as an essential component for the development of group work (Weed 1971).

OPEN SOURCE SOFTWARE

Open source software, can be characterized by collaboration among individuals and organizations with common interests, shared intellectual property, and a commitment to standards. The open source model includes the concept of concurrent yet different agendas and differing approaches in production, in contrast with more centralized models of development such as those typically used in commercial software companies. The main principle and practice to development is peer production by bartering and collaboration, with the end-product, source-material, blueprints; and documentation available at no cost to the public. This is increasingly being applied in other fields of endeavor, such as biotechnol-
ogy. Taken globally several healthcare facilities with the same goals can share information and goals. Open Source applications suffer from several additional barriers to the general Information Technologies implementation issues. They include lack of professional knowledge, as well as, executives lacking knowledge about the benefits of open source. Krogh and Hippel in 2006 presented the renewed urgency to adopt health and medical informatics applications and how open source approaches are gaining attention in the health care industry (Krogh and Hippel 2006). Information technology in the health care industry is evolving from an administrative tool for billing and bookkeeping to a clinical tool for improving the quality and efficiency of health care, and the scope of information sharing is expanding beyond the walls of individual institutions. Achieving this level of integration will require that software models overcome a host of technical obstacles, and that they are accessible, affordable, and widely supported. While not heralding the end of commercial software vendors, conditions are ripe for open source solutions to take root in health care, and that it will likely become the standard for capturing, sharing, and managing patient information to support quality care. It also notes that health care businesses have the opportunity to take the lead and drive the shift to this new model (Gould and Brown 2006). The main motivations for using Open Source software in healthcare are:

- reduced total cost-of-ownership;
- faster delivery of systems;
- systems being more secure;
- elimination of vendor lock-in;
- control over the software (possibility of adapting to local needs).

As Murray describes Information Systems have grown in a structured way and are now considered as being part of good and patient oriented healthcare unities (Murray et al. 2009). For example, Care2 X, an open source project is one of the several projects ongoing in healthcare. This project is based in four main areas and relies on open source standards. An Electronic Medical Record is under development in Europe and may well be one big substitute to commercial approaches, taking the best of open source standards (Care2x 2010).

**AASYS OVERVIEW**

One of major tasks to accomplish the desired goal was to collect several phone numbers among the population. New technologies are hard to implement but proximity actions were performed. In order to put this system to work, several public collections of correct demographic data were executed. Adding to these actions was instituted internally, that in every single appointment in 2010 the administrative crew had to complete the needed information of the patients or correct information that could lead to future errors in the system. After this first task was made from a total of numbers 163895 in January 2010 we managed to collect 285560 phone numbers in January 2011. On the other hand, 120739 emails are now correct where in 2010 there were only 80655 correct. A legal issue comes up when providing Short Message Service contact number since every patient should decide whether or not he or she wants to receive this kind of alert. In order to have control of this situation it was necessary to change the basic HIS and provide an extra field in database. This way, patients can provide a different number for SMS service and an emergent contact number when needed. Email was also one of the alert systems to be designed and although Internet is becoming part of everyone’s life for elderly people it is still hard to find an email contact.

Every day over 500 letters with appointments calling are issued to the post office meaning costs with paper, envelop and post fees. All this work is just affordable when all appointments are concluded otherwise all the money spent in that call will go to waste and nothing can take it back. Fig. 2 shows the global approach and workflow followed to achieve the final goal. The framework used was Mirth Connect® open-source software build in JAVA. Mirth Connect was used because is open source and gives correct procedure to all that was needed. Mirth is used, not only, to build information in a structured database to contact patients, but also to contact a commercial Web Service that provides a SMS service. In Fig. 2 the first databases are part of the HIS and combine patient demographic information such as SMS phone contact and email, schedule appointments for the day, doctors in scale to give medical care and chirurgical treatment. Having xml message service of Mirth working on the background it is possible to collect all the available data and present it in a user interface friendly web page. An administrative team can send in only one step all SMS and email for a specific day or, if wanted or needed, SMS and email can be sent to a certain patient. In the final step, web
services provided by a mobile phone company delivers the short message service that alerts the patient. As Dogac stated Web services in healthcare domain brings many advantages such as, providing interoperability of medical information systems through standardizing the access to data through WSDL and SOAP and extend the healthcare enterprises by making their own services available to others (Dogac et al. 2006).

To send a SMS to patients the phone company provided us a web service that only needs a few parameterizations to work. The major variables are \{user\}; \{pass\}; \{name\}; \{type\}; \{sms.content\}; \{sms.number\}.

On the other hand sending the email requires only an exchange domain, which was already configured in CHTS for internal and external email service to workers. SMTP is contacted providing only the basic email data: \{email.from\}; \{email.to\}; \{email.subject\}; \{email.content\}.

Whether in SMS or email alert a log database is kept and send data and time is saved. With this log database is possible to track potential errors. For this purpose two new agents were built in Mirth Connect. The first agent controls the number of email sent and provides an email and SMS alert to Administrative Chief and to Information System Project Leader. The second agent has the same goal but controls the number of SMS sent.

RESULTS AND DISCUSSION

AASYS was implemented in January 2011 and all the results presented bellow correspond to the first semester of 2011. In order to have some comparison the homologue period of 2010 was used. The total number of appointments scheduled from January until May in 2010 was 107.076, while in 2011 the total of appointments was 111.508. Looking at this number is possible to determine an increase of appointments, meaning the effort made to decrease waiting lists. To evaluate the results of the implemented systems was necessary to determine whether the show rate increased or decreased along the first six months of the year 2011 compared to the same period in 2010. Another important issue that was taken in consideration was if the missed appointments were canceled in time to change to other patient that needed that appointment. Table 1 shows the cancellation time of the appointments for the two periods.

<table>
<thead>
<tr>
<th>Cancellation time</th>
<th>2010</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before appointment</td>
<td>13.908</td>
<td>16.565</td>
</tr>
<tr>
<td>Same day as appointment</td>
<td>1.077</td>
<td>1.034</td>
</tr>
<tr>
<td>After appointment day</td>
<td>1.068</td>
<td>520</td>
</tr>
</tbody>
</table>

Figure 3: Number of canceled (b) and missed (c) appointments

As proved by Table 1, the number of cancellations before the appointment day has increased from 13.908 to 16.565. This result lead to new reschedules and helped substantially to reduce waiting lists, since new appointments were scheduled and other patient could take the appointment time instead. Side by side, since no free spaces are available, doctors, nurses and administrative time is monetize leading to better cost control. After appointment day cancellations decreased by 50% and this can already provide the final predicted results to this work and how well AASYS managed to deal with schedule and show rates problem. In 2010 a total of 103.121 appointments were actually performed, while in 2011 the number raised to 109.577. Fig. 3(a) presents the total number of canceled appointments and Fig. 3(b) presents the total of misses to the appointments. As result shows, there was an almost 80% reduction in misses from one year to the other, which is a good indicator of the good performance of the AASYS. Besides this good performance, there are several other aspects to be taken into consideration that could significantly affect the expected results. Indeed a better approach to users should be taken in consideration, since elderly population cannot undertake these changes as well as the younger one. Great effort was taken in the last semester of 2010 to collect as many phone numbers and emails as possible, however for a total of nearly 1.000 appointments per day, the medium number of SMS was 400, and 500 emails. Increasing correct phone numbers and emails is one of the major improvements that can be taken into consideration. No major problems with the performance of the agents were experienced since MIRTH CONNECT® based agents are pretty stable for current use. The major difficulty was to contact the phone company web service since healthcare units are within a closed LAN. Additionally, AASYS reduce 30% a day, the number of letters sent to post office to be deliver to patients. From a 0,30€ letter cost, just for post office fees, The AASYS platform managed to deliver the same information for patients for just 0,06€ per message. Faster, safer, law compatible and cheaper system was achieved when AASYS was installed at CHTS appointment department.
CONCLUSIONS

In this paper, we present a prototype for Appointment Alert system in a healthcare unity based on integration, interoperability and open source software. The AASYS is one of the objectives of a much more ambitious multidisciplinary effort of Centro Hospitalar do Tâmega e Sousa and University of Minho group of Artificial Intelligence, known as GIA. Such project seeks to fully integrate and interoperate a healthcare facility using known standards or applying new ones. Although the software presented is still at the initial stages of development it was capable of providing better performance at schedule cancels and cost reduction in CHTS. The current version of the AASYS serves as a basis for future implementation that will focus on end-to-end communication allowing users to contact the care provider via SMS and email.

ACKNOWLEDGEMENT

We would like to thank to all the crew in Centro Hospitalar do Tâmega e Sousa in Penafiel for their precious ideas, help and patience.

REFERENCES

ANACOM, 2010. ANACOM - Mobile telephone service has 13.9 million users in 2009.

Care2x, 2010. Care2x - The open source hospital information system.


BIOGRAPHIES

HUGO PEIXOTO is currently pursuing doctoral studies in Medical Informatics, at the University of Minho, focused on semantic interoperability and health records in healthcare units. He is now working at Centro Hospitalar Tâmega e Sousa where he is a software developer.

ANTONIO CORREIA worked in Centro Hospitalar do Tâmega e Sousa, in Penafiel, Portugal, in the Information System department until last year when he joined share services of the Portuguese Health Ministry.

MANUEL F. SANTOS holds a PhD in Computer Science - Artificial Intelligence. Actually he is associate professor in the department of information systems, school of engineering, University of Minho, Portugal. His main interests are: Knowledge Discovery from Databases and Data Mining, Intelligent Decision Support Systems, Machine Learning.

JOSÉ MACHADO, ANTÓNIO ABELHA AND JOSÉ NEVES hold a PhD in Computer Science - Artificial Intelligence. They are now Professors at the Department of Informatics of the University of Minho. Their research area spans the field of Biomedical Informatics, Databases and Intelligent Systems, in particular when applied to Medicine.
PERFORMANCE MEASURING OF A PAEDIATRIC OPERATING THEATRE: A DISCRETE EVENT SIMULATION APPROACH

Latifa DEKHICI
LAMOSI, USTOMB
University Mohamed Boudiaf
BP. 1505 Oran M’Naourer,
31000 Oran, Algeria
dekhicilatif@gmail.com,
Alain TANGUY
LIMOS, CNRS UMR 6158,
University Blaise Pascal,
Les Cézeaux, 63173 Aubière
Cedex, France
tanguy@isima.fr
Khaled BELKADI
LAMOSI, USTOMB
University Mohamed Boudiaf
BP. 1505 Oran M’Naourer,
31000 Oran, Algeria
belkadi@univ-usto.dz

KEYWORDS
Discrete Event Simulation, SIMULA, Operating Theatre, Staffing.

ABSTRACT
The work presented in this paper mainly aims the modelling and the design of a paediatric operating theatre of Canastel hospital in Oran in order to determine the number of technical and human resources according to the number of operating rooms. For this we use the ASDI methodology (Analysis-Specification-Design-Implementation) which has the principle of independence between the specification in a knowledge model and the simulation languages for the implementation of action models. We use the ARIS toolset for the specification. The simulation of 4 models over 16 scenarios is realized with the SIMULA class GPSSS. It helps us to find a best configuration that maximizes the effectiveness of operating rooms and meets quality criteria that we have laid out.

INTRODUCTION
The operating theatre consists of operating rooms (OR) and Post Anaesthesia Care Unit (PACU). This latter includes awakening bed. The patient to be operated must go through several steps including the preparation, induction, surgery and recovery. The occurrence of different types of physicians in each step may depend on the act specialty but, unfortunately, can be influenced by unavailable resources or delays. There are many optimization techniques for the operating theatre such as mathematical programming (Perdono et al., 2006; Pham and klinkert, 2008), simulation (Bowers and Mould, 2004; Dexter et al., 2000), analysis methods (Van Houdenhoen et al., 2007; Lovejoy and Lie, 2002), heuristics (Chaaban et al., 2006, Denton et al., 2003) and metaheuristics (Black et al., 2002; Hsu et al., 2003).
We use the ASDI methodology (Analysis, Specification, Design and Implementation) to model the different processes, their actors and to simulate the main process in order to optimize the work of human resources and the utilization of operating rooms. We present, in the second section, the used methodology, its principle, its advantages and its steps. In the third section, we give an example of the knowledge model which is the first step. Finally, we describe action models (simulation model implementation) and performance measuring of existing and proposed models.

METHODOLOGY
So as to model, simulate and design the surgical theatre of the paediatric Hospital of Canastel, we use the ASDI methodology. It has been proposed for production systems by (Gourgand and Kellert, 1991) used by (Ruch, 1994; Goujon, 1997; Lacomme, 1998) adapted to hospital systems in (Combes, 1994; Tanguy et al., 2010; Belkadi and Tanguy, 2009; Gourgand et al., 2005, 2006, 2007; Mebek et al., 2007) and used for the operating room and the Emergency department of a new hospital by (Lamiri et al., 2008, Chauvet et al., 2006). The methodology approach integrates recent works in enterprise modelling defined by P. Fenies (Chabrol et al., 2006).
This methodology is known for the independence between specification and design steps. So we can use the same specification for several design objectives: performance measuring, design, planning (Lamiri et al., 2008) and scheduling (Dekhici and Belkadi, 2010). The systemic view of the decompositon of an industrial system can improve the design, understanding and scalability of simulation models.
In the first step a knowledge model (Gourgand and Kellert, 1991) allows us to define the operating theatre elements (surgical teams, resources…), the interactions between them (organization) and the dynamics of the system (patient flows).
This model consists of three subsystems: the physical subsystem, the informational subsystem and the decision-making subsystem.
An action model is a translation of the knowledge model in a mathematical formalism or programming language (simulation language in our case) for the performance measuring. Several action models can be built from the same knowledge model.

KNOWLEDGE MODEL
Canastel Paediatric Operating Theatre Data
The Canastel Paediatric operating theatre contains a PACU (post anaesthesia care unit) of 4 beds, an ICU (Intensive Care Unit) of 2 beds and 4 OR (Operating Room) that ordinarily is in use 7 hours/day and a maximum of 9 hours. The 4th OR is currently closed because 3 ORs are enough. The number of human resources are: 4 surgeons (SUR) per day2 NA (Nurse’s Aides), 2 OTN (Operating Theatre Nurses) and 3 AN (Anaesthetist Nurses). The operating theatre is currently using one ARD (Anaesthetist Resuscitator Doctor) at the induction phase. It is released before the beginning of the
intervention. Officials concede the non-compliance with the legislative procedure for resuscitation monitoring. They accept the reinstatement of another ARD from another service if this will enhance the efficiency of OR and increase the number of surgeries that can be planned and treated. The collected data are described in Table 1.

### Table 1: Data of Surgeries in Paediatric operating theatre.

<table>
<thead>
<tr>
<th>Surgery Type</th>
<th>Surgery</th>
<th>Nb./Month</th>
<th>Surgery duration</th>
<th>Waking duration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Mod</td>
<td>Min</td>
</tr>
<tr>
<td>Orthopedic</td>
<td>Pin removal</td>
<td>4</td>
<td>80</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Bone biopsy</td>
<td>4</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Cast boot</td>
<td>16</td>
<td>35</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Epiphysiology</td>
<td>3</td>
<td>180</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>Polycicadacty</td>
<td>12</td>
<td>150</td>
<td>105</td>
</tr>
<tr>
<td></td>
<td>Chibfoot</td>
<td></td>
<td>60</td>
<td>45</td>
</tr>
<tr>
<td>Urology</td>
<td>Circumcision</td>
<td>8</td>
<td>30</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Endoscopic</td>
<td>6</td>
<td>45</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Bladder</td>
<td>3</td>
<td>180</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>extrophy</td>
<td></td>
<td>90</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>Vesicular</td>
<td>3</td>
<td>90</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>diverticulum</td>
<td>7</td>
<td>60</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Hynosphindus</td>
<td>7</td>
<td>90</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Ant.</td>
<td></td>
<td>45</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Botox injection</td>
<td>10</td>
<td>50</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>Splenectomy</td>
<td>6</td>
<td>90</td>
<td>60</td>
</tr>
</tbody>
</table>

**ARIS Model**

An example of modelling with the ARIS toolset is presented in figure 1. The diagram specifies the global process.

More detailed diagrams describe the processes and the events in the event driven process chains.

### FROM KNOWLEDGE MODEL TO ACTION MODELS

To pass from the knowledge model to the action model, we use a queuing network model (figure 2) describing the processes delays, the resources seizing and releasing.

**Input Setting**

The model characteristics are parameters such as the numbers of beds in PACU, some human resources, the number of incoming patients in a day and the system configuration. We collect the surgery types and their numbers during several months, in order to calculate the average per day. For each surgery type, we collect the minimum, maximum and most frequent wake up durations. To classify multiple surgeries, we calculate the weighted average of minimum, maximum and frequent durations of each class. We use the minimum, mode and maximum operating duration in a triangular law (Dekhici and Belkadi, 2009) and the duration of early and complete waking in a uniform law. One feature of the system is that a patient, for whom a bed cannot be allocated in the PACU, is blocked in the OR and he starts to wake up. Averages of durations required by surgery type are shown in Table 2.

### Table 2: Duration Averages of Surgery Classes (per hours)

<table>
<thead>
<tr>
<th></th>
<th>Orthopaedic</th>
<th>Urology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surgery</td>
<td>Nb./day</td>
<td>1,346</td>
</tr>
<tr>
<td>Mod</td>
<td>83,255</td>
<td>60,164</td>
</tr>
<tr>
<td>Min.</td>
<td>4,728</td>
<td>34,819</td>
</tr>
<tr>
<td>Max.</td>
<td>106,511</td>
<td>89,682</td>
</tr>
<tr>
<td>Waking up</td>
<td>Max.</td>
<td>20,488</td>
</tr>
<tr>
<td>Min.</td>
<td>14,651</td>
<td>16,754</td>
</tr>
</tbody>
</table>
ACTION MODEL

To implement the action model we use the SIMULA language with the GPSSS external class. The purpose of this simulation is to find the right operating process configuration, the reports of number of beds/OR and number of physicians/day to obtain an ideal OR occupancy time and to increase the number of treated patients per day.

Performance Indicators

To measure and evaluate the system performance, we use qualitative and quantitative performance indicators that are: the effectiveness of technical and human resources, the rate of treated patients, the overflow rate on the operating rooms occupancy and the feasibility of process.

Model 1

The first model M1 simulates a system with 3 common OR that use one ARD until the end of induction step. The confidence intervals are obtained for a 99% risk and 100 replications. In the first scenario M1, the system treats 5 patients: 3 in urology and 2 in orthopaedics (Table 3). We note the low utilization of some resources such as the PACU and ARD (Table 5). The occupancy time in OR is 194 minutes so about 3 hours.

<table>
<thead>
<tr>
<th>Table 3: Treated patients, model 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scen.</td>
</tr>
<tr>
<td>MIS1</td>
</tr>
<tr>
<td>MIS2</td>
</tr>
<tr>
<td>MIS3</td>
</tr>
<tr>
<td>MIS4</td>
</tr>
<tr>
<td>MIS5</td>
</tr>
</tbody>
</table>

In the second scenario M2S2, we increase the number of patients to 4 orthopaedic patients and 5 of urology. In M2S3, we take 5 of Orthopaedic and 6 of Urology and in M2S4, 6 Orthopaedics and 7 of Urology. In the latter scenario we note that there is a good use of resources but there is an uncertain risk that a patient cannot be operated. In the scenario M2S5, treatment of 14 patients including 7 of Urology is impossible because the OTN utilization rate exceeds 87% (table 4) and patients may not be treated. The values in bold denote a malfunction.

Model 2

The second model M2 concerns a system containing an OR dedicated to orthopaedics, another to urology and one shared. It uses a single ARD at the time of induction only. In the first scenario, the system treats 3 urology patients and 2 orthopaedic (Table 6). We note the use of some resources such as the PACU and ARD. The occupancy time SO is equal to 211.220 minutes so 3 hours and half (Table 7). In the second scenario we increase the patient number to 4 orthopaedics patients and 5 in urology.
407


Table 11: Treated patients, model 4

<table>
<thead>
<tr>
<th>Scen.</th>
<th>In</th>
<th>prepar ed</th>
<th>Indus</th>
<th>Oper ed</th>
<th>In PACU</th>
<th>Not yet out</th>
</tr>
</thead>
<tbody>
<tr>
<td>M4S1</td>
<td>e</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
<td>0.052</td>
</tr>
<tr>
<td></td>
<td>e⁺</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>0.092</td>
</tr>
<tr>
<td></td>
<td>e⁻</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Avg</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>0.04</td>
</tr>
<tr>
<td>M4S2</td>
<td>e</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.110</td>
<td>0.110</td>
</tr>
<tr>
<td></td>
<td>e⁺</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>10.89</td>
<td>10.89</td>
</tr>
<tr>
<td></td>
<td>e⁻</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>10.67</td>
<td>10.67</td>
</tr>
<tr>
<td></td>
<td>Avg</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>10.78</td>
<td>10.78</td>
</tr>
<tr>
<td>M4S3</td>
<td>e</td>
<td>0</td>
<td>0.052</td>
<td>0.086</td>
<td>0.142</td>
<td>0.142</td>
</tr>
<tr>
<td></td>
<td>e⁺</td>
<td>12</td>
<td>12.01</td>
<td>11.96</td>
<td>11.60</td>
<td>11.60</td>
</tr>
<tr>
<td></td>
<td>e⁻</td>
<td>12</td>
<td>11.90</td>
<td>11.79</td>
<td>11.32</td>
<td>11.32</td>
</tr>
<tr>
<td></td>
<td>Avg</td>
<td>12</td>
<td>11.96</td>
<td>11.88</td>
<td>11.46</td>
<td>11.46</td>
</tr>
</tbody>
</table>

Summary of Discussion

We simulated several configurations of the paediatric operating theatre common operating rooms or dedicated ones to urologic and orthopaedic specialties. To comply with legal surgery process and thereby increase the medical care quality, we propose to add one ARD that will be held throughout the surgery. We also opt to add one OTN, so as to reduce the use of all such physicians. The Model 3 was able to demonstrate all the qualities and quantitative performances keeping the same utilization rate and Operating Rooms occupancy times.

CONCLUSION

In this paper, we presented a modelling of the mono disciplinary paediatric theatre using the ASDI methodology. We use the ARIS toolset to define processes, material resources and personnel involved in surgery. The simulation of SIMULA models has allowed the identification of resource bottlenecks, numbers of resources and design needed to reach qualitative and quantitative aims. Our current work consists in the comparison of SIMULA model results with those implemented with ARENA Rockwell tools.

REFERENCES

Belkadi K., A. Tanguy Modelling and Simulation of the Stomatolgy Service for the RMUHO. ESM’2009.

Dekhici L., K Belkadi. Operating Theatre scheduling under constraints. Journal of applied sciences, 10(14), 2010
Lueguy W.S., Y. Li, Hospital operating room capacity expansion, Management Science 45 (11) 1369–1387, 2002
Marcon E., F. Dexter, An observational study of surgeons’ sequencing of cases and its impact on post anesthesia care unit and holding area staffing requirements at hospitals, Anesthesia and Analgesia 105 ( ) 119–126, 2007
Object trajectory simulation - An evolutionary approach

Pedro Gonçalves, Luciano Alves, Tiago Sá, Cesar Quintas, Miguel Miranda, António Abelha and José Machado
Universidade do Minho, CTC, Departamento de Informática, Braga, Portugal
pgoncalves@di.uminho.pt {pg18398,pg18591}@alunos.uminho.pt cesar.quintas@chporto.min-saude.pt {miranda,abelha,jmac}@di.uminho.pt

KEYWORDS
Data Mining, Evolutionary Intelligence

ABSTRACT

The ability to successfully predict the trajectory of an entity can have numerous interests. Using trajectory prediction we propose to enhance Radio-Frequency IDentification embedding intelligent behaviours that allow these systems to improve their accuracy in the detection and guidance of personal in RFID enabled infrastructures. This paper proposes a representative approach to a simulated area filled with sensors and travelled by an object. The object has an initial and end points and does random movement between them. The remaining unknown path must be provided by the sensors and the prevision module while error metrics must be calculated dynamically to help the prediction of the followed path. In this scenario, a trajectory is a path that an entity follows through space between iterations, and it can be represented as a set of coordinates sorted over time. The level of accuracy needed for the prediction model and objectives of this environment required a grid like representation. The presented solution is a multiple dimension structure that covers the environment variables and entities of a move within the environment. An application has been developed to simulate a censored place driven by a random trajectory object and to calculate as accurate as possible the path of the object.

INTRODUCTION

In (Machado et al. 2010), authors presented an interoperation platform for active monitoring in healthcare environments. In large-scale healthcare environments, control and spatial monitoring of patients, visits, staff, service providers, medication and equipments are extremely complex and generally do not respond to the needs, nor offer information on time for decision support. The solution is to implement a platform in order to implement integration of data and interoperability between different hospital information systems. It is the first step to have information with quality on time (Machado et al. 2008). This platform should be conceived for allowing the maximal integration and interoperability and must have a set of primitives, such as a web interface for management and administration and a set of web services for sending or receiving information, as well as for querying. On the other side, in Medicine a great effort has been made in the area of standards and normalization. The received information must be extracted, transformed and loaded in order to facilitate the information processing and the inference mechanisms. Security and authentication in message transport should be also object of great interest. This platform has been developed and is now operating in some Portuguese hospitals, being in charge of integrating information from several information sources and managing interoperability with different suppliers. One of the main partners is an electronic health record application where the medical and clinical information about patients is available.

The main goal of a Simulator is to reproduce behaviours, phenomena and sensations that are not happening. This kind of technology is used in many contexts as education, science, army, industry and video games. Its aim is to correct existing errors and prevent disasters that come with real life testing. The topic of trajectory prevision can be treated in a simulating context, saving time and money and optimizing resources.

The ability to successfully predict the trajectory of an entity can have numerous interests. Using trajectory prediction we propose to enhance Radio-Frequency IDentification embedding intelligent behaviours that allow these systems to improve their accuracy in the detection and guidance of personal in RFID enabled infrastructures. Radio-frequency identification (RFID) uses non-contact data access mechanism to read or write tag data through the radio frequency transmitted by the readers. The disposition of readers is of the essence for the performance of the system, however it depends on the experiences of the deployment engineers especially for the active readers (Hsu and Yuan 2011, Chen et al. 2010, Ngai et al. 2009). A simulation environment where not only one can test detection pathways for localization of the readers, but more importantly a prediction model to detect anomalies in detection events due to these readers localizations presents great advantages.

This paper proposes a representative approach to a sim-
ulated area filled with sensors and travelled by an object. The object has an initial and end points and does random movement between them. The remaining unknown path must be provided by the sensors and the prediction module while error metrics must be calculated dynamically to help the prediction of the followed path. In this scenario, a trajectory is a path that an entity follows through space between iterations, and it can be represented as a set of coordinates sorted over time. The level of accuracy needed for the prediction model and objectives of this environment required a grid like representation. The presented solution is a multiple dimension structure that covers the environment variables and entities of a move within the environment.

The developed prototype uses a Model-view-Controller and is based on the Java programming language. For the implementation of the graphical interface the features provided by Swing toolkit were harvested to enhance the visual effects of the prediction model. The prediction model was developed using the WEKA Data Mining API and the knowledge extraction tools it provides.

In this paper, the simulating environment will be presented, as well as the object trajectory and the path prediction. Some conclusions and the future work will be added.

**AMBIENT INTELLIGENCE IN MEDICINE**

Ambient Intelligence (AmI) is a multi-disciplinary area which integrates the experience from ubiquitous computation, ubiquitous communication and intelligent systems. The systems that follow AmI definition are heterogeneous, integrated, mobile and distributed. They are based in spontaneous and pro-active communications and have non-conventional, intelligent and adaptive interfaces.

Users are aware of their presence and context and digital environments are sensitive, adaptive, and responsive to needs, habits, gestures and emotions. In health care environments, they can not be separated from medical informatics, biomedical informatics or bio-informatics, aggregating electronic health records, decision support, telemedicine, knowledge representation and reasoning, knowledge discovery and computational biology. Radiological films, pathology slides and laboratory reports can be viewed in remote places. Remote robotics is used in surgery and telemedicine is becoming popular.

In a previous project, our group has developed an agency for interoperability in hospital environments, allowing the communication and knowledge sharing between several hospital information systems. This agency, called AIDA, is now being used in several Portuguese hospitals. It is like a symbiont, with a close association with core applications, namely the Picture Archive Communication System, the Radiological Information System and the Electronic Health Record Information System, that are built upon pro-active agents that communicate through message passing (Weiss 1999, Erl 2005). Electronic Health Record (EHR) is also a core application which covers horizontally the health care unit and makes possible a transverse analysis of medical records along the several services, units or treated pathologies. Beyond the organizational, functional, technical and scientific requisites, it must be taken into consideration ethical and legal issues, as well as data quality, information security, access control and privacy. An EHR is an assembly of standardized documents, ordered and concise, directed to the register of actions and medical procedures; a set of information compiled by physicians and others health professionals; and a register of integral facts, containing all the information regarding patient health data, and a follow up of the risk values and the clinical profile. Medical and clinical applications are usually used for discrete clinical and medical activities in specific areas and services, in particular diagnostics and pathologies. Indeed, in a telemedical information society the collection of vast amounts of medical data will not only support the requirements of archiving but also provide a platform for the application of data mining and knowledge discovery to determine possible medical trends and the real data to support educational training. Knowledge discovery techniques can be applied to identify pathologies and disease trends. The data can also be used for educational and training purposes because maybe one of the unique cases can be identified and used in expert system like applications to advise practitioners.

Medical and clinical applications are usually used for discrete clinical and medical activities in specific areas and services, in particular diagnostics and pathologies. The EHR application connects fragmented applications that when linked with services, shares a new space and merges physical and virtual worlds, enabling the calling of ambient intelligence practices in healthcare units. New opportunities and technologies make EHR attractive to medical environments in terms of the telemedical information system. But EHR will not be successful without physician, nursing and administrative people involvement. An active monitoring would not be possible to be successful without AIDA and EHR. A EHR application belongs to the AIDA suite, being also used in Centro Hospitalar do Porto, in Oporto (Machado et al. 2007).

**IMPLEMENTATION**

As described above, the implementation of the proposed trajectory prediction system is based on the Java programming language. Its robustness and scalability provide a set of numerous and broad APIs such as Swing, j-antigrav or Weka (Frank et al. 2005), which are essential to the core of of the prediction model itself. In the front-end of the application, Java-Swing is used as the
primary Java GUI widget toolkit, which facilitates the creation of the graphic interface as it offers a high level of interaction and visualization of the trajectory detection while having no problems of incompatibility with the underlying libraries.

Anti-gravity movement is a highly flexible technique that we use to generate random movements and simulate trajectories that are hard to predict using pattern-analyzing techniques. It works by allowing to definition of particular points designed called gravity points placed on the environment and it pretend to avoid.

In order to keep the knowledge base required for the knowledge extraction it was used an ORACLE database, which guarantees security, reliability and trust. In addition to using the technologies described above, we also use WEKA, which gives the necessary API to data mine the information to generate the prediction model.

MODEL AND ARCHITECTURE

Domain Model respects Model-view-Controller architecture, the actual default pattern in Software Engineering. This architecture is divided into three layers, first, named Model is the application’s data and computation, second the View represents all the graphic interface modules and the third is Control that links graphic interface with the data. This method represents a clean way of application making and prevents errors in implementation phase. The application is built around several intelligent agents:

- Business representing the Control layer
- Matrix, Object and Forecasting representing the Control layer
- Board representing the view layer

GRAPHIC INTERFACE

Being the graphic interface, the main environment, that the user have to observe all the process running, this have been developed carefully. Awareness that to whom use the software in a intuitive way understand the results that it returns. The graphic interface is represented by:

- BOARD, contains the coordinates of the X and Y axis and information about the present sensors.
- Robot Object ROBOT, runs throw the board from a beginning point to a ending point.
- Control CONTROLLER, generates points with repulsive gravitational forces to move the object on the board with a random movement, one of the generated points have a attractive forces, that leads the object to the desired ending point.

OBJECT TRAJECTORY

During the simulation, the object is attracted to the final point and, at the same time, random points repel the object in order to generate an unpredictable movement along the platform, this method is achieved by using gravitational points. Meanwhile Tabuleiro Graphic Interface is updated in order to make user aware of the object route.

At the end of the trip prevision module takes place, it is possible using information of the last executions.
Past prevision attempts information are combined in a structured and clean representation an used to guess the object path. Finally last execution output is saved, allowing the evolution of the Algorithm.

DATA MINING AND ERROR TREATMENT

The error treatment is the main part of this work. As mentioned before the way of resolving this problem is using Data Mining tools. Data mining is so the base for calculating the trajectory of the object, without it the, only information available was the points where the object have passed throw the sensors. This methodology, allows to, 
figure out where are going to be, the most probable points where the object are going to pass. Consequently this kind of approach, improves the accuracy over the course of attempts. Data mining have a lot of process of extracting and analyzing data, the one that we have used is CLUSTERING, and the algorithm is known as SIMPLE and K MEANS.

This algorithm returns the most likely clusters of points based on the average of the data. The past data are saved in a structured dataset where each coordinate corresponds to an attribute. In each iteration the path is stored to be used in subsequent iterations.

It only calculates the number of points representing the difference between the path and the points already calculated when passing by radars. At the end are shown up the points, that should be foreseen and have not been, the calculated by the sensors and by the prediction algorithm CLUSTERING. From that data mining we can analyze all the route that the object have made. That route made by the object throughout all is course, and the analysis described before the user can observe:

- Coordinate where the object was expected passing (planned by sensor or by CLUSTERING)
- Coordinate where the object was expected passing but did not (planned by CLUSTERING)

In conclusion, it is calculated, the accuracy of the algorithm, the number of correct predictions and ultimately the number of erroneous predictions. Precision is a metric defined by dividing the number of correct predictions by the total number of forecasts. In its turn, right and wrong predictions are calculated by iterating the path made and the path estimated.

These values and predictions described above are stored in the database, to be able to compare and improve the development of forecasts later.

CONCLUSIONS AND FUTURE WORK

Along the project many subjects as Data Base Modeling, object oriented programming, interface building and the most important construction of an algorithm able to predict behaves in a simulated ambient were approached. All these points have met with relative success, especially prevision module, the most time expensive phase.

The implementation problems are explained below. The prediction branch is a well studied issue, therefore born a challenge, to choose between a vast branch of available tools, Clustering Data Mining method was the chosen option. These algorithm has shown less accuracy in too dynamical data.

The object trajectory pushed by gravitational points proved to be very efficient, although a little problem, the object can pass tow times in the same coordinate, it
means that implementations of statistics became harder. As future work, another prevision algorithms must be tested and implemented as a mean of outline the data that we are dealing. Complementarily error metrics can be added and the object movement improved to configure more accuracy to the application.

BIOGRAPHIES

Pedro Gonçalves, Cesar Quintas and Miguel Miranda are PhD students in Biomedical Engineering (Medical Informatics), in University of Minho, Braga, Portugal. Luciano Alves and Tiago Sá are MsC students of the department of Informatics of University of Minho, in Braga, Portugal. António Abellha and José Machado are professors at the Department of Informatics, Universidade do Minho and researchers of CCTC (Scientific Center for Computation Technologies).

REFERENCES


Towards Intelligent Drug Electronic Prescription

Júlio Duarte¹, João Neves¹, Alexandra Cabral¹, Marco Gomes², Vitor Marques²,
Manuel Filipe Santos³, António Abela¹ and José Machado¹

¹CCTC, Universidade do Minho, Braga, Portugal
{jduarte,joaoneves}@di.uminho.pt
alexandra@chporto.min-saude.pt
{abelha,jmac}@di.uminho.pt

²Departamento de Informática, Universidade do Minho, Braga, Portugal
{pg18373,pg17203}@alunos.uminho.pt

³Algoritmi Center, Universidade do Minho, Guimarães, Portugal
mfs@dsi.uminho.pt

KEYWORDS – Electronic Health Record, Drug Electronic Prescription, Case Based Reasoning, Ontologies, Terminologies.

ABSTRACT — The errors associated with prescription drugs are common. The technology will help to reduce the error. The transition from the traditional method of prescription (written manually and on paper) for electronic prescribing of drugs has been done in developed countries. However, there is still some lack of efficiency. Some of the inefficiencies in the method of electronic prescribing are related to the interface of these systems, based on forms. These systems are still unwise and less useful as an aid in decision making on prescription. This study attempted to explore the automatic interpretation of text in electronic prescription systems and techniques of Case Based Reasoning for recommending drugs.

INTRODUCTION
Biomedical Informatics (BMI) is a multi-disciplinary area, which results from two main disciplines: Information Technology (IT) and the area of Medicine and Health Sciences. BMI is gaining importance worldwide. BMI can provide the modernization and improvement of the quality of delivering health services through better management of health information resources [1]. In the context of the need for management and organization of information, Hospital Information Systems (HIS) are arising. HIS can be defined as a subsystem hospital with a socio-technological development, which covers all information processing as well as the role of clinicians [2]. Its main purpose is to contribute to the quality and efficiency of healthcare. This objective is primarily oriented to the patient after being directed to health professionals as well as the functions of management and administration [3].

From the early history of mankind the concept of prescribing exists. Since the time when there are medicines, and a writing system comprehensive enough to describe guidelines on how to prepare and use, exists the prescribing of medicines. Currently, the prescription has an important role in the pharmacological treatment, after making a diagnosis, to eliminate or alleviate diseases or symptoms. Because there are several medicines names with a similar spelling and pronunciation, the error in the act of doctoring is considerable [4]. If we associate this fact with other types of error as an incorrect dosage, drug conflicts, bad or poor patient-physician interaction, we can conclude that it is imperative to try to reduce all these risk conditionings [5, 6, 7]. The application of artificial intelligence techniques in these systems can reduce costs, time and medical error [6, 8, 9].
AIMS
Currently there is a considerable amount of drugs, and with that comes the need to organize them into groups so as to make the search for a drug more quickly and easily. This kind of organization finds itself in a document provided by the Portuguese National Authority of Medicines and Health Products [10]. In this document there are several chapters for the various drug groups, and each chapter is usually associated with components of the human body, for example, respiratory system, digestive system. For each chapter is appended a set of descriptive information of what symptoms or diseases to guide the prescribing. This organization makes the prescription by the physician faster.

The main purpose of this work is to make proposals for assisting the medical decision making at the time of prescription. We intend to develop a system that implements an automated method for interpreting the text of the prescription and recommend the more efficient medicine, taking into account the text and past cases of prescriptions.

A prototype was implemented and tested with real data in the AIDA framework and deals directly with the AIDA-PCE system.

AIDA
AIDA (Agency for Integration, Diffusion and Archive of Medical Information) is a platform that consists of a Multi-Agent System (MAS) and overcomes difficulties in achieving uniformity of clinical systems, as well as medical and administrative complexity of different Hospital information sources [11]. AIDA was created by a group of researchers from the University of Minho, at the Artificial Intelligence Group, and is currently installed in various Hospitals.

AIDA is an agency that provides intelligent electronic workers, here understood as software agents, that present a pro-active behavior and are in charge of tasks such as the communications among its different sub-systems, sending and receiving information (e.g. medical or clinical reports, images, collections of data, prescriptions), managing and saving the information and answering to user requests, on time [12]. The main goal is to integrate, diffuse and archive large sets of information from heterogeneous sources (i.e. departments, services, units, computers, medical equipments); AIDA also provides tools in order to implement communication with human beings based on web-based services. Under these presuppositions, a healthcare information system (HIS) will be addressed in terms of the Administrative Information System (AIS), the Medical support Information System (MIS), the Nursing support Information System (NIS), the Electronic Health Record (HER) and the Department Information Systems (DIS), in particular of the Laboratories (Labs), Radiological Information System (RIS) and Medical Imaging, which deals with images in a standard format, the DICOM one.

Agents exchange messages, which are well-formed formulae of the communication language, performing acts or communicative actions [12, 13, 14]. The intelligence of the system as a whole arises from the interactions among all the systems components [15].

AIDA-PCE
The AIDA-PCE is an EHR and was implemented in the Centro Hospitalar do Porto. It is a subsystem of the HIS. The AIDA-PCE follows a problem-oriented organization suggested by Lawrence Weed in the 60’s. This information organization is known as the Problem Oriented Medical Record (POMR) and it assumes that registration is a production of clinical scientific document. In this type of organization, clinical information (annotations, therapeutic, diagnostic) should be recorded for specific problem solving, creating a list of issues organized in a tree structure, where each new problem derives from the main branch [16, 17]. One’s note that problems can be classified as active or inactive, in which active problems are those where the disease is still active or even when intervention is required immediately. On the other hand, inactive problems require no urgent action. In AIDA-PCE problems assets are monitored and recorded daily using a SOAP (Subjective, Objective, Assessment and Planning) framework. Thus, each record contains the patient's symptoms, a doctor's observation, an analysis of diagnosis and a treatment plan that the patient is subject to [16, 17]. An episode is the set of all operations for the patient, since the start of treatment until the end. Each episode is built upon an Integral Database
(BDI), the Problem List, the Therapeutic Plan and subsequent records. It is noted that this register can be updated during the episode and along the evolution of the patient.

NORMALIZATION

One important property of AIDA-PCE is its interaction with different systems and the communication with biomedical devices. The information to be transferred must be standardized and normalized in order avoid different structures and misinterpretations. We must also take into account the data semantics, so information can be understood by different systems. In addition, the use of standards in the AIDA-PCE ensures the best communication between health professionals and interoperability between systems, allowing some automation in the hospital recording. The standards used in EHR are divided into three different purposes: standards for representing clinical information; communication standards; and image standards [16]. International Classification of Diseases, Ninth Revision, Clinical Modification (ICD-9-CM), Systematized Nomenclature of Medicine-Clinical Terminology (SNOMED-CT) and International Classification for Nursing Practice (ICNP) are standards for classification of diseases and therapeutic clinics, where each therapy or disease is associated with a code recognized anywhere in the world. The use of these standards ensures that the EHR can be readable by any clinician in the world, allowing machines to interpret symptoms and assisting the clinicians in making a diagnosis and treatment plan decision [7]. As communication standards, the AIDA-PCE adopts the Health Level Seven (HL7) as a protocol for exchanging messages, and web architectures and service-oriented architectures (SOA).

PROTOTYPE

Technical overview

The system can be divided into two main functional components. The first component uses a word counting and its main goal is assisting the doctor in search of drugs, by describing the symptoms or diseases of the patient in question. Thus, when introducing a new word, the system processes this new knowledge to ensure that the drugs suggested are those that best relate to the set of words (symptoms or diseases) entered by the doctor. We can compare this component with a filtering system that replaces traditional forms. The second component uses Case Based Reasoning, and its main goal is to suggest to the doctor past prescriptions that best resemble the set of descriptions entered by the doctor, so that it could use some of the drugs prescribed in past situations. Once again, this component runs in the introducing of a new word by the doctor. The system can be divided into two sub-components represented in the figure 1.

Filtering

The way the document is organized is much like a tree structure where each node represents a chapter and may have child nodes corresponding to sub-chapters. In our system the information is represented on the form of a tree where each node has or not, a set of references to child nodes (sub-chapters), a reference to its parent node (if it has) and a counter that is used in the filtering process.

As already mentioned, when the doctor inserts a new word, the chapters that contain that word are found, and their counters are incremented. The incrementing occurs not only in the chapter of the occurrence. All the parent chapters increase. The main objective is to ensure that there are no low level chapters being harmed, or that are rarely suggested. This step is shown in figure 2. After the previous process, all the lists of chapters are ordered to ensure that drugs presented to the user belong to the chapters that best relate to the set of terms entered by the user. The expected outcome of these two previous steps can be illustrated in the figure 6. Once the chapters are ordered, the drugs from each one of them are selected; this selection takes into account drugs that have been previously presented to the user and those he has excluded.

Case Based Reasoning

The Case Based Reasoning (CBR) is a problem solving methodology that determines the solution to a given problem based on a reconstruction of past similar cases, reusing or adapting the knowledge of such cases. The CBR is suitable for areas where there is vast information and where this information is mostly past experiences. Therefore, this tool is suitable for the problem domain, since a doctor prescribes drugs using knowledge of diagnoses and
prescriptions made in the past. Particularly in our system, a case consists of a set of words that describe symptoms or diseases in which a solution was given; the solution in this case corresponds to a set of drugs that were prescribed (figure 7). The process of selection of similar cases is performed by calculating the similarity of the set of words in the current case and past cases, along with other constraints such as age and gender. The reuse or adaptation of knowledge of past cases in this area corresponds to adopting of one or more drugs from the solution of past cases. The CBR acquires a learning mechanism in order to acquire knowledge resulting of new cases. In our system, this step is verified upon completion of an electronic prescription; in this situation the current case is stored in the repository for future reference. A point to emphasize is that the cases only contain information of age and gender of the patient and not their name, in order to respect the privacy of the patient.

CONCLUSION
In this work we defined a CBR that can improve future applications of electronic prescription of drugs. The user, instead of filling a wide range of forms, describes the case and the system will provide a simple way to present a set of medicines and prescriptions based on past cases that are best suited to information from the typed text. This system allows physicians to introduce the traditional text of prescription, improving efficiency and reducing reluctance for the interfaces of such systems. The presentation of past prescriptions allows an important aid in decision-making and thus can reduce medical errors.

However, there are obvious improvements and optimizations to make. In the process of capturing keywords, it was essential that experts in pharmacology and physicians, which could enable a wider and better selection of words checked, accompanied this process. Moreover, the recommendation system implemented does not cover all of the factors that influence the decision to choose the best drug for the case. There are some parameters that are not considered.

FUTURE WORK
This project was not concluded. This prototype was implemented in Centro Hospitalar do Porto, integrated in AIDA framework and AIDA-PCE. The next step will go through the registration of clinicians monitoring. Thus possible changes will be made to the prototype to increase the advantages in its use for clinicians. In the end of this step, it will be possible to have a better evaluation of the system.

ACKNOWLEDGEMENT
The work of Julio Duarte is supported by the Portuguese Foundation for Science and Technology, with the grant SFRH/BD/70549/2010.

BIOGRAPHIES
Júlio Duarte, João Neves and Alexandra Cabral are currently pursuing doctoral studies in Medical Informatics, at the University of Minho.
Marco Gomes and Vitor Marques are MsC students at the University of Minho.
Manuel Filipe Santos holds a PhD in Computer Science – Artificial Intelligence. Actually he is associate professor in the department of information systems, school of engineering, University of Minho, Portugal.

António Abelha and José Machado hold a PhD in Computer Science – Artificial Intelligence. They are now Professors at the Department of Informatics of the University of Minho.

REFERENCES


Enabling Real-time Intelligent Decision Support in Intensive Care

Filipe Portela¹, Manuel Filipe Santos¹, Pedro Gago², Álvaro Silva³, Fernando Rua³, António Abelha⁴, José Machado⁴ and José Neves⁴

¹Centro Algoritmi, Universidade do Minho, Guimarães, Portugal
cfp.mfs@dsi.uminho.pt
²ESTG, Instituto Politécnico de Leiria, Leiria, Portugal
pedro.gago@ipleiria.pt
³Centro Hospitalar do Porto, Porto, Portugal
moreirasilva@clix.pt fernandoruau.sci@hgsa.min-saude.pt
⁴CCTC, Universidade do Minho, Braga, Portugal
{jmac, abelha, jneves}@di.uminho.pt

KEYWORDS
Real-Time, Intelligent Decision Support, Data Engineering, Data Mining, Intensive Medicine, Agents, KDD.

ABSTRACT

Medical devices in ICU allow for both continuous monitoring of patients and data collection. Nevertheless, the amount of data to be considered is such that it is difficult for doctors to extract all the useful knowledge. In order to help uncover some of that knowledge we have built an IDSS based on the agent's paradigm and using data mining techniques to build prediction models. With the intention of collecting as much data as possible the data acquisition process was automated. Furthermore, given the paramount importance of data quality for data mining a data quality agent responsible for detecting the errors in the data was devised. Indeed, data acquisition in the ICU is error prone as, for instance, sensors may be displaced as patients move. The aim of this paper is to present the overall KDD process implemented, presenting in detail the data transformations that were done and the benefits achieved.

INTRODUCTION

Intensive Medicine is a critical health area where lots of data are continuously collected. Most of it comes from electronic medical devices present in the Intensive Care Units (ICU) and are associated with some patient. However, this automated data collection is usually done without any type of data treatment or selection and because of that many of these values are null or considered out of range. Moreover, the volume of data is such that if makes it difficult for the medical staff to interpret this data. In addition, they can't extract useful knowledge and quickly understand which are the true values.

We can argue that this delay and the associated indecision about the values collected can keep doctors from reaching life saving conclusions that could have been reached if adequate decision support tools were present. With the objective to help the doctors in their decision making process and give to them the real values about the patient, is necessary develop an Intelligent Decision Support System (IDSS) that operates in a real-time and is able to present the data with a very fine granularity. The success of this system depends on its ability to fulfill all the tasks associated with the Knowledge Discovery in Databases (KDD) process.

The Extract, Transform and Load (ETL) phase is one of the most critical as many of the data collected needs to be validated and treated, with the objective to present data with a high level of quality and thus be able to get the best prediction models. To automate the data acquisition process and the ETL phase the development of some agents were necessary.

This paper is divided in five chapters, the first and second introduces the theme and all concepts related. The third chapter is the base of this paper and presents all phases of KDD that were completed on the ICU, some processing and transforming tasks performed, and some data mining result obtained. The last chapters present a little analysis of the results obtained with the completion of KDD, the conclusions of those and some future work to this area.

BACKGROUND

Intensive Medicine

Intensive Medicine (IM) is a “Multidisciplinary area of Medical Sciences that specifically addresses the prevention, diagnosis and treatment of acute potentially reversible in patients with failure of one or more vital functions” (Silva, 2007). This is a particular environment were anyone and anything can fail. The patients admitted to Intensive Care Units are in serious conditions. The critical care medicine allows the recovery of the patients in terminally ill or in a state of organ failure.

The quickly patient recovery depends largely, the patient data quality and decisions taken in the ICU. The main objectives are to diagnose, monitor and treat patients with serious illnesses and recover them to the quality of life they had prior to being admitted into the ICU (Suter et al., 1994). In the ICU physiological patient variables such as heart rate, blood pressure, temperature, ventilation and brain activity are constantly monitored on-line (Mahmoud, 2003) and others like administered drugs and fluid balance are registered Hourly. Normally for the doctors is very difficult to interpret this data quickly and in useful time.
Decision Making Process

The process of decision making is a key point in critical areas such as critical care medicine, or assessment of prognosis and diagnosis or treatment of the patient (Silva, 2007), since, according to some studies, the medical error can be the eighth leading cause of death in industrialized countries (Kohn, Corrigan, & Donaldson, 2000). Normally this process is complex, need to be done quickly and deals with a large number of variables that are in a constant modification.

Currently in the ICUs, doctors analyses the clinical data and relying on its experience they decide whether or not to intervene. The information usually comes from the monitors that are placed beside the bed, or notes that are taken periodically. This information is used to find out what action to take, which should be the course of this and the treatments to apply. However, we can verify that the information provided sometimes isn’t enough, isn't displayed in correct way or it arrives only after a decision has been made.

The decision making process is a continuous process that encompasses four phases: Intelligence, Decision, Choice and Implementation (Turban, Sharda, & Delen, 2010). After the implementation, the results obtained must be monitored and should be used to reformulate the models and these will be used with new sources on the consequently intelligence phase.

Touching the phases of the decision making process and analysing the area of critical care medicine there is none IDSS that follow these parameters. The INTCare system seeks to correct the gap in the ICU (Gago et al., 2006; Portela et al., 2010), in the provision of information and rapid response to a problem.

Real-Time

While never easy, the development of real-time systems is particularly demanding in critical areas where every second counts and nothing can fail. Such settings make it very difficult and even dangerous to develop this type of systems. Real-Time is a particularly feature of this systems and involves a lot of others characteristics like online learning, data security, flawless processing and autonomous process. In this cases any process that we try modify or create is a new and entire challenge for the human resources, technologies and the environment (Scicluna, Murray, Xiao, & Mackenzie, 2008).

The data acquisition process is one of the most critical phases because technical, human and environment factors are involved and may condition the quality of the data acquired and the success of each task (Portela, Santos, et al., 2010). However, in order to have an IDSS that can operate in real-time all tasks need to be developed in real-time mode, the data needs to be ready when they are necessary and we have to automate the highest number of tasks possible (Morik, Imhoff, Brockhausen, Joachims, & Gather, 2000).

Knowledge Discovery Process

Before you get the knowledge, the KDD process goes through five phases: Selection, Pre-Processing, Processing, Data Mining and Interpretation (Fayyad, Piatetsky-Shapiro, & Smyth, 1996). Error! Reference source not found. presents the KDD process and the results obtained at the end of each task. The process starts with the raw data obtained in the sources (databases) and culminates with the obtaining of new knowledge.

Figure 1: KDD Process

The Extract, transform and load (ETL) process involves the three initial phases and consists in the extraction of data from the sources, transformation of these data and loading the final data to the data warehouse, i.e. prepare the data to be used by data mining algorithms.

INTCare

INTCare is a research project which main goal is to develop an Intelligent Decision Support System for Intensive Medicine that is capable to predict the patient outcome and the organ failure (Gago, et al., 2006; Gago, Silva, & Santos, 2007) in real-time (Vilas-Boas, Santos, Portela, Silva, & Rua, 2010). The system is being developed and tested in the ICU of the Hospital Santo Antônio (HSA) in Porto, Portugal.

The INTCare system is divided into four subsystems: data acquisition, knowledge management, inference and interface (Portela, Santos, et al., 2010) and uses intelligent agents (Manuel Filipe. Santos, et al., 2011) who are responsible for a range of activities, such as the automation of data collection and updating of the predictive models in real-time, without the need for human intervention.

Multi-agent System

An agent is an intelligent program that acts autonomously on behalf of the user.” (Manuel Filipe. Santos, et al., 2011) The intelligent agents used by INTCare are capable to performs autonomous actions in order to meet its goals (Gago, et al., 2006), (Jennings, 2000). With the objective to support the KDD process a multi-agent system was integrated in INTCare. This agents are responsible for perform, automatically, all tasks of data processing and data transformation.

ICU KDD PROCESS

Data Engineering is a continuous and adaptive process. We started this process two years ago when after some requirements analyses for INTCare and consequent necessities of ICU we defined the reformulation of the information system architecture.
The base of the entire process was: the results obtained in the past (offline) (Silva, 2007; Silva, Cortez, Santos, Gomes, & Neves, 2008), the necessity of innovation, systems integration, dematerialization of processes, turn all information electronic and make it available online and in real-time eliminating the high number of errors present in the patients records.

The first challenge in the ICU, consisted of the change of record method and in the obtaining of some access privileges to some data sources. In 2009 80% of registers were made on paper, weren't available electronically (e.g. pdf format) or could not be accessed.

Nowadays, all data that are needed to create the models and obtain new knowledge, are computerized and are available online and in real-time, except to some blood gas results that is collected by ICU nurses and is not present on the normal lab results.

Table 1 presents the evolutions that took place in the ICU in the last 2 years, i.e., an analysis of the data that we now can monitor (consult, edit, validate) and document (save) electronically. Most of the changes were achieved with the development and introduction in the ICU, of the ENR (Portela, Vilas-Boas, Santos, & Fernado, 2010; M.F. Santos et al., 2009) and with the access to Electronic Health Process by AIDA (Abelha et al., 2003).

<table>
<thead>
<tr>
<th>Data Sources / Variables</th>
<th>2009</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nursing Record</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Vital Signs</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Drug System</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Patient EHR</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Fluid balance</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Procedures</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Lab Results</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Patient Events</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Ventilations</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Patient Scores</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>ICU Scores</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>ICU Blood</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

Figure 2 shows the ICU KDD process. The database is populated with data from seven major sources.

The data will be selected to the data warehouse to be processes or transformed, depending of the use goal to each variable. After this task the data will be available to be presented by the ENR and prepared for the creation of Data Mining Models. Finally all models will be evaluated and the knowledge obtained will be presented in INTCare.

### Table 1: Monitoring and Documentation of data sources

<table>
<thead>
<tr>
<th>Data Sources / Variables</th>
<th>2009</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nursing Record</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Vital Signs</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Drug System</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Patient EHR</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Fluid balance</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Procedures</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Lab Results</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Patient Events</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Ventilations</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Patient Scores</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>ICU Scores</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>ICU Blood</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

Figure 2: ICU Knowledge Discovery in Database Process

#### Selection

After the modifications we can obtain a lot of information regularly (up to once per minute) to be pre-processed and transformed.

The ICU Database is composed by various tables that store all data collected by the various systems. The database is compose by tables that contain the raw data collected by gateway and tables that store the data that were processed and transformed (real data).

The data are provided from a lot of systems that are integrated by a single application: the Electronic Nursing Record (ENR) that presents all data collected.

The ENR gives to the nurses and the doctors the possibility to consult, edit and validate a lot of values and results related with the patients. The Database integrates the following tables:

- **ICU_HL7**: {Vital Signs}
- **ICU_HL7_T**: {Vital Signs auto validated (real values)}
- **ICU_PARAM**: {ICU Limits (max, min) values}
- **ICU_LR**: {All Lab Results}
- **ICU_DRUGS**: {All Patient Drugs administrated}
- **ICU_ENR**: {Data validated and provided from ENR}
- **ICU_ADEVENTS**: {ICU Adverse Patient Events}
- **ICU_CEVEVENTS**: {ICU Critical Patient Events}
- **ICU_PATIENT**: {HER, Patient Information (admission, outcome, age)}
- **ICU_DATABASE**: {ICU_HL7, ICU_HL7_T, ICU_PARAM, ICU_DRUGS, ICU_LR, UCI_ENR, ICU_ADEVENTS, ICU_CEVEVENTS, UCI_PATIENT}

The data necessary to create the models and the Real-Time IDSS will be selected from these tables.

#### Pre-Processing

All data will be processed and transformed even if it is not necessary for the creation of the data mining models currently in use. This option makes possible, in the future, the introduction of new variables on the models (adaptive features).

All data collected that have null values and values out of the range will be treated. This operation will be two phases: the auto processing and the human validation. The auto processing is based in the achieving of automatic task, by intelligent agents (Manuel Filipe. Santos, et al., 2011), which will run after receipt some values.
The validation of values by the nurses is needed, because they are the only that are near the patient bed and can validate the veracity of the values. This validation will be required to specific variables that due the critical area that we are can have dubious values.

This phase will be autonomous and will be done by the pre-processing agent and is concerned with the problems:

a) No patient ID:
   b) Null Values; and
   c) Values out of range.

**No patient ID**

The process of collecting data from the gateway without patient identification (PID) is something normal in an ICU, because the PID is manually inserted by the nurses and isn’t usually considered important as patient treatment can proceed even if no PID is registered in the system.

Nonetheless, this is a very important field for automated systems and as such we needed to have it filled. Our first option was to motivate the nursing staff to insert the PID and we noticed some benefits to the system.

However the results weren’t as expected. We witnessed a decrease in the number of missing PID but this value continued very high. In 2011 we opted for another solution: auto-update hl7 message.

The next 3 charts (1, 2, 3) present the evolution (2009 - 2011) of the collect of data of patient: with all days, the PID failing one day or failing more than one day.

![Chart 1. PID in 2009](image1)

![Chart 2. PID in 2010](image2)

![Chart 3. PID in 2011](image3)

\[
\begin{align*}
\Omega & \text{ – Patient ID collected} \\
\beta & \text{ – Patient ID on EHR} \\
\gamma & \text{ – Bed ID} \\
\delta & \text{ – ICU ID} \\
\phi & \text{ – Date of Admission} \\
\psi & \text{ – message date} \\
\end{align*}
\]

if \( \Omega = \text{null} \) then
get \( \beta \) in \((\gamma, \delta, \psi)\)
set \( \Omega = \beta \)

**Null Values**

The null values occur when some sensor was disconnected from the patient and continues collecting the data. The solution found for this problem is simple. The function (2) will delete the row with null values:

\[
\begin{align*}
\Omega & \text{ – database row} \\
\beta & \text{ – variable_id} \\
\gamma & \text{ – result} \\
\end{align*}
\]

if \( \beta \) is null or \( \gamma \) is null
delete \( \Omega \)

**Values out of range**

This is a critical task and is based on a set of analyses that will determine if a value is possible or not in the ICU environment, i.e. if the value collected from a patient is or not true. If the value collected be within the range previously defined this will be validated e inserted on the table with the real values, otherwise this value will be inserted in a different table.

The equation (3) shows those operations:

\[
\begin{align*}
\Omega & \text{ – Database Row} \\
\beta & \text{ – variable_id} \\
\delta & \text{ – min\_value} \\
\gamma & \text{ – result} \\
\theta & \text{ – max\_value} \\
\end{align*}
\]

if \( \gamma >= \delta \) in (\( \beta \)) and \( \gamma <= \theta \) in (\( \beta \))
insert \( \Omega \) on table_in
else
insert \( \Omega \) on table_out

Now, like shows chart 4, ICU vital signs tables don't have any bad values stored because, if the gateway receive a value out of the range defined by icu, the vital signs agent ignore that hl7 message and didn't store this value in database.

![Chart 4. Example of comparison of values out of range](image4)
The manual validation of values will be done in hourly in the ENR for: vital signs, drugs system and fluid balance. The values that will appear in the ENR will be the data that were automatic validate and are present in real data table. With this option is possible, if necessary, correct the bad values or insert values out of defined range if that happen with any patient.

**Transformation**

Data collect as shown in Figure 2 will be divided in two parts, one that only needs to be processed and validated and other that must undergo some transformations. These transformations are necessary in order to calculate some variables necessary for the development of the DM models (Manuel Filipe Santos & Portela, 2011):

a) Critical Events;
b) Accumulated Critical Events;
c) SOFA;
d) SPAPS;
e) MEWS.

**Critical Events**
The calculation of critical events (4) is done with the base of the maximum values defined for ICU, the limits of a normal value and the time for an event be critical (Silva, et al., 2008). First is verified if the value is normal or critical, next, will calculated the time of the event and stored in the critical events table, an identification of the event.

\[
\begin{align*}
\Omega & \quad \text{Database Row} \\
\beta & \quad \text{variable_id} \\
\Omega & \quad \text{event_id (id, } \beta, \gamma, \text{ PID, C, } \Theta) \\
\beta & \quad \text{variable_id} \\
\gamma & \quad \text{result} \\
\delta & \quad \text{min_value} \\
\theta & \quad \max_value \\
\phi & \quad \max_value \\
\end{align*}
\]

if $\gamma \geq \delta$ in ($\beta$) and $\gamma \leq \theta$ in ($\beta$) and ($\gamma \leq \rho$ in ($\beta$) or $\gamma \leq \varphi$ in ($\beta$))

\[
\begin{align*}
\text{set } \mathbb{I} & = 1 \\
\text{else set } \mathbb{I} & = 0 \\
\mathbb{I} & = \text{sum (C - C)} \\
\text{If } \mathbb{I} \geq \Delta \text{ or sum (C)} & \geq \Theta \\
\text{insert } \Omega \text{ in } \mathbb{I} \\
\end{align*}
\]

**SCOR**

For all scores the method of calculation (5) is similar. The number allocated will be according the score punctuation table (Gardner-Thorpe, Love, Wrightson, Walsh, & Keeling, 2006; Le Gall, Lemeshow, & Saulnier, 1993; Vincent et al., 1998). Using the points associated the final score will be calculated and will the result will be inserted in the scores table.

\[
\begin{align*}
\Omega & \quad \text{final score} \\
\beta & \quad \text{variable_id} \\
\gamma & \quad \text{result} \\
\delta & \quad \text{score_result_punctuation} \\
\end{align*}
\]

\[
\begin{align*}
\Omega & = \text{sum( } \delta(\gamma, \beta) \\
\Omega & \Rightarrow 0 \\
\text{insert } \varphi \text{ in } \mathbb{I} \\
\end{align*}
\]

The process presented before are defined in database. To obtain all data some other functions, procedures or triggers were associated to the ICU database tables.

To perform these tasks, the multi-agent system for KDD process will be used, with the configurations defined the processes will start automatically trough some hourly schedule or are activated after some database operation (insert or update). The results obtained after each task will be stored in database according the necessary data and structures defined for the DM models.

**Data Mining**

Being of utmost importance to INTCare’s performance the knowledge management module provides the functionalities that allow the system to learn from the existing data. This subsystem is composed by three agents: Data Mining, Performance and Ensemble. Together they implement the Data Mining step of the KDD process.

The creation of predictive models is the responsibility of the Data Mining agent. Given a prediction objective, this agent retrieves the relevant data from the database and prepares it to be used by the chosen data mining algorithm. Next it creates a predictive model and stores it in the Knowledge Base.

Nevertheless, we can only have an autonomous system if it is capable of knowing what is going on and can evaluate itself in order to continue to operate with good performance levels. A Performance agent does exactly that by continuously sifting through the data that is being collected and updating a set of statistics that allow it to populate a Performance Database (Gago, et al., 2006; Portela, Santos, et al., 2010).

It analyzes the new data that was stored in data warehouse and verifies the performance of the prediction models through DM.

If the collected statistics show that the performance has fallen below a predefined parameter action must be taken in order to try to correct that situation (Gago, et al., 2006; Portela, Santos, et al., 2010).

One of the actions that can be triggered by the Performance agent is the replacement of a prediction model by another one built using the most recent available data. However, this procedure is frowned upon by doctors as it forces learning to happen in a sequence of “jumps” thus preventing a smooth learning curve for the system. Also, it is know that ensemble systems lead to better predictive results (as long as some conditions are met) (Dietterich, 2000). Hence, an Ensemble agent was created to enhance predictive performance by combining several models in order to produce models with better results.
Evaluation

Some work has been done in evaluating the effectiveness of INTCare. In one of the experiments we divided the available data into two mutually exclusive datasets. Models were created using the first dataset. Those models were then evaluated using the second dataset. That work showed that allowing for dynamic changes in the ensemble of predictive models gave way to better results (Gago & Santos, 2009) (Gago, et al., 2006; Portela, Santos, et al., 2010). This can be seen in Figure 3 where we show the increase in the area under the ROC curve (Zweig & Campbell, 1993) when we used this approach versus a traditional (static) ensemble.

DISCUSSION

The results obtained after the KDD process enforcement, were essentially related with the data quality. The new forms defined and implemented to collect, store and validate the data increase significantly the number of real values.

A major problem that normally was reported by the ICU staff was related with the lack of patient ID in the records collected automatically. Like we can verify in the chart 3 the success of our modifications and implementations, reaching a fantastic number of 0% of data collected without PID.

The same evolution was verified in the number of data collected with values out of the range defined by ICU. The biggest difference was verified in the Central Venous Pressure where the values down from 55% to 0%. Other variable that wasn't represented in the example (chart 4) but also was a common problem in ICU and now is resolved, was the body temperature. This variable sometimes had the sensor disconnected from the patient and collected the environment temperature, now this result is ignored by the agents.

The processing and data process was divided into some individual tasks, this tasks use the data collected in real-time and according the ranges of ICU, accept or not the values collected. On the other hand the data needed to for transformation and preparation is available to the automatic calculation of UCI Scores in order to integrate the data mining models, studied before.

The development of intelligent agents and the integration of a Multi agent system is a good choice when intended make some process autonomous. Improve the DM engine and obtain the better result as possible, having all information always updated, privileged models adaptive and in real-time and online processing.

The modification presented increases the IDSS performance giving guarantees of execution success in order to support a decision process in critical areas as is Intensive Medicine.

CONCLUSION AND FUTURE WORK

This paper presented a data engineering process for an IDSS in Intensive Medicine with base in the processing and transformation of data collected in real-time.

The results achieved are very important to improve the data mining models, acquire new knowledge and increase the sensibility and accuracy values inherent to the prevision models. Concluding, the INTCare data processing phase is now well defined and is prepared to run autonomously by the agents in charge to process and transform the data collected which makes possible the creation of real-time models. However, if some new variables or ICU configurations appear, this process can be adapted in order to obtain the best data quality possible.

In the future work, new models will be created using the data collected in real-time and treated using the KDD process presented in this paper. Also these models will be tested in a pervasive approach that are being developed and allow accessing by the ICU professionals to the Intelligent Decision Support System anywhere, anytime, with privileges access and security restrictions.

ACKNOWLEDGEMENTS

The authors would like to thank FCT (Foundation of Science and Technology, Portugal) for the financial support through the contract PTDC/EIA/72819/2006. The work of Filipe Portela was supported by the grant SFRH/BD/70156/2010 from FCT.

BIOGRAPHIES

Filipe Portela is a PhD Student in Information Systems in Minho University. He is inserted in the R&D Centre ALGORITMI and is developing his research work in the topic “Pervasive Intelligent Decision Support in the Critical Health Care”. This research was started with the INTTCare research project, has been doing in cooperation with the ICU of CHP and already has some important publication in Intelligent System, Pervasive Environment, Intensive Medicine, Data Mining Knowledge Discovery and Decision Support areas.

Manuel Filipe Santos holds a PhD in Computer Science – Artificial Intelligence. Actually he is associate professor in the department of information systems, school of engineering, University of Minho, Portugal. He is the Investigator Responsible in research projects (grid data mining and intelligent decision support systems for intensive
care). He is the head of the Business Intelligence Group in the Research Center Algoritmi. He has more than 60 publications in conference proceedings, chapter books and journals. His main interests are: Knowledge Discovery from Databases and Data Mining, Intelligent Decision Support Systems, Machine Learning.

Pedro Gago is assistant professor in Computer Science at the Instiuto Politecnico de Leiria. He is now finishing his PhD in Intelligent Decision Support Systems in the Doctoral Program in Technologies and Information Systems at the University of Minho. He is a researcher of the INTCare project.

Álvaro Silva, holds a PhD and is guest professor in Abel Salazar Biomedical Sciences Institute and works as a Doctor in the Intensive Care Unit of the Centro Hospital do Porto (CHP) in Portugal, where he has been doing research in the intensive medicine and data mining area. He belongs to the INTCare project team and is responsible by the connection between the Information System and Intensive Medicine area. In the last year, his knowledge and work contributed for publishing some important papers about Intelligent System in ICU.

Fernando Rua is a graduate doctor in Intensive Medicine. He is the Chief Officer of Intensive Care Unit of CHP in Portugal. He also develops research related with information systems and intensive medicine. He belongs to the INTCare project team and is responsible by the connection between the Information System and Intensive Medicine area. In the last year, his knowledge and work contributed for publishing some important papers about Intelligent System in ICU.

António Abelha and José Machado hold a PhD in Computer Science – Artificial Intelligence. They are now Professors at the Department of Informatics of the University of Minho. Their research area spans the field of Biomedical Informatics, Databases and Intelligent Systems, in particular when applied to Medicine. They have published several papers in book chapters, conference proceedings and journals.

José Neves is full professor at the Department of Informatics of the University of Minho. He got a PhD in Computer Science in 1984 in Edinburgh and has published more than hundred papers in book chapters, conference proceedings and journals.

REFERENCES


LATE PAPERS
ACTIVITYDIAGRAM2PETRINET : TRANSFORMATION-BASED MODEL IN ACCORDANCE WITH THE OMG SYSML SPECIFICATIONS

Damien Foures, Vincent Albert, Jean-Claude Pascal
CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse Cedex 4, France
University of Toulouse ; UPS, INSA, INP, ISAE ; UT1, UTM, LAAS ; F-31077 Toulouse Cedex 4, France
{dfoures, valbert, jcp}@laas.fr

KEYWORDS
MDE, ATL, TINA, transformation, verification, Petri Nets, SysML, OMG, VHDL-AMS, Ecore

ABSTRACT
This study aims to automate the transformation of activity diagrams (AD) to Petri nets (PN). Based on specifications given by the Object Management Group (OMG), we have established transformation rules in ATLAS Transformation Language (ATL) to obtain a model consistent with our Petri Net meta-model (model2model). The semantic of Activity Diagram was verified with PetriNet2Tina transformation (model2text) and has allowed us to verify that was the same in the corresponding PN. This verification is done with the "model-checker" Time petri Net Analyzer (TINA) and Linear Temporal Logic (LTL) language. The user needs only to set up the Activity Diagram from the stakeholder requirements; the transformation and verification is automatic. PetriNet formalism could enable us to provide valuable information on a Activity Diagram, to execute and simulate it.

INTRODUCTION

Context
The present work is based on the general context of systems engineering and integration of heterogeneous systems. These systems have software and hardware components, they generally hold high real-time constraints and disciplines (electrical, mechanical, information, hydraulic...). We seek to propose methods and tools for controlling the development cycle of such systems. The use of models and simulation is becoming dominant component in the development cycle, and we seek to improve (and eventually to automate) their use. The use of meta-modeling moves in this direction as it aims to make the modeling. Based on the instantiation of the meta-model, more clearer and less ambiguous.

Approach
Our approach uses the concepts advocated by the OMG through Model Driven Architecture (MDA), itself based on modeling and automatic transformation of models into other models. Presently we have the TINA formalism (LAAS 2011), TINA toolbox (LAAS 2011), and a transformation procedure from PN to VHDL-AMS (Albert et al. Octobre 2005). TINA formalism allows us to verify formally that Activity Diagram properties are preserved, using its model-checking tool (selt). The transformation to VHDL-AMS (IEEE 1999) allows us to propose a simulation phase, commonly called virtual prototyping. The addition of these two approaches allows us to validate the discrete and continuous part of the activity diagrams, and hence predict functional characteristics of the system.

In this work, we begin with a method to create meta-models of AD and PN. We will see a suggestion of concept mapping and its verification, and finally a simple example of transformation.

Till date transformation from State Machine to PetriNet were made (Bernardi et al. 2002), (Campos and Merseuger 2006), but Activity Diagrams (ADs) accentuate the internal control and data flow of systems. Further work on ADs were made (Bonhomme et al. 2008), (López-Grao et al. 2004) or (Thierry-Mieg and LHillah 2008) but do not take into account many properties of ADs.

SYSML PRESENTATION

SysML is a graphical modeling language developed by OMG and INCOSE. SysML is a UML profile adapted to systems engineering emerged in the 2000s. It can model the behavior of systems (continuous and discrete), with a hierarchical approach. OMG SysML specifications appeared only in 2007. We extracted the meta-model from (OMG 2010c) and (OMG 2010b) .
Activity Diagram

The activity diagram is one of the four behavioral diagrams included in SysML. They are useful to describe a hierarchical behavior, delayed, or a mixed systems. The TOPCASED framework permits to describe graphically all AD in accordance with the AD meta-model, itself in accordance with the Ecore(Budinsky et al. 2003) meta-meta-model. The Meta-Object Facility (MOF) is preconized by OMG, but Ecore is more or less aligned on Essential MOF (EMOF).

Activity Diagram Meta-model

The AD meta-model was extracted from OMG specification as described therein, without addition. Small part of it with ControlNode meta-class is shown in figure 1. We see that FlowFinalNode and ActivityFinalNode inherit FinalNode. The FinalNode inherits to ControlNode as InitialNode, similarly ForkNode, JoinNode and MergeNode. To create the AD meta-model, it is easier to begin with basic node of AD, directly afterwards all properties, all links with other meta-class were extracted. The biggest difficulty was to know where to stop the meta-model extraction. Indeed, in (OMG 2010c) and (OMG 2010b) all classes inherit from many other classes from SysML and UML metamodels. Subsequently it uses the AD meta-model from TOPCASED 1 framework. First, we have verified compliance with OMG specification for parts we have used.

![Activity Diagram](image1)

Figure 1: Part of Complete Activity Diagram in accordance with OMG specification.

PETRINET PRESENTATION

A PetriNet is a mathematical modeling language. There are currently a lot of Petri nets classes. Gradually basic, hierarchical and differential predicate transition Petri nets, will be transformed to, the control part, the hierarchy, and finally the continuous part of the activity diagram. A Petri net is composed of places, transitions, and arcs. Arcs connect a place to a transition or a transition to place, others possibilities are forbidden. This kind of constraint must appear in the meta-model of PN.

PetriNet Meta-model

The PN meta-model established in (Albert et al. October 2005) was adapted to this new work. Macro-place and macro-transition were removed because they are restrictive. For example, if an ActionNode of AD is transformed into macro-place it is impossible to put new value in this macroplace during execution. During execution macro-node becomes totally independant, so we decided to work flat. Flat PN, without hierarchy, are more easy to master communication links. TINA works also on only one abstraction level.

Figure 2 shows a simplified version of PN meta-model. We can read on it: PetriNet is composed of Node and ArcClassic. A node can be a Transition or a Place and they are linked with ArcClassic. A node can have multiple incoming or outgoing ArcClassics. An ArcClassic can only have one Source Node and one Target Node. This interpretation includes a description of the previous paragraph. However, constraints do not appear, they must be expressed, for example in Object Constraint Language (OCL)(OMG 2010a).

![PetriNet Model](image2)

Figure 2: Petri Net model conforms to simple Petri Net meta-model.

TRANSFORMATION WITH ATL AND ECLIPSE MODELLING FRAMEWORK (EMF)

Initially, our work was to be, totally in accordance with OMG. Tools for model transformation suggested by the OMG are still evolving, and to date we prefer to use EMF with Ecore meta-meta-model and ATL language which seems to be the best choice, with a framework that has been already tried and tested. Our transformation choices are pointed out in figure 3.

Mapping of Concepts

The original contribution of our transformation is to match an activity diagram artefact to a PN block which
will preserve the AD semantic, related to this artefact as defined by OMG. Such a PN block must also handle alternatives in AD modeling, e.g. an input pin may be stereotyped "optional" and becomes useless to start the activity. Table 4 illustrates the main mapping.

These design choices, reflects the analysis based on the generality of blocks (SendSignalAction or CallBehaviorAction inherited from Action), on block interconnection facility but also on properties defined by (OMG 2010c) and (OMG 2010b). For example, a ControlFlow can be modelled as a single transition (Thierry-Mieg and L'Hillah 2008), it can be also included on nodes like in (Bonhomme et al. 2008). In figure 4 the ultimate PN block acts as a buffer to respect ControlFlow properties written in OMG specification. They define ControlFlow like an edge that starts an activity node after the previous one is finished, with this simple definition an PetriNet arc is a correct model, but OMG add many specification on ControlFlow or which influences behavior. Finally, PetriNet arc is inadequate to meet all properties. Let's look at an excerpt of the properties and define possible solutions to respect them:

Property 1 (from ControlFlow): Tokens offered by the source node are all offered to the target node.

Property 2 (from ActionNode): When an action accepts the offer for control and object tokens, the tokens are removed from the original sources that offered them. If multiple control tokens are available on a single incoming control flow, they are all consumed.

Solutions: We can do with the property 1 that the first intuition is good, PetriNet arc carry tokens too. Property 2 and many others shows that the ControlFlow has behavior of token storage like a PetriNet place. The inability to know dynamically the number of token in a place to empty correctly ControlFlow brings us to the model as a buffer. Indeed, the presence of token is important but not the token multiplicity.

The same work was done with almost every ActivityDiagram node. Many stereotypes can be applied to nodes and was not considered to date.

The transition from one column to another in figure 4 is possible at M2 level (see figure 3 ) with ATL rules and Eclipse Modeling Framework. We'll see how we built a block PN during a small example.

**Complex Petri Net**

It was already seen how to build an atomic block. Building complex PN is relatively simple, in an activity diagram every or almost every node are connected to another by "ControlFlow" or "ObjectFlow". They will just have to connect each atomic block (can be viewed as:Transition-Place-Transition) to controlflow or objectflow block (can be viewed as:Arc-Place-Arc). We remind the reader that, analysis at model level should be higher than meta-model level to establish the rules in MDE context. Using the hierarchy can significantly reduce the amount of transformation rules. With AD2PN transformation, we could see that ATL cannot use easily the advantages of hierarchy. The language must be well controlled to limit significantly the coding rules.

**VERIFICATION**

After establishing the rules for "control flow" part, it is important to verify formally the transformation and, thus, verify that the PN had the same behavior as the activity diagram. In other words, it must check, through PN, to find the operational semantics of an AD. Subsequently, it is possible to imagine that users adds constraints (OCL) to the model, their validity in the PN can be proved with verification.

**ResolveTemp Meta-model**

Each PN block can be reduced to a sequence, Transition-Place-Transition. This meta-model defines each type of block to give essential features, but no behavior. It performs double transformation AD2PN and synchronized AD2ResolveTemp. This is to retrieve the name of input
transitions (isStarted), output transitions (isFinished), running place (isRunning) and this incoming/outgoing (incoming/outgoing) (see figure 5). Sometimes attributes are added to define better LTL property (isNotRunning, optionalIncoming,...)

**LTL Properties and self**

Owing to lack of space, we will not present LTL language. Our approach has been to develop, properties in blocks with properties with inputs and outputs. In accordance with transitivity relationship A ⇒ B and B ⇒ C then A ⇒ C. If block satisfies this properties, and if properties with connected blocks are satisfied then entire PN is verified. This verifies formally correct construction of the Petri net. This technique shows limitations indeed to have the expected Petri net (no problem in the construction). But it does not involve checking of correct behavior of the Petri net. If the building blocks have a limited or incorrect behavior, the model will be wrong and yet the verification will be positive. The user must know the limits of model transformation used.

**EXAMPLE: ACTIVITY DIAGRAM TO PETRI NET**

We can see in figure 6, at left, a simple AD. When Action1 is running, executes Action1, when this one is finished, starts parallel execution of both action (action 2 and 3). If action2 is finished, action 3 can run, but if action 3 is the first to finished, at this moment, Activity1 is stopped, and all action in Activity1 are stopped. We can see figure 6, at right, the resulting petri net after AD2PN transformation. It is more complex in appearance, it takes the behavior of Activity Diagram, that part is not really readable but does not provide specific information additional. To make this transformation, we must establish rules for each meta-class present through these instances in the model. For example: The ATL transformation rule for InitialNode meta-class.

```
rule initialnode_place{
  from a:MMAD!InitialNode
  to b:MMH!Place (Name<-'p_Initial_'+a.name+'.'+a.activity.name,
    OutputLink<-'c,
    ...
  ),
  c:MMH!ArcClassic( Name<-'a2_Initial_'+...
  ),
  d:MMH!Transition(
    OutputLink<-'a.outgoing,
    Name<-'...
  )
}
```

For each instance of InitialNode, a marked place is created, arc connects the latter to a transition. This transition is associated with meta-class instance after transformation, present in outgoing InitialNode argument (Outputlink—a.outgoing). This graphical version under TINA is possible with a second transformation, from Petri Net to TINA (model2text). To make this transformation we have used "Query" (LINA-INRIA 2006) from ATL:

```
helper context Hiles!Transition def:genTransition(): String = 'tr ' + self.Name + ' [0, w{'_
  + self.InputLink->iterate(arc;accPlam:Strings='|accPlam=arc.Source.Name +''+') + ' '=> ,
    + self.OutputLink->iterate(arc;accPlam:Strings='|accPlam=arc.Target.Name +''+') + '\n' ;
```

On this part of "Query", it is automatically generated the “arc part” of tina text. When the equivalent PN is implemented in TINA toolbox, we use "self" and its model-checking tool. To generate automatically properties in LTL language, we use an other transformation: from ResolveTemp to LTL (model2text) and other "Query" (LINA-INRIA 2006):

```
helper context ResolveTemp!RTC def:getPCF():String = '
''( '+self.isRunning+' '+self.isNotRunning='m1';\n' +...
```

The automated property created after this query is about ControlFlow and verifies invariant under block, the label ‘[]’ means that this invariant must always be true to validate this property.

```
[](p2_CFlow_CF1_Activity1+p3_CFlow_CF1_Activity1= 1);
```

The OMG specification says: "If multiple control tokens are available on a single incoming control flow, they are all consumed." (OMG 2010c). To respect this semantic, controlflow is modelised as seen in concepts of mapping subsection (presence or exclusively absence of tokken in ControlFlow).

**CONCLUSION AND FUTURE WORK**

This work has already set up a complex management of AD in accordance with the OMG specifications. OMG Specifications regularly contrain to refrain some short-cuts, which could simplify development. PN provides a mathematical formalism which is, at this step of the project not really exploited. It should eventually allow to highlight the invariant properties in AD, execute activity diagram in parallel with VHDL-AMS, to cover all the possibilities offered by the activity diagrams.
The future work will use differential predicate transition PN (Genrich 1987) for dataflow management. Today this first step can say if data are presents or not, but it does not convey any information. With this new class of petri nets, the TINA formalism becomes useless and we’ll go to validation by simulation (VHDL-AMS). During this work, it was also possible to show that management of interrupts areas is possible with basic PN.

We express concern about management of many changes which allows users ( stereotype, optional attribute,...), to manage all of these cases seems to overload rules of transformation. ATL language has sometimes seemed a bit complex, it will be interesting to see the contribution of Query / View / Transformation (QVT) language. The development of the reverse chain; Petri nets to activity diagram in this work will be completed by providing the user with an AD modified or re-organized according to the invariant or error detected in PN. Verification should be transparent to the user, however, a good mastery of AD seems crucial to create a really exploitable PN for us.

REFERENCES

Albert V.; Nikets A.; and Pascal J.C., Octobre 2005. Towards a model-based approach for hierarchical Petri net transformations to VHDL. European Simulation and Modelling Conference, Porto.


OMG, 2010a. OMG Object Constraint Language (OCL), Superstructure. OMG specification, , no. 2.3, 1–256.


An Objective Driven Framework for Discrete Event Simulation of Continuous Systems

V.Albert, A.Nketsa
CNRS, LAAS; 7 avenue du colonel Roche, F-31077 Toulouse, France
Université de Toulouse; UPS, INSA, INP, ISAE; LAAS; F-31077 Toulouse, France

email: valbert@laas.fr, nketsa@laas.fr

KEYWORDS
Experimental frame, Quantized integrators, DEVS, Second-order ODE

ABSTRACT
This paper introduces an objective-driven framework for discrete-event simulation of continuous systems. In industrial practice, a simulation is developed to verify and validate a given set of system requirements. As a matter of fact a test plan associated to those requirements is also elaborated. Nevertheless, test plans are not taken in consideration during the model design. We use the concept of experimental frame to design an operational formulation of the simulation intended purpose. Our models are based on the concept of quantization to describe traditional kinds of differential equation models into Discrete-Event Systems (DEVS). We provide the specification of a generic DEVS atomic model to represent a first-order Ordinary Differential Equation (ODE) usually called a quantized integrator. We illustrate our approach on the simulation of an adaptive cruise controller which uses a proportional-integral feedback control.

INTRODUCTION
A model abstraction is valid if it maintains the validity of the simulation results with respect to the questions the simulation is supposed to answer Frantz (1995). Hence, the validity of a model would never be assessed in isolation, but it should always be confronted to the context in which the model is experimented, i.e. the simulation intended purpose. If traditional M&S practices describe abstractions choices when building a model in order to provide a documentation of its "domain of use" Weld (1992), Frantz (1995), Albert and Nketsa (2009), those practices do not adopt the same process for the simulation intended purpose. Conducting simultaneously the same process for the model and its context would though make it possible to define a priori the sufficient and necessary model to reach an intended purpose and to check a posteriori if a model can be used in various contexts. In this context, challenging issues in M&S are mostly related to the specification/documentation of both, models capabilities and properties expected from these models to reach the intended purpose. There are many works which suggest processes and good practices for such documentation Balcı et al. (2000), Daum and Sargent (2001), Brade (2000), Oberkampf et al. (2004). Those proposals and the M&S process generally lack of formal methods. Some work has been done in integrating formal method in M&S Traoré (2006), Jacques and Wainer (2002). The idea is to construct a formal specification of models and to perform formal analysis on this formal specification. Furthermore, if a formal specification of the context is also provided, "capturing the dual relationship" between an experimental frame and a model become feasible by formal analysis too Albert et al. (2010), Traoré and Muzy (2006).

Ziegler’s Ziegler et al. (2000) M&S framework is a well accepted approach in the M&S community. In this framework there is the system which is the real or virtual element used as a source of observable data and subject to modelling. Through a modelling activity we obtain a model which is typically a set of instructions, controls, equations or constraints to generate its behaviour. Modelling is the process of abstraction of some aspects of the structure or behaviour of the real system regarding some questions. Those questions can be rigorously described with the concept of experimental frame. The experimental frame is a specification of the conditions in which a system is observed or experimented on. Simulation is a method which approximates the behaviour of the real system by executing the model over time. The simulator is a computer system used to execute the model and generate its behaviour based on model instructions and injected inputs. A simulator is correct if it faithfully generates the model output values given the model state and the input values. The correctness of a simulator refers to the principle of separating concerns between model design and its implementation. The validity of a model is defined as the degree to which a model faithfully represents a system within an experimental frame of interest. Ziegler also define the concept of applicability which determines wether an experimental frame can be applied to a model. This relation is fundamental as it serves to state wether a simulation intended purpose can be reached with a given model. Only few models can implement experimentation conditions required by an experimental frame to reach objectives and possibly supply valid simulation results.

Differential equations are traditionally used to model the continuous behaviour of dynamic systems. These modelling techniques describe the evolution, i.e. the rate of change of state variables via state equations which use derivative functions. At each time we only know this rate of change ac-
cording to the current state and possibly the input current value. A model can merely be classified according to two variables: its state space and its time space. A time space or a state space which can take values among a finite set is said to be discrete. A time space or a state space which can take values among an infinite set is said to be continuous. A computer simulation of a system described with differential equations approximates some elements of the system by discrete quantities. If the time space is discretized, the solution of the continuous system is computed at regular time interval, the state at time \( t + 1 \) is computed according to the state and the input at time \( t \). If the state is discretized, the system is approximated by a discrete-event system where an event, which corresponds to a state change on its discrete space, can occur at any time. This latter approximation is called quantization. These two approximation techniques require careful study of approximation criteria, e.g. error bounds, stability regimes, conservation properties.

This paper highlights the dual relationships between an experimental frame and a model through the simulation of a discrete-event representation of a continuous system using the concept of quantization. Section 2 introduces the concepts handled by the study. We successively promote a brief description of a general formalization of experimental frames, \textit{DEVS} and quantized integrators. In section 3 we contribute to a specification and an implementation of a generic \textit{DEVS} atomic model to represent a first-order Ordinary Differential Equation (ODE). Section 4 applies an objective-driven approach to an adaptive cruise controller which uses a second-order ODE to achieve a given speed.

BACKGROUND

Experimental frame

The experimental frame incorporates data gathering (statistics, performance measurements, etc.) and behavioral control (initialization, termination, etc.). An experimental frame can be seen as a system that interacts with the system or the model to obtain the data of interest in given conditions. Hence, defining the boudaries between the experimental frame and the model is fundamental in order to clearly distinguish among what drives the model, what is observed as its output and the model itself Albert and Nketsa (2009).

An experimental frame has three components as illustrated figure 1: a generator which generates a set of input segments for the system; an acceptor which selects the data of interest of the system while monitoring whether the desired experimental conditions are complied with and a transducer which observes and analyses the output segments of the system.

An experimental frame is given in Traoré and Muzy (2006) as a structure

\[
EF = \langle T, I_M, I_E, O_M, O_E, \Omega_M, \Omega_E, \omega_C, SU \rangle
\]

- \( T \) is a time base,
- \( I_M \) is the set of Frame-to-Model input variables, the set of model stimulation ports,
- \( I_E \) is the set of Frame input variables, the control input set,
- \( O_M \) is the set of Model-to-Frame output variables, the set of model observation ports,
- \( O_E \) is the set of Frame output variables, the summary set.
- \( \Omega_M \subseteq (I_M, T) \) is the set of segments injected onto the model inputs,
- \( \Omega_E \subseteq (I_E, T) \) is the set of admissible input segments for the experimentation control,
- \( \Omega_C \subseteq (O_M, T) \) the set of segments observed onto the model outputs
- \( SU \) is a set of conditions, also called summary mappings, which establish relationships between inputs and outputs within the frame.

A summary mapping can be seen as a set of pre- and post-conditions mapping the stimulation variables to the observation variables and so on monitoring the experimentation. For example let us consider the Boolean observation point \textit{alarm} in the experimental frame: "If the value of \textit{alarm} is true in one state of the execution, then there is a previous state in the execution where altitude \( \leq 10^9 \) is a summary mapping.

![Figure 1: The experimental frame and its components](image)

DEVS

A \textit{DEVS} atomic model is a structure Zeigler et al. (2000)

\[
M = \langle X, Y, S, \delta_{in}, \delta_{out}, \lambda, t_a \rangle
\]

where

- \( X = \{ p, v \} | p \in IPorts, v \in X_p \), is a set of inputs ports and their values
- \( Y = \{ p, v \} | p \in OPorts, v \in Y_p \), is a set of outputs ports and their values
- \( S \) is a set of sequential states
\( \delta_{\text{ext}} : Q \times X \rightarrow S \), is the external transition function

- \( Q = \{(s,e)| s \in S, 0 \leq e \leq ta(s) \} \) is the set of total states
- \( e \) is the time elapsed from the last transition

\( \delta_{\text{int}} : S \rightarrow S \), is the internal transition function

\( \delta_{\text{con}} : Q \times X \rightarrow S \), is the confluent transition function

\( \lambda : S \rightarrow Y \), is the output function

\( ta : S \rightarrow \mathbb{R}_{+} \), is the time advance function.

At any time the system is in state \( s \in S \), if no external event occurs then the system will stay in state \( s \) for time \( ta(s) \). If \( ta(s) = 0 \) we say that \( s \) is a transient state, the time staying in \( s \) is so short that no event can occur. If \( ta(s) = \infty \) the system is in a passive state, it stays in \( s \) while no event has occurred. When \( e = ta(s) \) the system produces an output value \( y = \lambda(s) \) and changes its state to \( s' = \delta_{\text{int}}(s) \). If an event occurs before \( ta(s), (s,e)| e \leq ta(s), \) then the system changes its state to \( s' = \delta_{\text{ext}}(s,e,x) \). The confluent transition function decides the next state in cases of collision between external and internal events.

Coupled models describe collections of interacting components, where components can be atomic or coupled models. A DEVS coupled model is a structure

\[ N = (X,Y,D,\{M_d|d \in D\},EIC,EOC,IC) \]

- \( X \) is a set of inputs
- \( Y \) is a set of outputs
- \( D \) is the name of the components
- \( EIC \) is the external input coupling function which links the inputs of \( N \) to the inputs of component \( d \in D \)
- \( EOC \) is the external output coupling function which links the outputs of component \( d \in D \) to the outputs of \( N \)
- \( IC \) is the internal coupling function which links the outputs of component \( a \in D \) to the inputs of component \( b \in D \)

Quantized integrators

The basic idea of the theory of Quantized DEVS (QDEVS) consists in discretizing the space of state variables using a fixed value called the quantum size \( D \). According to this quantum, a variable \( q \) can only take values among \( q \pm kD \) where \( k \) is an integer. The solution \( q(t) \) of a system described by a differential equation is approximated on a grid in the phase space of the system. The resolution of the phase space grid is \( D \). The time \( h \) required to move from one phase space grid point to another to occur on \( q(t) \) is approximated and a

![Figure 2: Quantization of a function \( q(t) \)](image)

state change will be informed only at this time. The figure 2 illustrates these techniques.

In fact \( h \) may be computed from classical ODE solvers Nutaro (2005), for example Euler or Runge-Kutta. Consider an ordinary differential equation in the form of

\[ \dot{q} = f(q(t)) \]

We consider the simple Euler integration method

\[ q(t+h) = q(t) + h\dot{q}(t) \]

Let the quantum \( D \) be defined by

\[ D = |q(t+h) - q(t)| \]

Then the time required for a change of size \( D \) to occur on \( q(t) \) is approximatively

\[ h = \begin{cases} 
\frac{D}{|\dot{q}(t)|} & \text{if } \dot{q}(t) \neq 0 \\
\infty & \text{otherwise}
\end{cases} \]

This approach is easily extended to a set of coupled ordinary differential equations in the form of \( \dot{q} = f(q(t)) \), where \( q \) is a vector of differential variables. For each ordinary differential equation \( i \) of such a system, two variables are necessary. A variable \( q_i \) which is the position of state variable \( i \) on its phase space axis and a variable \( q_l \) which contains the last grid point occupied by the variable \( q_i \). These two variables are necessary because the function \( f_i \) is now computed at grid points in the discrete phase space of each element of the vector \( q \). A variable \( q_i \) may have reach a grid point in its discrete phase space while the variable \( q_l \) has not reach its next grid point yet. Then the time required for the variable \( q_l \) to be updated becomes

\[ h = \begin{cases} 
\frac{D_{|q_i - q_l|}}{|\dot{q}(t)|} & \text{if } \dot{q}(t) \neq 0 \\
\infty & \text{otherwise}
\end{cases} \]

where \( |q_i - q_l| \) is the distance already traveled along the phase space axis of state variable \( i \).

DEVS REPRESENTATION OF QUANTIZED INTEGRATORS

Consider an equation in the form of \( \dot{q} = f(q(t),u(t)) \), where \( q \in \mathbb{R}^n \) is a vector of differential variables which represents the state variables vector and \( u \in \mathbb{R}^m \) is a vector of algebraic variables which represents the inputs vector, then a continuous model is formed by a set of first-order differential equations:
\[ \dot{q}_1 = f_1(q_1(t), q_2(t), \ldots, q_n(t), u_1(t), u_2(t), \ldots, u_m(t)) \]
\[ \dot{q}_2 = f_2(q_1(t), q_2(t), \ldots, q_n(t), u_1(t), u_2(t), \ldots, u_m(t)) \]
\[ \vdots \]
\[ \dot{q}_n = f_n(q_1(t), q_2(t), \ldots, q_n(t), u_1(t), u_2(t), \ldots, u_m(t)) \]

We call \( U \) and \( Q \) the set of inputs and the set of state variables of the system, respectively. Let \( U_i \cup Q_i \) with \( U_i \subseteq U \) and \( Q_i \subseteq Q \) be the domain of \( f_i \) with \( 1 < i < n \).

We use an atomic model \( M = (X, Y, S, \delta_{\text{int}}, \delta_{\text{est}}, \delta_{\text{con}}, \lambda, ta) \) to represent the system. The set \( \mathcal{X} \) of inputs is equal to \( U_i \cup Q_i \). The set \( Y \) has one output \( q_i \). The model’s inputs and outputs are real numbers, i.e., \( X = \mathbb{R} \) with \( l \) the cardinality of \( U_i \cup Q_i \) and \( Y = \mathbb{R} \). Note that if the equation is in the form of \( \dot{q}_i = f(q_i(t)) \), the set \( X \) of inputs is empty.

This atomic model has four state variables:

- \( q_l \), the last output value of the integral,
- \( q \), the current value of the integral,
- \( \dot{q} \), the last known value of the derivative,
- \( \sigma \), the time until the next output event.

The state variables are real numbers, i.e., \( S = \mathbb{R} \times \mathbb{R} \times \mathbb{R} \). The internal transition function, the confluent transition function, the external transition function, the output function, and the time advance function are given by the following equations:

\[ \delta_{\text{int}}(q_l, q, \dot{q}, \sigma) = (q_n, q_n, f(q_n, x), \frac{d}{df(q_n, x)}) \]
\[ \delta_{\text{con}}(q_l, q, \dot{q}, \sigma) = (q_n, q_n, f(q_n, x), \frac{d}{df(q_n, x)}) \]
\[ \delta_{\text{ext}}((q_l, q, \dot{q}, \sigma), e, x) =
\[ (q_l, q + \dot{q} \ast e, f(q, x), \frac{d}{df(q, x)}) \]
\[ \lambda(q_l, q, \dot{q}, \sigma) = q_n \]
\[ ta(q_l, q, \dot{q}, \sigma) = \sigma \]

with \( q_n = q_l + d \ast \text{sgn}(\dot{q}) \) the next value of the integral.

If an external input event occurs before \( \sigma = ta \), the value of the integral at this time is computed, the new value of the derivative is computed according to the previous value of the integral and the input values and finally \( \sigma \) is set to the time required to reach the next grid point. In that case, \( q \neq q_l \).

The confluent transition function is the same as the internal transition function.

**Example** A single ordinary differential equation.

We have built a DEVSJAVA Zeigler (2001) implementation of the DEVS specification given above for the first-order ODE \( \dot{q} = 0.2 \ast (2.0 - q) \) (appendix A.1). We have used the simulation environment DEVS-SUITE Sarjoughian (2004) to track simulation results of interest. Figure 3 shows the DEVS-SUITE simulation tracking log of the atomic model and the Runge-Kutta numerical solution.

![Figure 3: DEVS Simulation vs Runge-Kutta of \( \dot{q} = 0.2 \ast (2.0 - q) \) with \( q(0) = 0 \) and \( d = 0.15 \)](image)

The algorithm to simulate a single ordinary differential equation is given in Nutaro (2005). We have implemented this procedure in C, the table 1 below gives accounts of the computed values of variables \( \dot{q}, q, q_l \) at each time iteration for \( \dot{q}(0) = 0 \) and \( d = 0.15 \). We can see that the DEVS simulation gives the same results as the table 1 below. After \( t = 13.8 \) the solution will alternate between 1.95 and 2.1.

**APPLICATION**

The adaptive cruise controller Palmquist (1993), Schulz et al. (1997) adapts the speed vehicle according to the distance and the speed of the vehicle ahead. The safe speed is the maximum speed that keeps the car within a safe distance of the vehicle ahead. The desired speed is the speed the driver requests. The simulation intended purpose is to validate that the control unit keeps the speed of the vehicle within \( \pm 2 \text{km.h}^{-1} \) of the safe speed or the desired speed, whichever is lower.
Table 1: Computed solution of \( \dot{q} = 0.2 \cdot (2.0 - q) \) with \( q(0) = 0 \) and \( a = 0.15 \)

<table>
<thead>
<tr>
<th>( t )</th>
<th>( \dot{q} )</th>
<th>( \sigma )</th>
<th>( q )</th>
<th>( q' )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.4</td>
<td>0.275</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.375</td>
<td>0.337</td>
<td>0.405</td>
<td>0.15</td>
<td>0.15</td>
</tr>
<tr>
<td>0.780</td>
<td>0.34</td>
<td>0.444</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>1.221</td>
<td>0.31</td>
<td>0.484</td>
<td>0.45</td>
<td>0.45</td>
</tr>
<tr>
<td>1.705</td>
<td>0.28</td>
<td>0.536</td>
<td>0.6</td>
<td>0.6</td>
</tr>
<tr>
<td>2.241</td>
<td>0.25</td>
<td>0.600</td>
<td>0.75</td>
<td>0.75</td>
</tr>
<tr>
<td>2.841</td>
<td>0.22</td>
<td>0.682</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>3.523</td>
<td>0.19</td>
<td>0.789</td>
<td>1.05</td>
<td>1.05</td>
</tr>
<tr>
<td>4.312</td>
<td>0.16</td>
<td>0.938</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>5.250</td>
<td>0.13</td>
<td>1.154</td>
<td>1.35</td>
<td>1.35</td>
</tr>
<tr>
<td>6.404</td>
<td>0.11</td>
<td>1.500</td>
<td>1.5</td>
<td>1.5</td>
</tr>
<tr>
<td>7.904</td>
<td>0.07</td>
<td>2.143</td>
<td>1.65</td>
<td>1.65</td>
</tr>
<tr>
<td>10.047</td>
<td>0.04</td>
<td>3.750</td>
<td>1.8</td>
<td>1.8</td>
</tr>
<tr>
<td>13.797</td>
<td>0.01</td>
<td>15.000</td>
<td>1.95</td>
<td>1.95</td>
</tr>
<tr>
<td>28.797</td>
<td>-0.02</td>
<td>7.500</td>
<td>2.1</td>
<td>2.1</td>
</tr>
<tr>
<td>36.297</td>
<td>0.01</td>
<td>15.000</td>
<td>1.95</td>
<td>1.95</td>
</tr>
<tr>
<td>51.297</td>
<td>-0.02</td>
<td>7.500</td>
<td>2.1</td>
<td>2.1</td>
</tr>
<tr>
<td>58.797</td>
<td>0.01</td>
<td>15.000</td>
<td>1.95</td>
<td>1.95</td>
</tr>
</tbody>
</table>

Adaptive cruise controller simulation

Remember that defining the boundaries between the model and the experimental frame is an important step. Selecting the scope of the model implicitly includes selecting the exogenous parameters. All other components of the global system (e.g., sensors, actuators) now belong to the experimental frame to represent the environment of the cruise controller. Some of these elements can be abstracted or simplified if their contributions are not relevant to the experimentation. Figure 4 sketches the adaptive cruise controller simulation structural definition.

The cruise controller calculation module has three components. \( I1 \) and \( I2 \) are two quantized integrators, they compute the speed of the vehicle to maintain according to the safe speed \( (\text{safespd}) \) or the desired speed \( (\text{wantspd}) \), whichever is lower. According to \( \text{safespd} \) and \( \text{wantspd} \) values, the update module continually check which speed \( (\text{currspd}) \) must be used for cruise control and transmit it to \( I2 \).

In that context, the experimental frame consists of scenarios of different values for \( \text{wantspd} \) and \( \text{safespd} \). This is done by the generator. The transducer observes the output \( q1 \) which is the speed of the vehicle. It also observes the output \( q1 \) in order to record the speed only at stable state, e.g., when \( -2 \leq q1 \leq 2 \). The acceptor establishes relationships between inputs and outputs within the frame: it takes as input \( \text{currspd} \) and the speed at stable state \( (\text{recSpd}) \) and continually check if \( \text{recSpd} \leq \text{currSpd} - 2 \) and \( \text{recSpd} \leq \text{currSpd} + 2 \).

Adaptive cruise controller model

The adaptive cruise controller uses a proportional-integral feedback control to maintain a given speed. The equation of the feedback control block is

\[
\dot{q}_1(t) = k_P(v^* - q_1(t)) + k_I \int_{t_1}^{t_2} (v^* - q_1(t)) \, dt \quad (9)
\]

where \( \dot{q}_1(t) \) is the controlled acceleration to achieve the required speed, \( k_P \) and \( k_I \) are tuning parameters for proportional gain and integral gain respectively, \( v^* \) is the desired speed and \( q_1(t) \) is the speed at time \( t \).

Equation 9 can be written as a second-order ODE

\[
\ddot{q}_1 + k_P \dot{q}_1 + k_I q_1 = k_I v^*
\]

which is converted into two first-order ODES

\[
\dot{q}_1 = q_2 \quad (11)
\]

\[
\dot{q}_2 = k_I v^* - k_P q_2 - k_I q_1
\]

We have a system of two first-order ODES in the form of

\[
\dot{q}_1 = f_1(q_2(t))
\]

\[
\dot{q}_2 = f_2(q_1(t), q_2(t), v^*(t))
\]

According to our specification a DEVS representation of this system is a coupled component with two atomic model \( I1 \) and \( I2 \). With \( X_I = \{ q_2 \} \), \( X_R = \{ v^*, q_1 \} \), \( Y_I = \{ q_2 \} \). External transition functions for those components are given in Appendix A.2. To be compliant with the experimental frame expectations, the output \( \dot{q}_1 \) is added to the model, i.e. \( Y_I = \{ q_1, \dot{q}_1 \} \).

The update module is a DEVS atomic model given by

\[
X_U = \{ \text{safespd}, \text{wantspd} \}, \quad Y_U = \{ \text{currspd} \}, \quad S_U = \{ \text{passive}, \text{select} \} \times \text{safespd} \times \text{wantspd} \times \text{currspd}.
\]

External transition functions are used to update the values of \( \text{wantSpd} \) and \( \text{safespd} \). An internal transition function is triggered when the component is in passive and when \( \text{safeSpd} \geq \text{currSpd} \) &\& \( \text{safeSpd} \geq \text{wantSpd} \) to assign \( \text{currSpd} \) to \( \text{wantSpd} \). An second internal transition function is triggered when the component is in passive and when \( \text{safeSpd} > \text{currSpd} \) &\& \( \text{safeSpd} < \text{wantSpd} \) to assign \( \text{currSpd} \) to \( \text{safeSpd} \).

Adaptive cruise controller experimental frame

We give hereafter, the specification of the experimental frame

- \( T = \mathbb{R} \),

Figure 4: Adaptive cruise controller simulation

![Figure 4: Adaptive cruise controller simulation](image-url)
• \( I_M = \{ \text{wantSpd, safeSpd} \} \),
• \( O_M = \{ \dot{q}_1, \text{speed, currSpd} \} \),
• \( I_E = \{ \text{start/stop} \} \),
• \( O_E = \{ \text{speed, ok/nok} \} \).

\[ \Omega_E = \{ (s, t_{init}), (stop, t_{end}) \} \] where \( t_{init} \) is the begin
of the simulation and \( t_{end} \) is the end of the simulation,

\[ \Omega_M = \{ (\text{wantSpd}, t), (\text{safeSpd}, t) \} \] where \( t \in (t_{init}, t_{end}) \)
is the current simulation time,

\[ \Omega_C = \{ (\dot{q}_1, t), (\text{speed}, t - \varepsilon \leq \dot{q}_1 \leq s) \} \] where \( t - \varepsilon \leq \dot{q}_1 \leq s \in [t_{init}, t_{end}] \) is the simulation time for which the speed
must be recorded, \( \varepsilon \) is the threshold for which the exper-
iment assumes that speed has reached is stable state and \( t \in (t_{init}, t_{end}) \)
is the current simulation time,

\[ SU = (\text{recSpd} \geq \text{currSpd} - 2), (\text{recSpd} \leq \text{currSpd} + 2), \]

\[ D = (\text{Generator, Transducer, Acceptor}). \]

Coupling functions can be retrieve from the diagram figure 4.

Results

We have implemented a simple scenario which assigns the
speed to maintain to 80. At the initial state the speed of the
vehicle is 0.

The figure 5 below shows the DEVS-SUITE simulation track-
ing log of the speed of the vehicle (output \( q_1 \)). We have plot
the Runge-Kutta numerical solution on the same figure.

![Figure 5: DEVS Simulation vs Runge-Kutta of \( \dot{q}_1 + k_p q_1 + k_I q_1 =
\)](image_url)

\[ k_I v^2 \] with \( k_I = 0.5, k_p = 0.8, v_s = 80, q_1(0) = 0 \) and \( d = 2 \)

We have implemented the procedure given by Nutaro (2005)
which extends the algorithm introduced previously to a set
of coupled ordinary differential equations. The table 2 gives
the computed values of this system for \( k_I = 0.5, k_p = 0.8, \]
\( v_s = 80, q_1(0) = q_2(0) = 0 \) and \( d = 2 \). The first part of the ta-
bles give the step by step account of the first 9 iterations. The
second part gives the iterations around the maximum values
of \( q_1(t) \) corresponding to the overshoot caused by the inte-
rval term. The last part gives the iterations for \( q_1(t) \) reaching
its setpoint value. Each row in the table shows the computed
value of the variables of each ODE at the end of the iteration.
Recall that the resolution of the phase space grid is \( D = 2 \). At
each iteration, the time \( \sigma \) required to move from one phase
space grid point to another to occur on \( q \) of each ODE is ap-
proximated. The time \( t \) is advanced to the next change to
occur, i.e. the slower \( \sigma \) of both ODE. A blank cell means
that no changes occurred on the variable. This is the case when an
external event has occurred on the corresponding component.
As we can see the DEVS simulation gives the same results
as the table 2. Indeed, we obtain \( q_1 = 90 \) at time \( t = 4.7 \).
At time \( t = 8 \), the solution will alternate around the setpoint
value \( q_1 = 80 \).

Furthermore this table shows phases computed in the course
of the simulation for the transducer and the acceptor. The
transducer record the speed of the vehicle only when the
derivative is lower than 2 or greater than -2. Otherwise it
waits in its passive phase. The acceptor reaches an error
phase when the control unit does not keep the speed of the
vehicle within \( \pm 2km.h^{-1} \) of the current speed. As we can
see in the table, the acceptor switches in error phase when
the speed reaches its maximum value caused by the over-
shoot. This is normal, since the derivative is close to zero,
but obviously, the stable state is not reach yet. While
the model is valid, the experimental frame must be modified
to reduce this uncertainty. The simulation results exhibit errors
which are introduced by the experimental frame.

CONCLUSION

We have shown in this paper an objective driven framework
using a formal concept of experimental frame. We have il-
ustrated that the experimental frame and the model are two
components which must be confronted two assess the valid-
ity of simulation. Typically we have seen that an experimen-
tal frame could bias simulation results. Furthermore, check-
ing the applicability of a model to an experimental frame is a
fundamental step before processing a simulation. The objec-
tives are clear: the abstractions made when the model is de-
sign must match with a set of acceptance conditions given
by the experimental frame.

Futhermore we have provided the essential aspects of
discrete-event approximation for simulating a continuous
system. We have given a DEVS specification and a DEVS-
JAVA implementation of a generic atomic model to describe
a first-order differential equation. We have seen that this
atomic model can be easily coupled to others atomic mod-
els to describe a set of first-order differential equation. The
model implementation has been validated against numerical
solvers and discrete-event procedures given by Nutaro
(2005). Study of convergence properties and stability has not
been studied here. It should be interesting to assess the num-er of iterations to reach the stable state of a solution. In our
example we have used the derivative to know when the solu-
Table 2: Computed solution of \( \ddot{q}_1 + k_p q_1 + k_f q_1 = k_f r^* \) with \( k_i = 0.5 \), \( k_p = 0.8 \), \( v_r = 80 \), \( q_1(0) = 0 \), \( d = 2 \).

<table>
<thead>
<tr>
<th>t</th>
<th>q_1</th>
<th>dq_1</th>
<th>\sigma_i</th>
<th>\dot{q}_1</th>
<th>\dot{q}_2</th>
<th>\dot{q}_3</th>
<th>\sigma_1</th>
<th>\dot{phi}_{phase}</th>
<th>\phi_{phase}</th>
<th>phase</th>
<th>phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>34.8</td>
<td>2</td>
<td>0.052</td>
<td>record</td>
<td>passive</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td>0.05</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>34.8</td>
<td>2</td>
<td>0.052</td>
<td>record</td>
<td>passive</td>
<td>0.05</td>
<td>0</td>
</tr>
<tr>
<td>0.102</td>
<td>0.1</td>
<td>4</td>
<td>0.477</td>
<td>4</td>
<td>36.8</td>
<td>4</td>
<td>0.054</td>
<td>passive</td>
<td>passive</td>
<td>0.102</td>
<td>0.1</td>
</tr>
<tr>
<td>0.156</td>
<td>0.32</td>
<td>6</td>
<td>0.792</td>
<td>6</td>
<td>35.2</td>
<td>6</td>
<td>0.056</td>
<td>passive</td>
<td>passive</td>
<td>0.156</td>
<td>0.32</td>
</tr>
<tr>
<td>0.213</td>
<td>0.66</td>
<td>8</td>
<td>0.167</td>
<td>8</td>
<td>33.6</td>
<td>8</td>
<td>0.060</td>
<td>passive</td>
<td>passive</td>
<td>0.213</td>
<td>0.66</td>
</tr>
<tr>
<td>0.273</td>
<td>1.13</td>
<td>10</td>
<td>0.086</td>
<td>10</td>
<td>32.1</td>
<td>10</td>
<td>0.092</td>
<td>passive</td>
<td>passive</td>
<td>0.273</td>
<td>1.13</td>
</tr>
<tr>
<td>0.335</td>
<td>1.76</td>
<td>12</td>
<td>0.052</td>
<td>12</td>
<td>30.4</td>
<td>12</td>
<td>0.066</td>
<td>passive</td>
<td>passive</td>
<td>0.335</td>
<td>1.76</td>
</tr>
<tr>
<td>0.355</td>
<td>2</td>
<td>12</td>
<td>0.167</td>
<td>12</td>
<td>29.4</td>
<td>12</td>
<td>0.048</td>
<td>passive</td>
<td>passive</td>
<td>0.355</td>
<td>2</td>
</tr>
<tr>
<td>0.403</td>
<td>2.57</td>
<td>14</td>
<td>0.102</td>
<td>14</td>
<td>27.8</td>
<td>14</td>
<td>0.072</td>
<td>passive</td>
<td>passive</td>
<td>0.403</td>
<td>2.57</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.352</td>
<td>89.4</td>
<td>4</td>
<td>0.140</td>
<td>4</td>
<td>-7.2</td>
<td>4</td>
<td>0.278</td>
<td>passive</td>
<td>passive</td>
<td>4.352</td>
</tr>
<tr>
<td></td>
<td>5.992</td>
<td>90.4</td>
<td>490</td>
<td>0.5</td>
<td>2.99</td>
<td>-5.8</td>
<td>2</td>
<td>0.12</td>
<td>passive</td>
<td>passive</td>
<td>5.992</td>
</tr>
<tr>
<td></td>
<td>4.813</td>
<td>90.4</td>
<td>2</td>
<td>0.758</td>
<td>2</td>
<td>-6.6</td>
<td>2</td>
<td>0.303</td>
<td>record</td>
<td>error</td>
<td>4.813</td>
</tr>
<tr>
<td></td>
<td>5.116</td>
<td>91.08</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-5.0</td>
<td>0</td>
<td>0.4</td>
<td>record</td>
<td>error</td>
<td>5.116</td>
</tr>
<tr>
<td></td>
<td>5.116</td>
<td>91.08</td>
<td>2</td>
<td>0.455</td>
<td>-2</td>
<td>-1.4</td>
<td>-2</td>
<td>0.598</td>
<td>record</td>
<td>error</td>
<td>5.116</td>
</tr>
<tr>
<td></td>
<td>5.971</td>
<td>88</td>
<td>-298</td>
<td>0.5</td>
<td>-4.2</td>
<td>1.54</td>
<td>-2</td>
<td>0.188</td>
<td>record</td>
<td>error</td>
<td>5.971</td>
</tr>
<tr>
<td></td>
<td>7.066</td>
<td>84</td>
<td>844</td>
<td>0.5</td>
<td>-4.2</td>
<td>1.45</td>
<td>1</td>
<td>2.126</td>
<td>passive</td>
<td>passive</td>
<td>7.066</td>
</tr>
<tr>
<td></td>
<td>7.566</td>
<td>82</td>
<td>828</td>
<td>0.5</td>
<td>-3.49</td>
<td>2.37</td>
<td>1</td>
<td>0.628</td>
<td>passive</td>
<td>passive</td>
<td>7.566</td>
</tr>
<tr>
<td></td>
<td>8.066</td>
<td>80</td>
<td>80</td>
<td>0.5</td>
<td>-2.3</td>
<td>2.79</td>
<td>1</td>
<td>0.109</td>
<td>passive</td>
<td>passive</td>
<td>8.066</td>
</tr>
<tr>
<td></td>
<td>8.175</td>
<td>79.56</td>
<td>-2</td>
<td>0.782</td>
<td>-2</td>
<td>1.6</td>
<td>-2</td>
<td>1.25</td>
<td>record</td>
<td>passive</td>
<td>8.175</td>
</tr>
<tr>
<td></td>
<td>8.957</td>
<td>78</td>
<td>78</td>
<td>1</td>
<td>-0.75</td>
<td>2.6</td>
<td>1</td>
<td>0.288</td>
<td>record</td>
<td>passive</td>
<td>8.957</td>
</tr>
<tr>
<td></td>
<td>9.245</td>
<td>77.42</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3.35</td>
<td>record</td>
<td>error</td>
<td>9.245</td>
</tr>
<tr>
<td></td>
<td>11.245</td>
<td>77.42</td>
<td>2</td>
<td>0.712</td>
<td>2</td>
<td>-0.6</td>
<td>2</td>
<td>3.35</td>
<td>record</td>
<td>error</td>
<td>11.245</td>
</tr>
<tr>
<td></td>
<td>11.957</td>
<td>80</td>
<td>80</td>
<td>1</td>
<td>1.57</td>
<td>-1.6</td>
<td>2</td>
<td>0.98</td>
<td>passive</td>
<td>passive</td>
<td>11.957</td>
</tr>
</tbody>
</table>

This has reached its stable state which is not really suitable. It may be also interesting to study an adaptive quantum Bolduc and Vangheluwe (2003) scheme allowing the quantum size to be changed during the computation.

**APPENDIX**

**A.1 DEVSJAVA atomic component for a single ordinary differential equation**

This atomic model has no input and so on it has no external and confluent transition functions.

```java
public class integrator extends ViewableAtomic {
  protected double q, q1, dq, sigma;
  protected double d;

  public integrator() {
    this("integrator");
  }

  public integrator(String name) {
    super(name);
    addOutput("y");
  }

  public void initialize() {
    d = 0.15;
    q = q1 = 0.0;
    dq = 0.2*(2.0-q);
    sigma = computeSigma();
    holdIn("init", 0);
  }

  public void delint() {
    if (phaseIs("init")) {
      holdIn("active", sigma);
    }
    if (phaseIs("active")) {
      q1 = q1 + dq * sgn(dq);
      q = q1;
      dq = 0.2*(2.0-q);
      sigma = computeSigma();
      holdIn("active", h);
    }
  }

  private double computeSigma() {
    if (-0.0000001 < dq && dq < 0.0000001)
      return INFINITY;
    else
      return (d-abs(q-q1))/abs(dq);
  }

  public void output() {
    message = new message();
    if (phaseIs("active")) {
      m.add(makeContent("y", new double[]{q1 + dq * sgn(dq)}));
    }
    return m;
  }

  private double abs(double v) {
    if (v < 0)
      return -v;
    else return v;
  }

  private int sgn(double v) {
    if (v < 0)
      return -1;
  }
```
A.2 DEVSJAVA external transition function of component I2 then I1

```java
public void deltext(double e, message m) {
    for (int i = 0; i < m.getLength(); i++) {
        if (messageOnPort(m, "q1", i)) {
            entity ent = m.getValOnPort("q1", i);
            doubleEnt dEnt = (doubleEnt) ent;
            q1 = dEnt.getv();
            double aq = q;
            q = q - dq * e;
            dq = 0.5 * v - 0.8 * aq + 0.5 * q1;
            sigma = computeSigma();
            holdIn("active", sigma);
        }
    }
    for (int i = 0; i < m.getLength(); i++) {
        if (messageOnPort(m, "v", i)) {
            entity ent = m.getValOnPort("v", i);
            doubleEnt dEnt = (doubleEnt) ent;
            v = dEnt.getv();
            holdIn("active", 0);
        }
    }
}
```

public void deltext(double e, message m) {
    for (int i = 0; i < m.getLength(); i++) {
        if (messageOnPort(m, "q2", i)) {
            entity ent = m.getValOnPort("q2", i);
            doubleEnt dEnt = (doubleEnt) ent;
            q2 = dEnt.getv();
            double aq = q;
            q = q - dq * e;
            dq = q;
            sigma = computeSigma();
            holdIn("active", sigma);
        }
    }
}

REFERENCES


AUTHOR LISTING
### AUTHOR LISTING

<table>
<thead>
<tr>
<th>Name</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abelha A</td>
<td>387/399/409</td>
</tr>
<tr>
<td>Duarte J</td>
<td>414</td>
</tr>
<tr>
<td>Abidi H</td>
<td>288</td>
</tr>
<tr>
<td>Duinkerken M.B.</td>
<td>300</td>
</tr>
<tr>
<td>Adegoke A</td>
<td>91</td>
</tr>
<tr>
<td>Dupleac D</td>
<td>309</td>
</tr>
<tr>
<td>Adelantado M</td>
<td>69</td>
</tr>
<tr>
<td>Fernandes N.O.</td>
<td>297</td>
</tr>
<tr>
<td>Albert V</td>
<td>429/434</td>
</tr>
<tr>
<td>Fikej J</td>
<td>225</td>
</tr>
<tr>
<td>Alves L</td>
<td>409</td>
</tr>
<tr>
<td>Fischer M</td>
<td>5</td>
</tr>
<tr>
<td>Amblard F</td>
<td>38</td>
</tr>
<tr>
<td>Foure D</td>
<td>429</td>
</tr>
<tr>
<td>Azzopardi L.M.</td>
<td>112</td>
</tr>
<tr>
<td>Gago P</td>
<td>419</td>
</tr>
<tr>
<td>Bachelet B</td>
<td>354</td>
</tr>
<tr>
<td>Geraldes C.A.S.</td>
<td>275/281</td>
</tr>
<tr>
<td>Barbier G</td>
<td>130</td>
</tr>
<tr>
<td>Gomes M</td>
<td>414</td>
</tr>
<tr>
<td>Bas D</td>
<td>365</td>
</tr>
<tr>
<td>Goncalves P</td>
<td>409</td>
</tr>
<tr>
<td>Bataus M</td>
<td>168</td>
</tr>
<tr>
<td>Góra W</td>
<td>338</td>
</tr>
<tr>
<td>Belkadi K</td>
<td>404</td>
</tr>
<tr>
<td>Gourgand M</td>
<td>209</td>
</tr>
<tr>
<td>Bellochhi G</td>
<td>354</td>
</tr>
<tr>
<td>Hajdasz M</td>
<td>247</td>
</tr>
<tr>
<td>Bernal C</td>
<td>161</td>
</tr>
<tr>
<td>Hill D.R.C.</td>
<td>76/130/354/365</td>
</tr>
<tr>
<td>Braekers K</td>
<td>257</td>
</tr>
<tr>
<td>Ighoroje U.B.</td>
<td>61</td>
</tr>
<tr>
<td>Brandão R.M.</td>
<td>14</td>
</tr>
<tr>
<td>Janssens G.K.</td>
<td>252/257</td>
</tr>
<tr>
<td>Breton V</td>
<td>381</td>
</tr>
<tr>
<td>Jassbi J</td>
<td>27</td>
</tr>
<tr>
<td>Brito A.C.</td>
<td>33/125</td>
</tr>
<tr>
<td>Ježek D</td>
<td>99</td>
</tr>
<tr>
<td>Brodziak R</td>
<td>338</td>
</tr>
<tr>
<td>Cabral A</td>
<td>414</td>
</tr>
<tr>
<td>Kandel C</td>
<td>235/240/288</td>
</tr>
<tr>
<td>Cálinoiu C</td>
<td>320</td>
</tr>
<tr>
<td>Kavička A</td>
<td>225</td>
</tr>
<tr>
<td>Camacho A.M.</td>
<td>161</td>
</tr>
<tr>
<td>Keruzeci J</td>
<td>365</td>
</tr>
<tr>
<td>Camilleri L</td>
<td>112</td>
</tr>
<tr>
<td>Klaas A</td>
<td>266</td>
</tr>
<tr>
<td>Carmo-Silva S</td>
<td>297</td>
</tr>
<tr>
<td>Klumpp M</td>
<td>235/240/263</td>
</tr>
<tr>
<td>Carvalho M.S.</td>
<td>201/275/281</td>
</tr>
<tr>
<td>Chaudron J.-B.</td>
<td>69</td>
</tr>
<tr>
<td>Kocy T</td>
<td>108</td>
</tr>
<tr>
<td>Conley W</td>
<td>197/343</td>
</tr>
<tr>
<td>Kožusznik J</td>
<td>45/48</td>
</tr>
<tr>
<td>Correia A</td>
<td>399</td>
</tr>
<tr>
<td>Kuchaf Š</td>
<td>108</td>
</tr>
<tr>
<td>Correia A.G.</td>
<td>326</td>
</tr>
<tr>
<td>Lardy R</td>
<td>354</td>
</tr>
<tr>
<td>Cortez P</td>
<td>117/326</td>
</tr>
<tr>
<td>Laroque C</td>
<td>5/266</td>
</tr>
<tr>
<td>Costa À</td>
<td>392</td>
</tr>
<tr>
<td>Laureano R.M.S.</td>
<td>117</td>
</tr>
<tr>
<td>Cuevas D</td>
<td>387</td>
</tr>
<tr>
<td>Leitão A</td>
<td>53</td>
</tr>
<tr>
<td>Dahmani A</td>
<td>20</td>
</tr>
<tr>
<td>Leroy A</td>
<td>365</td>
</tr>
<tr>
<td>Dangelmaier W</td>
<td>266</td>
</tr>
<tr>
<td>Letagneaux A</td>
<td>365</td>
</tr>
<tr>
<td>de Haan A.B.</td>
<td>314</td>
</tr>
<tr>
<td>Lodewijks G</td>
<td>300</td>
</tr>
<tr>
<td>de Oliveira C.B.</td>
<td>33/125</td>
</tr>
<tr>
<td>Louvet B</td>
<td>381</td>
</tr>
<tr>
<td>Debacker M</td>
<td>349</td>
</tr>
<tr>
<td>Machado J</td>
<td>387/399/409</td>
</tr>
<tr>
<td>Dekhici L</td>
<td>404</td>
</tr>
<tr>
<td>................................</td>
<td>414/419</td>
</tr>
<tr>
<td>Deroussi L</td>
<td>381</td>
</tr>
<tr>
<td>Dian A</td>
<td>201</td>
</tr>
<tr>
<td>Mahmoudzadeh M</td>
<td>27</td>
</tr>
<tr>
<td>Dian L.S.</td>
<td>53/150</td>
</tr>
<tr>
<td>Maigne L</td>
<td>381</td>
</tr>
<tr>
<td>Dobre C</td>
<td>182</td>
</tr>
<tr>
<td>Makarov A</td>
<td>177</td>
</tr>
<tr>
<td>Duarte A</td>
<td>387</td>
</tr>
<tr>
<td>Marin M.M.</td>
<td>161</td>
</tr>
</tbody>
</table>

445
AUTHOR LISTING

Marlewski A. .......... 247  Romero G. .......... 373
Marques E.M.D. .... 190  Rooda J.E. ....... 314
Marques P.S. ....... 125  Rossetti R.J.F. ... 217
Marques V. ........... 414  Rua F. ............. 419
Martinez M.L. ....... 373  Sá T. ............. 409
Martinović J. ...... 108  Safarik J. ......... 102
Mazel C. ............ 76/365  Salazar M. ........ 387
Mihalescu B. ........ 320  Sampaio P.N.M. ... 190
Miranda M. ........ 387/409  Santos M.F. ....... 399/414/419
Moreno I. .......... 373  Sarramia D. ....... 381
Moro S. ............ 117  Schiffelers R.R.H. ... 314
Mouly D. .......... 365  Schott D.L. .......... 300
Muzy A. ........... 365  Sebastián M.A. .... 161
Netrvalova A. ...... 102  Selberherr S. ...... 177
Neves J. ......... 392/399/414  Silva Á. .......... 419
.......................... 419  Siregar P. ........ 76
Nicolici S. .......... 309  Siron P. ............ 69
Ništá I.P. .......... 309  Sousa J.J.F. .... 190
Nketsa A. .......... 434  Sourček E.M. ...... 155
Novais P. .......... 392  Stoop P. .......... 300
Oliveira J.A. ....... 150  Studzinski J. ...... 333
Oliveira S. ....... 150  Suwittayaruk P ... 145
Oliveira T. ....... 392  Sverdlov V. ...... 177
Osintsev D. ....... 177  Taillardier P. ... 138
Ourbih-Baghdali L. .. 20  Taillardier P. ... 38
Ourbih-Tari M. .. 20  Tanguy A. ....... 209/404
Pangilinan J.M. .... 252  Telhada J. ....... 201
Pascal J.-C. ........ 429  Therond O. .......... 138
Passerat-Palmbach J. .... 76  Tinoco J. .......... 326
Pearce G. ........... 373  Touraille L. ....... 365
Peito F. ........... 53  Traoré M.K. ..... 61/91
Peixoto H. .......... 399  Ullrich C. .......... 349
Pereira J.L.F. ....... 217  Urbaniai A. ...... 338
Pereira G.A.B. .... 53/150/275/281  van der Geest J.T. ... 300
Perkinson N.D. ....... 373  Pham T.-H. ....... 209  Van Goubergen D ... 145
Pinel F. ........... 130  Van Utterbeeck F ... 349
Pontes G. .......... 387  Vasilii D. .......... 168
Pop F. ............ 84  Vasilii N. .......... 168/320
Porta Nova A.M.O. ... 14  Weinbub J. ........ 177
Portela F. .......... 419  Wick C. .......... 240
Prisecaru I. ....... 309  Zondervan E. ..... 314
Quintas C. .......... 409
Renken H. .......... 5/266