17th INTERNATIONAL INDUSTRIAL SIMULATION CONFERENCE 2019

ISC'2019

EDITED BY

Jorge Sequeira Gonçalves Paulo

and

Manuel Ferreira Calado João

June 5-7, 2019

LISBON, PORTUGAL

A Publication of EUROSIS-ETI

17th Industrial Simulation Conference 2019

LISBON, PORTUGAL

JUNE 5-7, 2019

Organised by

ETI - The European Technology Institute

Sponsored by

EUROSIS

University of Skövde

University of Žilina

Co-Sponsored by

Ghent University

GODAN

Hosted by

ISEL
Instituto Superior de
Engenharia de Lisboa

Lisbon, Portugal

EXECUTIVE EDITOR

PHILIPPE GERIL
(BELGIUM)

EDITORS

General Conference Chair
Paulo J.S. Gonçalves, Instituto Politécnico de Castelo Branco and
IDMEC,Instituto Superior Técnico, Lisbon, Portugal

General Program Chair
João Calado, ISEL, Lisbon, Portugal

Past Conference Chair
José Machado, Uminho, Braga, Portugal

Publications Chairs
Peter Lawrence, Swinburne University, Australia
Yan Luo, NIST, Gaithersburg, USA
Wan Ahmad Yusmawiza Wan Yusoff, International Islamic Univ. Malaysia, Kuala Lumpur, Malaysia

INTERNATIONAL PROGRAMME COMMITTEE

Discrete Simulation Methodology
Sang Do No, Sungkyunkwan University, South Korea
Vincent Rodin, European Centre for VR, Brest, France

Discrete Simulation Analysis
Pascal Berruet, Université Bretagne Sud, Lorient Cedex, France

Discrete Simulation Methodology, Languages and Tools
Matthias Becker, University Hannover, Hannover, Germany
Helge Hagenauer, Universitaet Salzburg, Salzburg, Austria
Bernhard Heinzl, TU Wien, Vienna, Austria
Sophie Hennequin, ENIM, Metz Cedex, France
Stefano Marrone, Seconda Universita di Napoli. Naples, Italy
Jiri Safarik, University of West Bohemia, Plzen, Czech Republic

Open Source Simulation Languages and Tools
Sang Do No, Sungkyunkwan University, South Korea
Philippe Geril, ETI Bvba, Ostend, Belgium
Maciej Majewski, Koszalin University of Technology, Poland

Verification, Validation and Accreditation
Maciej Majewski, Koszalin University of Technology, Poland
Roberto Revetria, University of Genoa, Genoa, Italy

Simulation in Computer Science
Ernst Kesseler, NLR, Amsterdam, The Netherlands

Ambient Intelligence and Simulation
Sang Do No, Sungkyunkwan University, South Korea
Maciej Majewski, Koszalin University of Technology, Poland
Selwyn Piramuthu, University of Florida, Gainesville, USA

Artificial Intelligence
Chrissanti Angeli, Technological Institute of Piraeus, Athens, Greece
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Castelo Branco, Portugal

INTERNATIONAL PROGRAMME COMMITTEE

Sang Do No, Sungkyunkwan University, South Korea
Sanaul Haque, Institute of Technology, Carlow, Ireland
Biju Theruvil Sayed, Dhofar University, Salalah, Sultanate of Oman

VR and Graphical Simulation in Industry
Track Chair: Guodong Shao, NIST, Gaithersburg, USA
Pascal Berruet, Université Bretagne Sud, Lorient Cedex, France
Sanaul Haque, Institute of Technology, Carlow, Ireland
Fabrizio Lamberti, Politecnico di Torino, Turin, Italy
Christoph Laroque, University of Applied Sciences, Zwickau, Germany
Sudhir Mudur, Concordia University, Montréal, Canada
Paulo Moura Oliveira, UTA University, Vila Real, Portugal
Dorin-Mircea Popovici, Universitatea OVIDIUS din Constanta, Constanta, Romania
Marcos A.Rodrigues Sheffield Hallam University, Sheffield, United Kingdom

Augmented Reality and Pervasive Systems in Factories
Ricardo Chalmeta, Universitat Jaume I., Castillon, Spain
Alessandro Genco, University of Palermo, Palermo, Italy
Sanaul Haque, Institute of Technology, Carlow, Ireland
Eleni Mangina, University College Dublin (UCD), Dublin, Ireland

Simulation-based evaluation of interactive systems
Sanaul Haque, Institute of Technology, Carlow, Ireland
Stefan Hellfeld, FZI - House of Living Labs, Karlsruhe, Germany
Benjamin Herd, King's College, London, United Kingdom

Simulation and Training
Manuel Alfonseca, Universidad Autonoma de Madrid, Madrid, Spain
Eleni Mangina, University College Dublin (UCD), Dublin, Ireland
Dorin-Mircea Popovici, Universitatea OVIDIUS din Constanta, Constanta, Romania

Industrial Internet of Things
Ricardo Chalmeta, Universitat Jaume I., Castillon, Spain
Norbert Jastroch, MET Communications GmbH,Bad Homburg, Germany
Sanaul Haque, Institute of Technology, Carlow, Ireland
Maciej Majewski, Koszalin University of Technology, Poland

e-Business in Collaborative Engineering
Ricardo Chalmeta, Universitat Jaume I., Castillon, Spain
Heiko Duin, BIBA, Bremen, Germany
Sanaul Haque, Institute of Technology, Carlow, Ireland
Norbert Jastroch, MET Communications GmbH, Bad Homburg, Germany

Simulation in Nanotechnology
Clemens Heitzinger, Cambridge University, Cambridge, United Kingdom
Yong K. Kim, University of Massachusetts Dartmouth, Dartmouth, USA
Javier Marin, ETSI, University of Malaga, Malaga, Spain

Complex Systems Modelling
Track Chair: Igor N Litvine, Nelson Mandela Metropolitan University, Port Elizabeth, South Africa
Frantisek Capkovic, Slovak Academy of Sciences, Bratislava, Slovak Republic
Alexandre Nketsa, LAAS, Toulouse, France
Alfonso Urquia, UNED, Madrid, Spain

Simulation of Complex Multiprocessor Systems
Orhan Gemikonakli, Middlesex University, London, United Kingdom

Simulation in Electronics, Computer and Telecommunications
Teresa Alvarez, University of Valladolid, Valladolid Spain
Christos Bouras, University of Patras, Patras, Greece
Adnane Latif, Cadi Ayyad University, Marrakech, Morocco
Silvia Mirri, University of Bologna, Bologna, Italy
Maurizio Palesi, Universita di Catania, Catania, Italy
Marco Roccetti, University of Bologna, Bologna, Italy

INTERNATIONAL PROGRAMME COMMITTEE

Fernando Boronat Segui, UPV, Valencia, Spain
Pedro Sousa, University of Minho, Portugal

Simulation in Aerospace
Reza Azadegan, Urmia University, Urmia, Iran
Wolfgang Kuehn, University of Wuppertal, Wuppertal, Germany

Marine Simulation
Sergeij Kalashnkikow, DANFOSS, Austria

Simulation in Military and Defense
Roberto de Beauclair Seixas, IMPA, Rio de Janeiro, Brazil
Carlos Palau, UPV, Valencia, Spain
Matthias Reuter, CUTEC Gmbh, TU-Clausthal, Clausthal, Germany
Joseph M. Saur, Regent University, USA

Civil and Building Engineering
Alistair Borthwick, Oxford University, Oxford, United Kingdom
Graham Saunders, Loughborough University, Loughborough, United Kingdom

Simulation in Chemical, Petroleum and Mining Engineering
Pouria Homayonifar, Acona Flow Technology, Skien, Norway
Mohamad R. Riazi, Kuwait University, Kuwait

Simulation in Energy and Power Systems
Sérgio Leitão, CETAV/UTAD, Vila Real, Portugal
Sergeij Kalashnkikow, DANFOSS, Austria
Janos-Sebestyen Janosy, KFKI Atomic Energy Research Institute, Budapest, Hungary

Servitization
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Castelo Branco, Portugal
Sang Do No, Sungkyunkwan University, South Korea

Automation/Robotics
Leopoldo Arnesto Angel, UPV, Valencia, Spain
Maciej Majewski, Koszalin University of Technology, Poland
Martin Mellado, UPV, Valencia, Spain
Paulo Moura Oliveira, UTA University, Vila Real, Portugal
Sergiu-Dan Stan, Technical University of Cluj-Napoca, Cluj-Napoca, Romania

Machine to Machine
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Castelo Branco, Portugal
Sang Do No, Sungkyunkwan University, South Korea

Intelligent Connected Systems
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Castelo Branco, Portugal
Sang Do No, Sungkyunkwan University, South Korea
Maciej Majewski, Koszalin University of Technology, Poland

Cyber Physical Systems Simulation
Adina Magda Florea, University POLITEHNICA of Bucharest, Bucharest, Romania
Pascal Berruet, Université Bretagne Sud, Lorient Cedex, France
Cornel Burileanu, University POLITEHNICA of Bucharest, Bucharest, Romania
Corneliu Bilan, University POLITEHNICA of Bucharest, Bucharest, Romania
Maciej Majewski, Koszalin University of Technology, Poland
Eleni Mangina, University College Dublin (UCD), Dublin, Ireland

Fog computing for Robotics and Industrial Automation
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Castelo Branco, Portugal
Sang Do No, Sungkyunkwan University, South Korea

Simulation in Engineering Processes
Chrissanti Angeli, Technological Institute of Piraeus, Athens, Greece
Alejandra Gomez Padilla, University of Guadalajara, Mexico
Bernhard Heinzl, TU Wien, Vienna, Austria

INTERNATIONAL PROGRAMME COMMITTEE

Jan Studzinski, Polish Academy of Sciences, Warsaw, Poland
Joao Tavares, University of Porto, Porto, Portugal
Henk Versteeg, Loughborough University, Loughborough, United Kingdom

Model Driven Engineering
Ricardo Chalmeta**,** Universitat Jaume I., Castillon, Spain
Norbert Jastroch, MET Communications GmbH,Bad Homburg, Germany
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Castelo Branco, Portugal

Simulation in Industrial Design and Product Design
Chiara Catalano, IMATI-CNR, Genoa, Italy
Yan Luo, NIST, Gaithersburg, USA
Catarina Rizzi, University of Bergamo, Bergamo, Italy

Simulation in Automotive Systems
Track Chair: José Machado, Uminho, Braga, Portugal
Antonio Abelha, UMinho, Braga,Portugal
Naoufel Cheikhrouhou, EPFL, Lausanne, Switzerland
Valerian Croitorescu, University of Bucharest, Bucharest, Romania
José António Oliveira, University of Minho in Portugal
Julien Richert, Daimler AG, GR/PAA, Sindelfingen, Böblingen, Germany

Choice Modelling Methods and Applications
Karim Labadi, ECAM-EPMI, QUARTZ-Lab, France

Simulation in Multibody Systems
Ignacio Garcia-Fernandez, University of Valencia, Valencia, Spain

Organization and Management in Collaborative Engineering
Eduardo Castellano, IK4-IKERLAN, Mondragon-Arrasate, Spain
Florina Covaci, "Babes-Bolyai" University, Cluj-Napoca, Romania
Iris Graessler, Cologne University of Applied Sciences, Cologne, Germany

Supporting Technologies
Ricardo Gonçalves, Univ. Nova de Lisboa, UNINOVA, Caparica, Portugal

Formal Methods, Techniques and Design
Yan Luo, NIST, Gaithersburg, USA
Stefano Marrone, Seconda Universita di Napoli, Caserta, Italy

Engineering of Embedded Systems
Petr Hanacek, Brno University of Technology, Brno, Czech Republic
Jose Machado, University do Minho, Braga, Portugal
Stefano Marrone, Second University of Naples, Caserta, Italy

Collaborative Engineering data management and information modeling
Uwe Baake, Daimler Buses, Evobus GmbH, Neu-Ulm, Germany
Florina Covaci, "Babes-Bolyai" University, Cluj-Napoca, Romania
Joachim Herbst, Daimler AG, Research and Technology, Ulm, Germany
Jiri Vondrich, Czech Technical University, Prague, Czech Republic

Collaborative Engineering process management and simulation
Antonio Carvalho, Brito FEUP, University of Porto, Porto, Portugal
Florina Covaci, "Babes-Bolyai" University, Cluj-Napoca, Romania
Imed Kacem, University of Technology of Troyes, Troyes, France
Carlo Meloni, Politecnico di Bari, Bari, Italy
Roberto Montemanni, IDSIA, Manno, Switzerland
Maurizio Palesi, University of Catania, Catania, Italy
Christian Andreas Schumann, University of Zwickau, Zwickau, Germany

Collaborative Environments for Virtual Teams
Veronique Baudin, LAAS-CNRS, Université de Toulouse, Toulouse Cedex, France
Paul Stratil, SAP Deutschland SE & Co. KG, Walldorf, Germany

Collaborative Enhanced Lean Manufacturing Simulation
Jean-Claude Hochon, AIRBUS, Blagnac, France
Jose Antonio V. Oliveira, University of Minho, Braga, Portugal

INTERNATIONAL PROGRAMME COMMITTEE

Practical applications and experiences
James Gao, University of Greenwich, Chatham Maritime, Kent, United Kingdom
Sanaul Haque, Institute of Technology, Carlow, Ireland
Irfan Manarvi, Prince Sultan University, Riyadh, Kingdom of Saudi Arabia

Factory Planning and Control for SME's
Sílvio do Carmo Silva, University of Minho, Portugal
Pascal Berruet, Université Bretagne Sud, Lorient Cedex, France
Paulo J.S. Gonçalves, Polytechnic Institute of Castelo Branco, Portugal
Nuno O. Fernandes, Polytechnic Institute of Castelo Branco, Portugal
Veronique Limere, Ghent University, Ghent, Belgium

Data-Driven Tradespace Exploration and Analysis
Philippe Geril, ETI, Ostend, Belgium
Sanaul Haque, Institute of Technology, Carlow, Ireland
Peter K. K. Loh, Nanyang Technological University, Singapore

Production of One: Digital Direct Manufacturing and Additive Manufacturing
Hans Degraeuwe, Degraeuwe Consulting, Brussels, Belgium

Smart Factory 1.0 and Industry 4.0
Florina Covaci, "Babes-Bolyai" University, Cluj-Napoca, Romania
Norbert Jastroch, MET Communications GmbH,Bad Homburg, Germany
Philippe Geril, ETI, Ostend, Belgium
Maciej Majewski, Koszalin University of Technology, Poland

Simulation in Steel Manufacturing and New Materials Manufacturing

Simulation in Electronics Manufacturing
Gerald Weigert, Dresden University of Technology, Germany

Simulation-based Optimization in Industry
Bernhard Heinzl, TU Wien, Vienna, Austria
Karim Labadi, ECAM-EPMI, QUARTZ-Lab, France
Amos H.C. Ng, University of Skovde, Skovde, Sweden
Anna Syberfeldt, University of Skovde, Skovde, Sweden

Apparel and Textile Simulation
Jocelyn Bellemare, University of Quebec in Montreal (UQAM), Montréal (Québec) Canada
Ali Shams Nateri, University of Guilan, Iran

Simulation in Logistics, Transport and Harbour Simulation
El-Houssaine Aghezzaf, Ghent University, Ghent, Belgium
Maria Sameiro Carvalho, University of Minho, Guimaraes, Portugal
Remy Dupas, Université de Bordeaux , Bordeaux, France
Olivier Grunder, UTBM, Belfort, France
Isabel Garcia Guttierez, Univ. Carlos III de Madrid, Madrid, Spain .
Peter Lawrence, Swinburne University, Lilydale, Australia
Marie-Ange Manier, UTBM, Belfort, France
Roberto Montemanni, IDSIA, Manno-Lugano, Switzerland
Rosaldo Rossetti, University of Porto, Porto, Portugal
Hans Veeke, TU Delft, Delft, The Netherlands
Pengjun Zheng, University of Southampton, Southampton, United Kingdom

Simulation in Supply Chains
Florina Covaci, "Babes-Bolyai" University, Cluj-Napoca, Romania
Matthias Klumpp, ild Institute for Logistics and Service Management at FOM, Essen, Germany
Eleni Mangina, University College Dublin (UCD), Dublin, Ireland
Sang Do No, Sungkyunkwan University, South Korea

Hospital Logistics Simulation
Track Chair: Giorgio Romanin-Jacur, University of Padova, Vicenza, Italy
Antonio Abelha, Universidade do Minho, Braga, Portugal
Joel Colloc, Le Havre Normandy University, Le Havre, France
Jose Machado, University of Minho, Braga, Portugal

INTERNATIONAL PROGRAMME COMMITTEE

Jose Antonio V. Oliveira, University of Minho, Braga, Portugal
Peter Summons, University of Newcastle, Australia

Transport Simulation
Jose Antonio V. Oliveira University of Minho, Braga, Portugal

Intelligent Transport Systems
Track Chair: Anna Syberfeldt, University of Skovde, Skovde, Sweden
Petr Hanacek, Brno University of Technology, Brno, Czech Republic
Jairo Montoya Torres, Universidad de la Sabana, Chia, Columbia

Urban Simulation
Hans Degraeuwe, Degraeuwe Consulting Consulting, Belgium

Statistical Analysis and Data Mining of Business Processes
Paulo Alexandre Ribeiro Cortez, University of Minho, Guimaraes, Portugal
Sanaul Haque, Institute of Technology, Carlow, Ireland
Selwyn Piramithu, University of Florida, Gainesville, USA
Inna Pivkina, New Mexico State University, USA
Danny Van Welden, KBC Bank, Brussels, Belgium

Applications using Data Mining
Marco Furini, Università di Modena e Reggio Emilia, Reggio Emilia, Italy
Sanaul Haque, Institute of Technology, Carlow, Ireland
Maciej Majewski, Koszalin University of Technology, Poland

Applications using data mining and open source text information mining and analysis
Marco Furini, Università di Modena e Reggio Emilia, Reggio Emilia, Italy
Sang Do No, Sungkyunkwan University, South Korea
Maciej Majewski, Koszalin University of Technology, Poland

Patient Centered Health Care with Meaningful Use of IT
Sanaul Haque, Institute of Technology, Carlow, Ireland

Geosimulation and Big Data
Marwan Al-Akaidi, Arab Open University, Kuwait

Simulation Applied to Economics
J. Manuel Feliz Teixeiro, FEUP-University of Porto, Porto, Portugal
Leonid Novickis, IDC Information Technologies, Riga, Latvia
Kamelia Stefanova, University of National and World Economy, Sofia, Bulgaria

Agents in Business Automation and Economics
Florina Covaci, "Babes-Bolyai" University, Cluj-Napoca, Romania
Martin Fredriksson, Blekinge institute of technology, Karlskrona, Sweden
Markus Koch, Orga Systems GmbH, Paderborn, Germany
Isabel Praca, Polytechnic of Porto, Porto, Portugal

Simulation in Business and Economics
Sudhir Mudur, Concordia University, Montréal, Canada
Yan Luo NIST, Gaitherburg, USA

Simulation in Business Games
Marco Furini, Università di Modena e Reggio Emilia, Reggio Emilia, Italy
David Wortley, Coventry University, Coventry, United Kingdom

Simulation and Analysis of Environmental effects of business
Dietmar Boenke, Hochschule Reutlingen, Reutlingen, Germany
Elias A. Hadzilias, Universite Catholique de Lille, Lille, France
Roberto Razzoli, University of Genova, Genova, Italy

Simulation Tools for Business Management & Business Intelligence
Carlo Meloni, Politecnico di Bari, Bari, Italy

Simulation in OR and Knowledge Management
Chrissanthi Angeli, Technological Institute of Piraeus, Athens, Greece
Imed Kacem, Université Paul Verlaine Metz, Metz, France

INTERNATIONAL PROGRAMME COMMITTEE

Christophe Roche, Universite de Savoie, Le Bourget du Lac Cedex, France
Biju Theruvil Sayed, Dhofar University, Salalah, Sultanate of Oman

Simulation in OR and Knowledge Management
Chrissanthi Angeli, Technological Institute of Piraeus, Athens, Greece
Imed Kacem, Université Paul Verlaine Metz, Metz, France
Christophe Roche, Universite de Savoie, Le Bourget du Lac Cedex, France
Biju Theruvil Sayed, Dhofar University, Salalah, Sultanate of Oman

Emergency Management and Risk Analysis Management
Ismail Bouassida, LAAS-CNRS, Université de Toulouse, Toulouse, France
Konstantinos Kirytopoulos, University of the Aegean, Greece
Lode Vermeersch, Credendo, Brussels, Belgium

Policy Modelling and Simulation in E-Management, E-Government, E-Commerce and E-Trade
Vrassidas Leopoulos, Nat. technical University of Athens, Athens, Greece

Political and Government Simulation
Joseph M. Saur, Regent University, USA

Sustainable Development Management
Philippe Geril, ETI Bvba, Ostend, Belgium

HORIZON 2020 Workshop
Valerian Croitorescu, University of Bucharest, Bucharest, Romania

INDUSTRIAL SIMULATION 2019

EUROSIS-ETI Publications are ISI-Thomson, IET, SCOPUS and Elsevier Engineering Village referenced
Legal Repository: Koninklijke Bibliotheek van België, Keizerslaan 4, 1000 Brussels, Belgium
CIP 12.620 D/2011/12.620/1

Selected papers of this conference are published in scientific journals.

For permission to publish a complete paper write EUROSIS, c/o Philippe Geril, ETI Executive Director, Greenbridge NV, Ghent University - Ostend Campus, Wetenschapspark 1, Plassendale 1, B-8400 Ostend, Belgium.

EUROSIS is a Division of ETI Bvba, The European Technology Institute, Torhoutsesteenweg 162, Box 04.02, B-8400 Ostend, Belgium

Printed and bound in Belgium by Reproduct NV, Ghent, Belgium
Cover Design by C-Design Bvba, Wielsbeke, Belgium

EUROSIS-ETI Publication
ISBN: 978-94-92859-07-5
EAN: **978-94-92859-07-5**

Preface

It is our privilege and pleasure to welcome you to the 17th Industrial Simulation Conference – ISC'2019, held in Association with ISEL (Instituto Superior de Engenharia de Lisboa) and IDMEC. As in previous editions, this conference gives a complete overview of this year's industrial simulation related research and to provide an annual status report on present day industrial simulation research within the European Community and the rest of the world in line with European industrial research projects

We have some twenty participants from eleven countries with presentations grouped in different themes such as: Market Analysis, Datamining Forecasting and Data Constructs, Decision Management Simulation, Manpower Management, Automation/Robotics and Digital Twins, Engineering Simulation, Traffic Simulation in Urban Mobility; to mention a few. In this conference, we will have the opportunity to listen to Helder Coelho, who will talk about Street Violence and Social Conflicts, Joe□l Colloc giving a talk on The Evolution of Artificial Intelligence towards autonomous systems with personality simulation and last but not least Lars Mönch, talking about Discrete-event Simulation Applications in Semiconductor Manufacturing.

We would like to welcome all participants and to thank all authors for sharing their knowledge and experience. Our thanks also go to all members of the Program Committee for the reviewing work that was key to maintaining the high scientific quality of ISC'2019. We are also grateful to the Keynote Speakers who are willing to share their extensive knowledge and experience in the field of simulation and modelling with us.

Furthermore, a special thanks to Philippe Geril from EUROSIS, whose continued dedication and hard work, as the conference organizer, has enabled the organization to maintain the standard expected for ISC'2019. Finally, we would like to express our gratitude to ISEL for its support and to the hosting venue for a job well done.

Finally, we would like to wish you all a fruitful and productive experience at the conference and an enjoyable and enriching stay in Lisbon.

Paulo J.S. Gonçalves*, João Calado**

*IDMEC, Instituto Politécnico de Castelo Branco, Portugal

**IDMEC, ISEL, Instituto Politécnico de Lisboa, Portugal

ISC'2019 General Conference* and Program** Chairs

CONTENTS

Preface ..... **XIII**
Scientific Programme **1**
Author Listing **141**

MARKET ANALYSIS

Investigating the Factors which affect the Performance of the EM Algorithm in Latent Class Models
Liberato Camilleri, Luke Spiteri and Maureen Camilleri **5**

Four and Eight Commodity Economic Equilibrium Problems with Extensive Substitution Possibilities
William Conley **11**

DATAMINING FORECASTING AND DATA CONSTRUCTS

A Smart Approach to harvest Date Forecasting
Beatriz Charneca, Vanda Santos, Ana Crespo, Henrique Vicente, Humberto Chaves, Jorge Ribeiro, Victor Alves and José Neves **19**

Operators' Structural Knowledge Evaluation as a Feedback in Computer-Based Training in Context of the Novel "Anna Karenina"
Victor Dozortsev and Anastasia Mironova **26**

DECISION MANAGEMENT SIMULATION

Definition of Strategies based on Simulation and Design of Experiments
António Abreu, José Requeijo, J. M. F. Calado and Ana Dias **35**

Situational Information System for Decision-Making Support regarding Industry-Natural Complexes
Alexander Fridman and Andrey Oleynik **41**

Reliability and Quality in Manufacturing
William Conley **46**

MANPOWER MANAGEMENT

Manpower Planning using Simulation and Heuristic Optimization
Oussama Mazari-Abdessameud, Johan Van Kerckhoven, Filip Van Utterbeeck and Marie-Anne Guerry **53**

CONTENTS

Modelling Fire Fighters Actions in a Forest Fire Simulation Algorithm
Yves Dumond **59**

Flexible and scalable Field Service with Condition Base Maintenance Models for simulating Large Scale Scenarios
Gabriel G. Castañé, Helmut Simonis, Kenneth N. Brown, Cemalettin Ozturk, Yiqing Lin and Mark Antunes **67**

AUTOMATION/ROBOTICS AND DIGITAL TWINS

Intelligent Robotic Devices using Knowledge from the Cloud
P.J.S. Gonçalves and F.M.S. Santos **77**

Influence of Task Allocation Patterns on Safety and Productivity in Human-Robot-Collaboration
Titanilla Komenda, Fabian Ranz and Wilfried Sihn **85**

Enhanced Interaction with Industrial Robots through Extended Reality Relying on Simulation-Based Digital Twins
Moritz Alfrink and Jürgen Roßmann **90**

A Case Study of a Digital Twin for Designing Intermodal Railways Operations for a Maritime Terminal
Emanuele Morra, Lorenzo Damiani, Roberto Revetria and Anastasiia Rozhok **98**

ENGINEERING SIMULATION

Influence of Parametric Uncertainty of selected Classification Models of Sewage Quality Indicators on the Sludge Volume Index Forecast
Bartosz Szelag, Jan Studziński and Izabela Rojek **105**

Automatically generated, Real-time, Logical Simulation of the Electrical System
Jerzy Kocerka and Michal Krześlak **111**

TRAFFIC SIMULATION AND URBAN MOBILITY

Acs Simulation and Mobile Phone Tracing to redraw Urban Bus Routes
Giuseppe Iazeolla and Fabio Pompei **117**

Policy Appraisal using Iteration Games in Artificial Transportation Systems
Hajar Baghcheband, Zafeiris Kokkinogenis and Rosaldo J.F. Rossetti **123**

CONTENTS

Using Simulation Games for Traffic Model Calibration
Gonçalo Leão, João Ferreira, Pedro Amaro and Rosaldo J.F. Rossetti **.....131**

SCIENTIFIC PROGRAMME

MARKET ANALYSIS

INVESTIGATING THE FACTORS WHICH AFFECT THE PERFORMANCE OF THE EM ALGORITHM IN LATENT CLASS MODELS

Liberato Camilleri, Luke Spiteri and Maureen Camilleri
Department of Statistics and Operations Research
University of Malta
Msida (MSD 06)
Malta
E-mail: liberato.camilleri@um.edu.mt

KEYWORDS
Latent class model, Market segmentation, EM algorithm, Monte Carlo simulation

ABSTRACT

Latent class models have been used extensively in market segmentation to divide a total market into market groups of consumers who have relatively similar product needs and preferences. The advantage of these models over traditional clustering techniques lies in simultaneous estimation and segmentation, which is carried out using the EM algorithm. The identification of consumer segments allows target-marketing strategies to be developed.

The data comprises the rating responses of 262 respondents to 24 laptop profiles described by four item attributes including the brand, price, random access memory (RAM) and the screen size. Using the facilities of R Studio, two latent class models were fitted by varying the number of clusters from 2 to 3.

The parameter estimates obtained from these two latent class models were used to simulate a number of data sets for each cluster solution to be able to conduct a Monte-Carlo study, which investigates factors that have an effect on segment membership and parameter recovery and affect computational effort.

1. INTRODUCTION

Latent class models (LCM) differ from standard regression models because they accommodate discrete latent variables. In layman terms, LCM assume that the heterogeneous observations in a sample arise from a number of homogenous subgroups (segments) mixed in unknown proportions. The main inferential goals of LCM are to identify the number of segments and simultaneously estimate the regression model parameters for each segment; and classify the individuals in their most likely segment. The characteristics of each segment can be deduced based on the demographic information of the members within each segment. In the past decade, LCM has increased in popularity, particularly in market segmentation, which is mainly due to technological advancements, rendering complex LCM computationally feasible, even on large data sets.

2. THEORETICAL FRAMEWORK

Latent class models assume that the population consists of S segments having unknown proportions $\pi_1, \pi_2, ..., \pi_S$. These proportions must satisfy the following two constraints:

$$\pi_s \geq 0 \ \forall s, \text{ and } \sum_{s=1}^{S} \pi_s = 1 \quad (1)$$

The conditional probability density function of the responses $\mathbf{Y}_i$, given that $\mathbf{Y}_i$ comes from segment s is given by:

$$\mathbf{Y}_i \sim f_{i|s}(y_i | \boldsymbol{\theta}_{si}, \varphi_s) \ , \quad (2)$$

where, the conditional density function is assumed to be a mixture of segment-specific densities, $f_{ik|s}(y_i | \theta_{sik}, \varphi_s)$. These component mixtures are assumed to be independent within the latent classes, such that:

$$f_{i|s}(y_i | \boldsymbol{\theta}_{si}, \varphi_s) = \prod_{k=1}^{K} f_{ik|s}(y_i | \theta_{sik}, \varphi_s) \quad (3)$$

If $\mathbf{Y}_i$ has conditional multivariate normal distribution then $f_{i|s}(y_i | \boldsymbol{\theta}_{si}, \varphi_s) = f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s)$ can be expressed as:

$$(2\pi)^{-K/2} |\boldsymbol{\Sigma}_s|^{-1/2} \exp\left[-\frac{1}{2} (y_i - \mathbf{X}_i \boldsymbol{\beta}_s)' \boldsymbol{\Sigma}_s^{-1} (y_i - \mathbf{X}_i \boldsymbol{\beta}_s) \right] \quad (4)$$

The unconditional probability density function of $\mathbf{Y}_i$, given the vector of unknown parameters $\boldsymbol{\Omega}' = (\boldsymbol{\pi}', \boldsymbol{\beta}', \boldsymbol{\Sigma})$, is:

$$f_i(y_i | \boldsymbol{\Omega}) = \sum_{s=1}^{S} \pi_s f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s) \quad (5)$$

The likelihood function $L(\boldsymbol{\Omega}, \mathbf{y}_i) = \prod_{i=1}^{N} f_i(y_i | \boldsymbol{\Omega})$ is given by:

$$\prod_{i=1}^{N} \left[\sum_{s=1}^{S} \pi_s f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s) \right] \quad (6)$$

The likelihood function, formulated by equation (6), is used to estimate the parameter vector $\boldsymbol{\Omega}$. The estimate $\hat{\boldsymbol{\Omega}}$, is obtained by using the maximum likelihood (ML) technique, in particular, through the use of the EM algorithm. Using Bayes' theorem, the posterior probability $\alpha_{is}(\mathbf{y}_i, \boldsymbol{\Omega})$ can be computed using the parameter estimates $\hat{\boldsymbol{\Omega}}$.

$$\alpha_{is}(\mathbf{y}_i, \boldsymbol{\Omega}) = \frac{\pi_s f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s)}{\sum_{s=1}^{S} \pi_s f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s)} \quad (7)$$

The procedure updates the parameter estimates iteratively, and when it ultimately converges, the posterior probabilities given by (7) will be used to assign each respondent to that segment with the largest posterior probability.

3. THE EM ALGORITHM

Dempster *et al.*, (1977) are credited with presenting the EM algorithm in its current form, where parameters of a mixture distribution are estimated by using the concept of incomplete data. The central idea behind the EM algorithm is to augment the data by including unobserved, referred to as missing, data, which comprises of unknown 0-1 indicators indicating whether a respondent belongs or not to a particular segment. Hence, instead of maximizing the likelihood via standard optimization methods, the expected complete-data log-likelihood function is maximized using the EM algorithm.

Let z_{is} be the unknown 0-1 indicator variables representing the unobserved data, which are assumed to be independent and identically multinomially distributed.

$$f(z_i, \pi) = \prod_{s=1}^{S} \pi_s^{z_{is}} \quad (8)$$

where, $\mathbf{z}_i = (z_{i1}, ..., z_{is})$ and $\mathbf{z} = (\mathbf{z}_1', ..., \mathbf{z}_s')$. Since z_{is} is considered as missing data, the complete-data likelihood function, $L_c(\boldsymbol{\Omega}, \mathbf{y}_i, \mathbf{z})$ is given by:

$$L_c(\boldsymbol{\Omega}, \mathbf{y}_i, \mathbf{z}) = \prod_{s=1}^{S} \prod_{i=1}^{N} \left(\pi_s f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s) \right)^{z_{is}} \quad (9)$$

The complete log-likelihood function $\log[L_c(\boldsymbol{\Omega}; \mathbf{y}_i, \mathbf{z})]$ is:

$$\sum_{s=1}^{S} \sum_{i=1}^{N} \left[z_{is} \log(\pi_{is}) + z_{is} \log\left(\pi_s f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s) \right) \right] \quad (10)$$

In the E-step, the expectation of the complete log-likelihood function, given by (10) is calculated with respect to the conditional distribution of the missing data, given both the observed data and the initial estimates of $\boldsymbol{\Omega}$. Since the complete-data log-likelihood function is linear in z_{is}, the expectation $E\left[\log\left(L_c(\boldsymbol{\Omega} | \mathbf{y}_i, \mathbf{z}) \right) \right]$ is obtained by replacing the z_{is} by their conditional expectation, given the observed data.

$$\sum_{s=1}^{S} \sum_{i=1}^{N} \left[E(z_{is} | \mathbf{y}_i, \boldsymbol{\Omega}) f_{i|s}(y_i | \mathbf{X}_i, \boldsymbol{\beta}_s \boldsymbol{\Sigma}_s) + E(z_{is} | \mathbf{y}_i, \boldsymbol{\Omega}) \log(\pi_s) \right]$$

where $E(z_{is} | \mathbf{y}_i, \boldsymbol{\Omega})$ is given by:

$$\frac{\hat{\pi}_s f_{i|s}(y_i | \mathbf{X}_i, \hat{\boldsymbol{\beta}}_s \hat{\boldsymbol{\Sigma}}_s)}{\sum_{s=1}^{S} \hat{\pi}_s f_{i|s}(y_i | \mathbf{X}_i, \hat{\boldsymbol{\beta}}_s \hat{\boldsymbol{\Sigma}}_s)} = \hat{\alpha}_{is} \quad \text{where} \quad \hat{\pi}_s = \frac{1}{N} \sum_{i=1}^{N} \hat{\alpha}_{is} \quad (11)$$

These posterior probabilities are updated iteratively by replacing the estimates of $\hat{\boldsymbol{\beta}}_s$ and $\hat{\boldsymbol{\Sigma}}_s$ obtained from the previous iteration.

4. APPLICATION

Two latent class models were fitted to identify factors that influence the customer choices when buying laptops and identify the product attributes that most influence the consumers in buying the product. In this application, the four selected laptop attributes included the brand (HP, Asus); the price (€500, €600, €700); RAM (4GB, 8GB) and the screen size (12 inch, 15 inch). This survey was designed and devised on Survey Monkey (an online survey questionnaire) where a number of laptop profiles having distinct attributes were generated and these profiles had to be assessed on a 7-point Likert scale where 1 corresponds to 'Not worthy' and 7 corresponds to 'Very worthy'. A rating scale was selected since it expresses the intensity of a preference better than a ranking scale. A full-profile method and full factorial design were chosen for the data collection method yielding a total of 24 distinct profiles. 69.8% of 262 participants who completed the online questionnaire were females, 74.4% were university students and 73.7% were less than 30 years. All participants owned a laptop. The first latent class model assume a 2-segment solution and the second assume a 3-segment solution. The parameter estimates of the two latent class models will be used in a simulation study, described in section 5, to investigate factors that affect the performance of the EM algorithm.

For the 2-segment solution, 175 (66.8%) respondents were allocated to segment 1 and the remaining 87 (33.2%) were allocated to segment 2. Respondents in both segments rated HP laptops more than Asus; rated cheaper laptops more than expensive ones; rated 4GB RAM laptops less than 8GB RAM; and rated 12 inch screen laptop less than 15 inch screen. However, participants in segment 2 are discriminating more between the brands, prices, screen sizes and random access memories compared to participants in segment 1. Table 1 displays the parameter estimates and standard errors for the 2-segment solution.

Table 1: Parameter estimates for the 2-segment solution

Parameter	Segment1		Segment2	
	Est.	S.E.	Est.	S.E.
Intercept	5.18	0.05	3.66	0.08
Brand (HP)	0.07	0.04	0.39	0.06
Brand (Asus)	0		0	
Price (€500)	0.37	0.05	0.56	0.08
Price (€600)	0.31	0.05	0.20	0.07
Price (€700)	0		0	
RAM (4GB)	-0.72	0.04	-0.82	0.06
RAM (8GB)	0		0	
Size (12inch)	-0.58	0.04	-1.29	0.06
Size (15inch)	0		0	

For the 3-segment solution, 117 (44.7%) respondents were allocated to segment 1, 24 (9.2%) respondents were allocated to segment 2 and the remaining 121 (46.1%) were allocated to segment 3. Respondents in both segments rated HP laptops more than Asus; rated cheaper laptops more than expensive ones; rated 4GB RAM laptops less than 8GB RAM; and rated 12 inch screen laptop less than 15 inch screen. However, participants in segment 1 are discriminating more between the prices and screen sizes; participants in segment 2 are discerning more between the brands; and participants in segment 3 are discriminating more between the random access memories. Table 2 displays the parameter estimates and standard errors for the 3-segment solution.

Table 2: Parameter estimates for the 3-segment solution

Parameter	Segment1		Segment2		Segment3	
	Est.	S.E.	Est.	S.E.	Est.	S.E.
Intercept	4.47	0.03	2.37	0.07	5.32	0.03
Brand (HP)	0.19	0.03	0.46	0.06	0.11	0.03
Brand (Asus)	0		0		0	
Price (€500)	0.51	0.03	0.08	0.07	0.43	0.03
Price (€600)	0.28	0.03	-0.04	0.07	0.32	0.03
Price (€700)	0		0		0	
RAM (4GB)	-0.53	0.03	-0.63	0.06	-0.91	0.03
RAM (8GB)	0		0		0	
Size (12inch)	-1.27	0.03	-0.60	0.06	-0.43	0.03
Size (15inch)	0		0		0	

Table 3: BIC value for the 2- segment and 3-segment solutions

Number of segments *S*	Deviance (-2 log *L*)	Number of parameters *P*	BIC
2	10868	12	21858
3	10545	18	21273

Table 3 displays the deviances, number of parameters and BIC values of the two-segment and three-segment solutions. Figures 1 to 4 provide graphical displays of the mean rating scores grouped by segment and laptop attributes. Respondents in segments 1 and 2 are price sensitive but not brand sensitive, while respondents in segment 3 are brand sensitive but not price sensitive. Respondents in all three segments prefer 8GB RAM and 15 inch screen laptops more than 4GB RAM and 12 inch screen laptops.

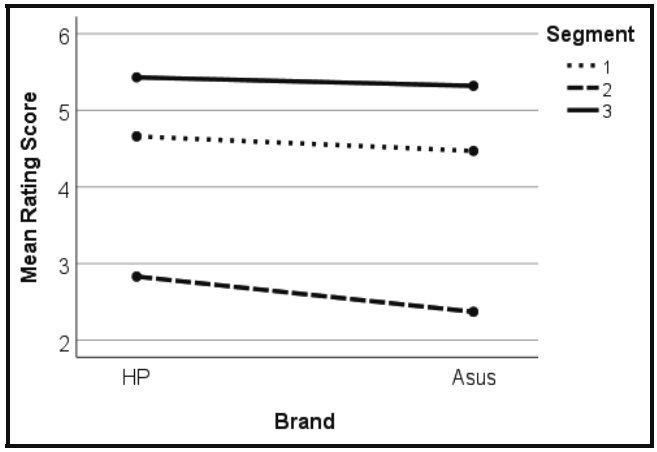


Figure 1: Mean rating scores grouped by segment and brand

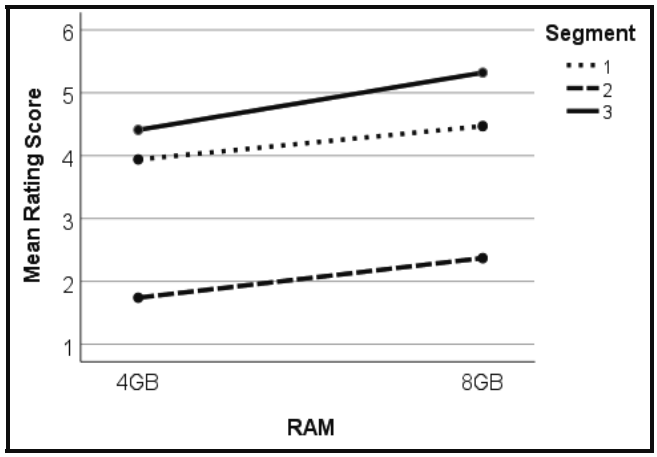


Figure 2: Mean rating scores grouped by segment and RAM

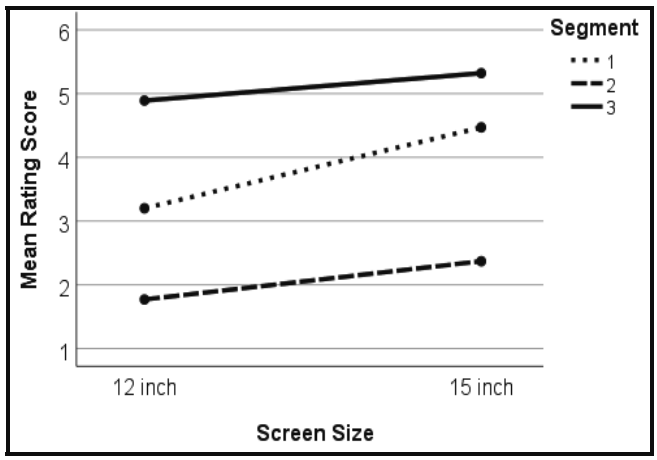


Figure 3: Mean rating scores grouped by segment and size

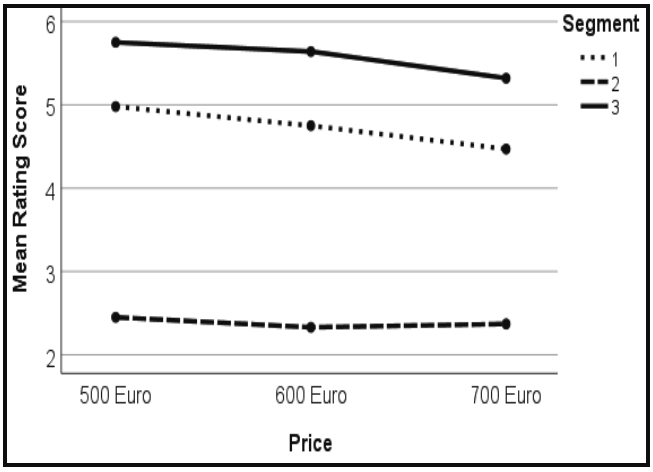


Figure 4: Mean rating scores grouped by segment and price

5. MONTE CARLO SIMULATION

A further task was to examine the performance of latent class models by modifying a number of factors. Three of the factors that are highlighted in literature as having potential effect on model performance include:

- Number of simulated respondents
- Number of segments
- Size of perturbation parameter σ_i^2 of the error terms.

The above three factors reflect a variation in conditions in many applications which are expected to affect the performance of the model fit. The design used in the study was 3 x 2^3 full factorial design, which yielded 24 observations. The following four measures are normally used to assess computational effort, parameter recovery, predictive power, goodness of fit and segment membership recovery. The root-mean-squared error between the true and estimated parameters is a measure of parameter recovery. β_p and $\hat{\beta}_p$ are the true and estimated parameters, where *P* is the number of parameters.

$$RMS\left(\hat{\boldsymbol{\beta}}\right) = \left[\sum_{p=1}^{P} \frac{\left(\beta_p - \hat{\beta}_p\right)^2}{P}\right]^{\frac{1}{2}} \tag{13}$$

The root-mean-squared error between the true and estimated segment membership probabilities is a measure of segment proportion recovery. π_s and $\hat{\pi}_s$ are the true and estimated segment membership probabilities, where *S* is the number of segments.

$$RMS\left(\hat{\boldsymbol{\pi}}\right) = \left[\sum_{s=1}^{S} \frac{\left(\pi_s - \hat{\pi}_s\right)^2}{S}\right]^{\frac{1}{2}} \tag{14}$$

The root-mean-squared-error between the true and predicted responses is a measure of the predictive power. y_{ik} and y_{ik} are the true and estimated responses, where *N* and *K* are the number of hypothetical subjects and the number of profiles assessed by each subject.

$$RMS\left(y\right) = \left[\sum_{i=1}^{N}\sum_{k=1}^{K} \frac{\left(y_{ik} - y_{ik}\right)^2}{N.K}\right]^{\frac{1}{2}} \tag{15}$$

In order to assess the factors that affect the performance of latent class models, synthetic data sets were generated, where the simulation was devised to mimic the laptop application. To allocate hypothetical subjects to segments the proportion π_s of members in each segment was specified, satisfying the constraint these proportions sum to 1. This was carried out by first generating *N* uniformly distributed pseudo-random real values in the range [0,1] and then by computing the cumulative probabilities $q_s = \sum_{j=1}^{s} \pi_j$. Every subject whose corresponding value was in the range (q_{s-1}, q_s) was allocated to segment *s*. This gives a random segment allocation to each hypothetical subject. To simulate the subjects' rating responses, the linear predictors and the corresponding parameters $\boldsymbol{\beta}_k$ were specified for the *S* segments. Moreover, the design and the linear predictor were set the same as in the application. Given the segment allocation of each member, synthetic data values were generated for each subject. These values were then perturbed by adding an error term having a normal distribution. Six specified cut-points α_r were used to convert these values to rates ranging from 1 to 7. Values in the range (α_{r-1}, α_r) were converted to rate *r*. This gives a random rating category allocation to each profile by each hypothetical subject.

The number of simulated respondents was varied at three levels (100, 300 and 500). It is expected that a greater number of simulated subjects improve the precision of the estimated segment-level parameters. The number of segments was also varied at two levels (2 and 3 segments) because these represent the range of segments commonly found in segmentation applications. It is expected that a greater number of segments deteriorate the precision of the estimated segment-level coefficients as a greater number of model parameters have to be estimated. The error terms were assumed to be normally distributed and the parameter σ_i^2 was set to 0.1, 0.5 and 1. It is expected that a larger perturbation value reduces the precision of the estimated segment-level parameters since there will be less cohesion in each segment and lower segment separation.

Table 4: Parameter recovery using simulated data

Number of subjects	Perturbation value	Number of segments	$RMS(\hat{\boldsymbol{\beta}})$
100	0.1	2	0.2402
300			0.2396
500			0.2376
100	0.5		0.2716
300			0.2545
500			0.2606
100	1.0		0.3270
300			0.3138
500			0.3327
100	0.1	3	0.2427
300			0.2401
500			0.2341
100	0.5		0.3940
300			0.3674
500			0.3276
100	1.0		0.4049
300			0.3679
500			0.3490

Table 5: Segment proportion recovery using simulated data

Number of subjects	Perturbation value	Number of segments	$RMS(\hat{\boldsymbol{\pi}})$
100	0.1	2	0.0475
300			0.0260
500			0.0249
100	0.5		0.0485
300			0.0260
500			0.0246
100	1.0		0.0721
300			0.0607
500			0.0547
100	0.1	3	0.0638
300			0.0288
500			0.0257
100	0.5		0.0638
300			0.0295
500			0.0276
100	1.0		0.0780
300			0.0689
500			0.0557

Ten data sets were generated for each factor level combination according to the number of subjects, number of segments and the perturbation value. Each simulated data set was re-fitted using a latent class model.

Table 6: Assessing predictive power using simulated data

Number of subjects	Perturbation value	Number of segments	$RMS(y)$
100	0.1	2	1.3920
300			1.3945
500			1.3952
100	0.5		1.4364
300			1.4420
500			1.5088
100	1.0		1.4833
300			1.4195
500			1.5260
100	0.1	3	1.5620
300			1.7948
500			1.4629
100	0.5		1.5634
300			1.5828
500			1.4799
100	1.0		1.5633
300			1.5931
500			1.5061

The $RMS(\hat{\boldsymbol{\beta}})$ $RMS(\hat{\boldsymbol{\pi}})$ and $RMS(\hat{y})$ values shown in tables 4, 5 and 6 were computed after permuting the parameters and predicted responses to match estimated and true segments optimally. All the three measures were averaged over these ten data sets.

Table7: Segment membership recovery using simulated data

Number of subjects	Perturbation value	Number of segments	Segment membership recovery
100	0.1	2	100%
300			100%
500			100%
100	0.5		100%
300			99.86%
500			99.46%
100	1.0		98.80%
300			97.92%
500			96.42%
100	0.1	3	100%
300			100%
500			99.94%
100	0.5		99.92%
300			97.67%
500			97.96%
100	1.0		93.80%
300			97.20%
500			91.36%

The percentage number of subjects that are correctly classified into their true segments is a measure of segment membership recovery. Table 7 displays the percentage number of subjects, averaged over the ten data sets, which are correctly classified into their true segments. It should be noted that after assigning each hypothetical subject to a segment with highest posterior probability these segments were permuted to maximize match with the true segments.

6 CONCLUSIONS

In general, the percentage of correctly classified hypothetical subjects in their true segment improves with a decrease in the number of segments and a reduction in the perturbation value; however, it is unaffected by changes in sample size. Parameter recovery improves with a decrease in the perturbation value, a decrease in the number of segments, and an increase in the sample size. Predictive power improves with a decrease in the perturbation value, however it is unaffected by changes in the number of segments or sample size. Segment proportion recovery improves with an increase in sample size, a decrease in the number of segments and a decrease in the perturbation value. The results corroborate with the findings of Camilleri and Portelli (2007); Wedel and DeSarbo (1995); and Vriens, Wedel and Wilms (1996).

REFERENCES

Camilleri, L., Portelli, M. (2007). Segmenting the heterogeneity of tourist preferences using a Latent Class model combined with the EM algorithm, *Proceedings of the 6th APLIMAT International Conference, Bratislava*. 343-356.

Dempster, A.P., Laird, N.M. Rubin, D.B. (1977). Maximum Likelihood from Incomplete Data via the EM algorithm, *Journal of the Royal Statistical Society*, B, 39, 1-38.

Vriens, M., Wedel, M. and Wilms, T. (1996), Metric Conjoint Segmentation Methods A Monte Carlo Comparison, *Journal of Marketing Research*, 23, 73-85.

Wedel, M., DeSarbo, W.S. (1995), A Mixture Likelihood Approach for Generalized Linear Models, *Journal of Classification*, 12, 1-35.

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, Item Response models, Generalized Estimation Equations models, Multilevel models, Structural Equations models and Survival models. He is presently an associate professor in the department of Statistics and Operations Research at the University of Malta.

FOUR AND EIGHT COMMODITY ECONOMIC EQUILIBRIUM PROBLEMS WITH EXTENSIVE SUBSTITUTION POSSIBILITIES

William Conley
Austin E. Cofrin School of Business
University of Wisconsin at Green Bay
Green Bay, Wisconsin 54311-7001
U.S.A.
Conleyw@uwgb.edu

KEYWORDS
Nonlinear equations systems, economics, prices, statistical optimization, Monte Carlo methods

ABSTRACT

Economists generally agree that in the vast majority of cases as the price of a commodity increases, more of it will be produced or mined or made. Also, it follows that as the price of it decreases, less of the commodity will be supplied or manufactured. Some treatments of finding the equilibrium price for the commodity assume linear supply and demand equations with no substitutes available for them. However, in the real world of economics and big business there could be (and generally are) substitutes for almost all commodities and their supply and demand equations are nonlinear. This makes solving them with theoretical mathematics very difficult if all the commodity supply and demand equations are dealt with simultaneously in one large system. Therefore, here are presented two such examples of economic equilibrium equations (EEE) which are solved with multi stage Monte Carlo optimization (MSMCO).

INTRODUCTION

Monte Carlo optimization generally means taking a fairly large random sample (5,000, 50,000 or 100,000 for example) of feasible solutions of an optimization problem with a computer program and printing out the best answer found.

Even though this can be very useful in many applications, theoretical mathematicians pointed out that it would be quite limited as the number of variables increases. However (Conley 1981), views this as merely stage one to be followed by several more stages following the trail of better and better answers to the optimal solution or a useful approximation. The ever increasing speed, capacity and ubiquitous nature of computers in our 21st Century has made this multi stage Monte Carlo optimization simulation approach more viable.

Figure 1 gives a partial geometric and statistical representation of this statistical optimization simulation approach.

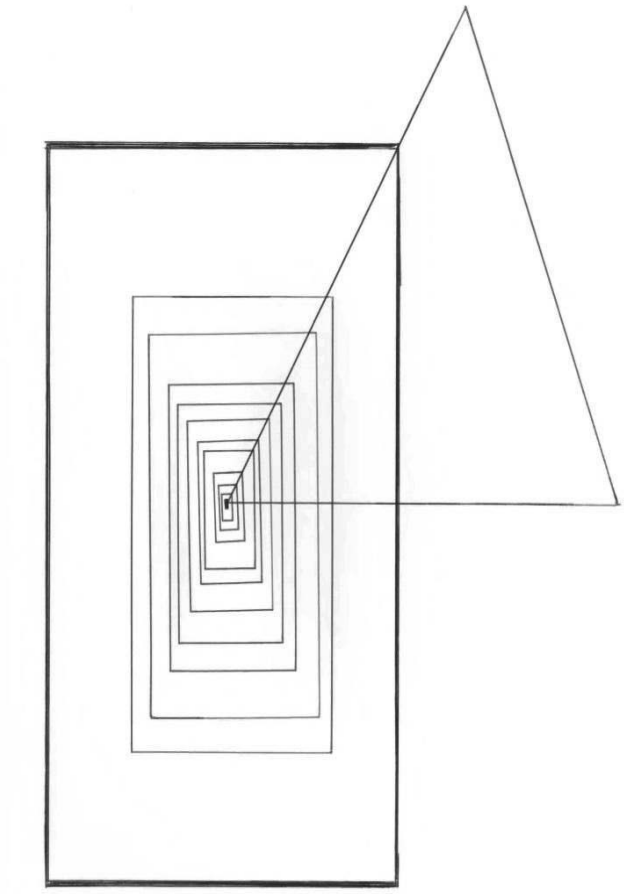


Figure 1. A Twelve Stage Statistical Optimization (MSMCO) Search for the Minimum Error Solution

Let us look at two examples. Please note that Figure 2 represents the forces of supply and demand for just one commodity, where X is the price and the star is the equilibrium point.

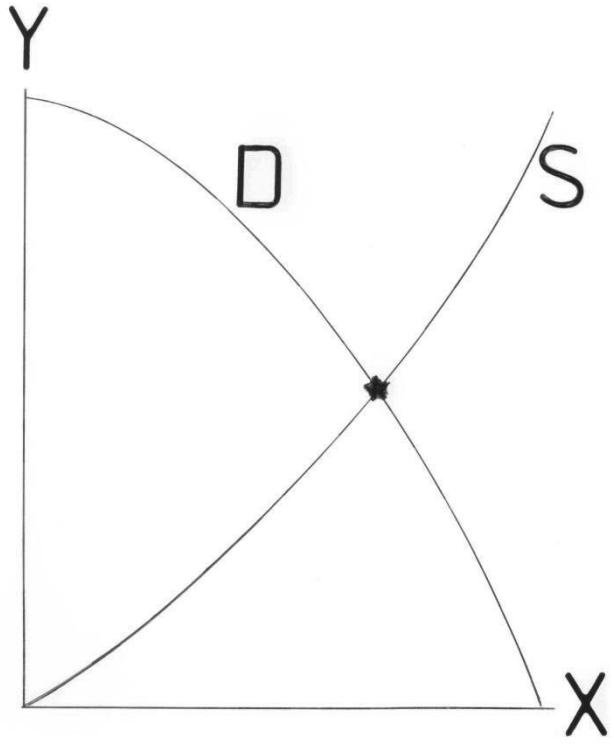


Figure 2. Nonlinear Supply and Demand Equations. Star is Equilibrium Point. X is Price and Y is Commodity Amount.

A FOUR COMMODITY EEE PROBLEM

Four commodities that are all substitutes for each other are studied by economists and business people with the desire to find out what four prices will bring the whole system into equilibrium if possible. The results of their studies yield the following eight equations: D_1 and S_1 are for demand and supply of commodity one. Additionally, D_2 and S_2 followed by D_3 and S_3 and D_4 and S_4 represent the demand and supply equations for commodities 2, 3, and 4. Note that the x_i's for i=1,2,3 and 4 are the commodity prices in EUROS per ton and D_i's and S_i's are the resulting supplies and demands in units of tons.

$$D_1 = 521{,}548 - 12X_1 - 4X_2 - 18X_3 - 35X_4 - 19X_1^{2}X_2^{5}X_3^{2} - X_1^{3}X_2^{6}X_4^{1} - 2X_1^{5}X_3^{1}X_4^{3} - 17X_2^{4}X_3^{2}X_4^{4} - 34X_1^{3}X_2^{1}X_3^{1}X_4^{4} \quad (1)$$

$$S_1 = 259{,}309 + 11X_1 + 31X_2 + 25X_3 + 15X_4 + 38X_1^{3}X_2^{4}X_3^{5} + 11X_1^{2}X_2^{7}X_4^{1} + 27X_1^{2}X_3^{2}X_4^{8} + 30X_2^{5}X_3^{2}X_4^{3} + 40X_1^{2}X_2^{4}X_3^{1}X_4^{3} \quad (2)$$

$$D_2 = 428{,}992 - 15X_1 - 17X_2 - 9X_3 - 36X_4 - 40X_1^{3}X_2^{4}X_3^{1} - 16X_1^{6}X_2^{1}X_4^{2} - 8X_1^{1}X_3^{8}X_4^{2} - 12X_2^{3}X_3^{3}X_4^{5} - 26X_1^{2}X_2^{3}X_3^{3}X_4^{2} \quad (3)$$

$$S_2 = 225{,}000 + 41X_1 + 19X_2 + 15X_3 + 35X_4 + 27X_1^{1}X_2^{1}X_3^{7} + 50X_1^{2}X_2^{2}X_4^{2} + 13X_1^{5}X_3^{2}X_4^{2} + 20X_2^{7}X_3^{1}X_4^{2} + 49X_1^{5}X_2^{1}X_3^{3}X_4^{2} \quad (4)$$

$$D_3 = 622{,}972 - 23X_1 - 16X_2 - 39X_3 - 24X_4 - 15X_1^{9}X_2^{1}X_3^{1} - 60X_1^{2}X_2^{2}X_4^{2} - 38X_1^{4}X_3^{1}X_4^{2} - 21X_2^{4}X_3^{1}X_4^{3} - 20X_1^{3}X_2^{3}X_3^{3}X_4^{2} \quad (5)$$

$$S_3 = 275{,}000 + 53X_1 + 36X_2 + 29X_3 + 45X_4 + 8X_1^{5}X_2^{4}X_3^{6} + 13X_1^{6}X_2^{3}X_4^{2} + 30X_1^{1}X_3^{5}X_4^{2} + 7X_2^{5}X_3^{5}X_4^{4} + 29X_1^{2}X_2^{3}X_3^{3}X_4^{3} \quad (6)$$

$$D_4 = 451{,}702 - 11X_1 - 42X_2 - 14X_3 - 28X_4 - 34X_1^{2}X_2^{2}X_3^{3} - 13X_1^{5}X_2^{4}X_4^{3} - 7X_1^{4}X_3^{6}X_4^{5} - 36X_2^{2}X_3^{3}X_4^{1} - 18X_1^{3}X_2^{4}X_3^{1}X_4^{1} \quad (7)$$

$$S_4 = 195{,}000 + 34X_1 + 27X_2 + 54X_3 + 37X_4 + 5X_1^{3}X_2^{2}X_3^{4} + 20X_1^{4}X_2^{1}X_4^{3} + 16X_1^{5}X_3^{3}X_4^{1} + 30X_2^{3}X_3^{2}X_4^{3} + 8X_1^{3}X_2^{3}X_3^{3}X_4^{1} \quad (8)$$

Therefore, the economists seek to minimize $f(X_1, X_2, X_3, X_4) = |D_1-S_1| + |D_2-S_2| + |D_3-S_3| + |D_4-S_4|$ subject to $0 \le X_i \le 1000$ Euros for i=1, 2, 3, 4 and accurate to two decimal places using a 40 stage multi stage Monte Carlo optimization (MSMCO) simulation that draws 100,000 feasible solutions at each stage. The printout of the solution follows here.

$X_1 = 382.15$, $X_2 = 499.18$, $X_3 = 574.32$, $X_4 = 580.99$
$e_1 = 0.00000$, $e_2 = 0.00000$, $e_3 = 0.00000$, $e_4 = 0.00000$
$d_1=s_1 = 460{,}017.03$, $d_2=s_2=345{,}408.09$,
$d_3=s_3 = 531{,}887.81$, $d_4=s_4 = 290{,}530.66$

Stage Number	Total Equations Error
1	3920.81250
2	3920.81250
3	3599.75000
4	2943.43750
5	2577.43750
6	1897.62500
7	497.46875
8	497.46875
9	444.37500
10	444.37500
11	210.59375
12	1 6 2.156 25
13	120.15625
14	71.37500
15	69.59375
16	60.43750
17	46.68750

18	20.09375
19	14.68750
20	6.09375
21	5.40625
22	5.40625
23	5.40625
24	2.00000
25	2.00000
26	1.53125
27	0.93750
28	0.53125
29	0.37500
30	0.37500
31	0.21875
32	0.15625
33	0.12500
34	0.12500
35	0.03125
36	0.00000
37	0.00000
38	0.00000
39	0.00000
40	0.00000

DISCUSSION

Notice that after stage one of the 40 stage simulation the total equation error $e_1 + e_2 + e_3 + e_4$ of the four commodities in one system is about 3920. Then it starts dropping in stage 2 and so on until in stage 10 it is 444 and stage 20 about 6 and by stage 27 it is less than one (.93750). Toward the end of the simulation (stages 36-40) the total error is 0.00000 which is certainly accurate enough for this type of problem.

However, if more accuracy were required in an engineering problem, switching to double precision arithmetic (14 places of accuracy) in one's computer language could still be handled with multi stage Monte Carlo optimization (MSMCO). Also, MSMCO) still works in complex variables (Conley 1993) arithmetic ($a+b_i$, etc). However, in these latter two cases a little more computer time would be required.

Figure 1 shows a two-dimensional representation of an eleven stage Monte Carlo simulation closing in on and surrounding and finding the true minimum solutions of an optimization problem whose feasible solution distribution is represented by the unsymmetric distribution drawn in Figure 1. The ever decreasing in size and ever moving shapes (driven by thousands of sample answers) learn from its sampling where the better and better answers are in the feasible solution space. Then they close in on that region and surround the optional solution and then find it (or a useful approximation in very big problems). Figure 3 gives a three-dimensional representation of the MSMCO process.

This technique becomes more difficult to apply as the number of variables increase, but it can still work. (Conley 2008) and (Conley 2010) demonstrated its usefulness on a variety of optimization problems.

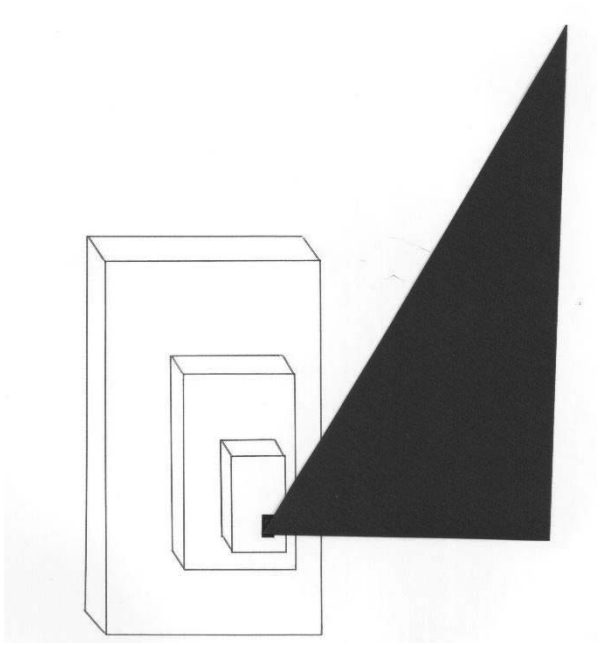


Figure 3. Statistical Optimization Closing on the Solution

Let us now look at an eight commodity economic equilibrium equation setting problem and try multi stage Monte Carlo optimization (MSMCO) or statistical optimization) simulation on it and see what happens.

AN EIGHT COMMODITY PROBLEM

An industry's statisticians and economists have developed the following sixteen demand (eight) and supply (eight) equations for eight commodities that are all intertwined in the competitive marketplace. Any and all of them can be substitutes for each other as market forces react to their unit prices.

For these sixteen following equations the x_i's are the prices for commodities 1, 2, 3, 4, 5, 6, 7, and 8 respectively, while the D_is and S_is are the resulting demands and supplies for the commodities in units of tons for i=1, 2, 3, 4, 5, 6, 7 and 8 respectively.

$$D_1 = 1014158 - 9X_1^{1.07} - 11X_6^{.89} - 14(X_1X_2X_3X_4X_5X_6X_7X_8)^{.15}$$

$$D_2 = 849273 - 12X_2^{1.05} - 7X_3^{1.15} - 5(X_1X_2X_3X_4X_5X_6X_7X_8)^{.14}$$

$$D_3 = 1207151 - 4X_3^{.97} - 10X_8^{1.11} - 9(X_1X_2X_3X_4X_5X_6X_7X_8)^{.12}$$

$$D_4 = 903215 - 21X_4^{1.09} - 15X_2^{1.08} - 18(X_1X_2X_3X_4X_5X_6X_7X_8)^{.11}$$
$$D_5 = 853209 - 14X_5^{.95} - 12X_7^{1.03} - 13(X_1X_2X_3X_4X_5X_6X_7X_8)^{.13}$$
$$D_6 = 1110034 - 19X_6^{.99} - 14X_1^{.98} - 9(X_1X_2X_3X_4X_5X_6X_7X_8)^{.16}$$
$$D_7 = 853290 - 15X_7^{1.13} - 10X_4^{1.07} - 11(X_1X_2X_3X_4X_5X_6X_7X_8)^{.12}$$
$$D_8 = 799468 - 11X_8^{1.14} - 9X_5^{1.05} - 6(X_1X_2X_3X_4X_5X_6X_7X_8)^{.15}$$

$$S_1 = 916576 + 10X_1^{1.03} + 7X_7^{1.06} + 11(X_1X_2X_3X_4X_5X_6X_7X_8)^{.17}$$
$$S_2 = 810627 + 9X_2^{1.11} + 3X_5^{.95} + 8(X_1X_2X_3X_4X_5X_6X_7X_8)^{.11}$$
$$S_3 = 1179086 + 14X_3^{1.1} + 8X_4^{.88} + 16(X_1X_2X_3X_4X_5X_6X_7X_8)^{.10}$$
$$S_4 = 843613 + 19X_4^{1.03} + 12X_6^{1.05} + 8(X_1X_2X_3X_4X_5X_6X_7X_8)^{.14}$$
$$S_5 = 817213 + 10X_5^{1.05} + 10X_3^{1.08} + 12(X_1X_2X_3X_4X_5X_6X_7X_8)^{.14}$$
$$S_6 = 1036682 + 16X_6^{1.07} + 9X_8^{1.07} + 7(X_1X_2X_3X_4X_5X_6X_7X_8)^{.15}$$
$$S_7 = 789350 + 11X_7^{.90} + 17X_1^{.97} + 9(X_1X_2X_3X_4X_5X_6X_7X_8)^{.16}$$
$$S_8 = 674033 + 21X_8^{1.09} + 11X_2^{1.11} + 8(X_1X_2X_3X_4X_5X_6X_7X_8)^{.18}$$

The question is which combinations of prices for $1 \leq x_i \leq$ 1000 dollars (in dollars and cents + 2 decimal places) will bring the whole system of the eight commodities into equilibrium where

$D_1 = S_1$
$D_2 = S_2$
$D_3 = S_3$
$D_4 = S_4$
$D_5 = S_5$
$D_6 = S_6$
$D_7 = S_7$
$D_8 = S_8$

Therefore, we transform this system of equations to minimize $f(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8) = |D_1-S_1| + |D_2-S_2| + |D_3-S_3| + |D_4-S_4| + |D_5-S_5| + |D_6-S_6| + |D_7-S_7| + |D_8-S_8|$ subject to $1 \leq X_i \leq 1000$ dollars and cents for i = 1, 2, . . . 8.

It is then solved with a statistical optimization simulation (also called multi stage Monte Carlo optimization or MSMCO) of 45 ever decreasing in size) stages drawing 100000 feasible solutions at each stage and always following the trail of better and better (less error) answers to an optimal solution of total equilibrium of:

X_1 = \$813.23, X_2 = \$758.10, X_3 = \$291.48, X_4 = \$357.30, X_5 = \$203.58, X_6 = \$475.84, X_7 = \$509.25, X_8 = \$641.66 commodity prices
$e_1 = 0.0$, $e_2 = 0.00$, $e_3 = 0.00$, $e_4 = 0.00$, $e_5 = 0.0$, $e_6 = 0.00$, $e_7 = 0.00$, $e_8 = 0.00$ tons or no error at all (all errors in units of tons)

$S_1=D_1$=977806.31 $S_2=D_2$=827010.50 $S_3=D_3$=1189857.88 $S_4=D_4$=867177.69 $S_5=D_5$=836001.00 $S_6=D_6$=1068481.00 $S_7=D_7$=826756.19 $S_8=D_8$=770204.44 tons (all amounts in tons)

Therefore, the industry executives can now be told by the economists and statisticians involved in this study that commodity prices of 813.23, 758.10, 291.48, 357.30, 203.58, 475.84, 509.25 and 641.66 in dollars and cents will lead to demands and supplies of 977806.31 tons, 827010.50 tons, 1189857.88 tons, 867177.69 tons, 836001.00 tons, 1068481.00 tons, 826756.19 tons, and 770204.44 tons of commodities 1, 2, 3, 4, 5, 6, 7, and 8 respectively. This could be useful information that the managers and decision makers in companies in this industry can act on.

DISCUSSION

Presented here were hypothetical four and eight commodity nonlinear economic equilibrium systems driven by the four and eight prices for the commodities (or products) that are all substitutes for each other. With all eight prices between 1 and 1000 dollars and two decimal places of accuracy for the cents, we have 100000 x 10^8 = $1.x10^{40}$ possible feasible solutions to search through. That would take trillions of years of computer time (on the world's fastest computer). However, we use the survey sampling (market research approach) of drawing 100000 sample answers not to solve the problem but to locate in eight dimensional space where the best answers lowest total errors in this case are. Then, in stage two we read over these and close in with another 100000 sample feasible solutions to even better answers. Then in stages 3, 4, 5, . . . 45 we draw 100000 more answers closing in using only 45 x 100000 = 4,500,000 feasible solutions out of the total of $1x10^{40}$ feasible solutions. This took a few minutes of computer run time on the author's desk top computer in his office. The eight commodity simulation took a little longer to run than the four commodity one.

Therefore, survey sampling works to estimate market share and customers preferences, wants and needs for products. But it also works to solve complex multivariate optimization problems. (Anderson 2003), (Anderson, Sweeney and Williams 1999), (Black 2014), (Cochran 1977), (Hayter, 2002) and (Keller and Warrack 2003) provide good reviews of forecasting, curve fitting and survey sampling, while (Samuelson and Nordhaus 2009) and (Samuelson and Nordhaus 2010) give important economics information to better understand economic equilibrium equations.

The power of statistical optimization simulation can be applied to many fields in our computer age as the 21st

century and its computer developments advance. It is an exciting and useful time for computer simulation in general.

CONCLUSION

Presented here were two hypothetical nonlinear economic equilibrium systems of equations which were solved for the equilibrium prices and subsequently the equilibrium amounts that these prices would make available. Good reviews of statistics which may be needed to produce economic equilibrium equations in a real world problem can be found in (Anderson 2009) (Anderson 1999), (Hayter 2002), (Black 2014), (Keller and Warrack 2003), (Triola 2003) and (Weiers 2002).

The general purpose nature of the featured MSMCO simulation technique can be shown in (Conley 2008) and (Conley 2010) for examples.

REFERENCES

Anderson, T. W. 2003. *Multivariate Statistical Analysis*, 3rd edition, Wiley and Sons, New York.

Anderson, D. R., Sweeney, D. J., Williams, T. A. 1999. *Statistics for Business and Economics,* 7th edition. Southwestern College Publishing, Cincinnati, Ohio.

Black, K. 2014. *Business Statistics for Contemporary Decision Making*, 8th edition. John Wiley & Sons, New York.

Conley, W. C. 1981. *Optimization: A Simplified Approach*. Petrocelli Books, Princeton and New York.

Conley, W. C. 1993. "Simulation applied to a degree three hundred polynomial." *International Journal of Systems Science*, Vol. 24, No. 5, pp 819-828.

Conley, W. 2008. "Ecological optimization of pollution control equipment and planning from a simulation perspective." *International Journal of Systems Science*, Vol. 1, pp. 1-7.

Conley, W. C. 2010. "We can save our world if worlds collide." In *2010 Proceedings of Industrial Simulation Conference* ISC2010 (Budapest, Hungary, June 7-9) EUROSIS-ETI, Belgium, 45-49.

Hayter, A. J. 2002. *Probability and Statistics for Engineers and Scientists*, 2nd Edition, Duxbury Press, Pacific Grove, California.

Keller, G. and Warrack, B. 2003. *Statistics for Management and Economics*. 6th Edition. Thompson Brooks/Cole, Pacific Grove, California.

Samuelson, P. and Nordhaus, R. 2009. *Macroeconomics*, 19th edition. Irwin McGraw Hill, New York.

Samuelson, P. and Nordhaus, R. 2010. *Economics*. 19th International Edition. McGraw-Hill, New York.

Triola, M. 2003. *Elementary Statistics*. 9th Edition. Addison Wesley, Pearson Education, Boston.

Weiers, R. 2002. *Introduction to Business Statistics*. 4th Edition. Duxbury Thompson Learning Wordworth Group. Belmont, California.

BIOGRAPHY

WILLIAM CONLEY received a B.A. in mathematics (with honors) from Albion College in 1970, an M.A. in theoretical mathematics from Western Michigan University in 1971, a 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 five books and more than 230 publications world-wide. He also gave a speech and presentation on his research at the 7th International Genetic Algorithms Conference held in the Unites States in July 1997. 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. He was also named to Omicron Delta Epsilon (ODE) the national economics honorary and Kappa Mu Epsilon (KME) the national mathematics honorary.

DATAMINING FORECASTING AND DATA CONSTRUCTS

A SMART APPROACH TO HARVEST DATE FORECASTING

Beatriz Charneca, Vanda Santos, Ana Crespo
Departamento de Química, Escola de Ciências e Tecnologia
Universidade de Évora
Évora, Portugal
E-mail: bibas-c@hotmail.com,
{vanda.sofiasantos, anacfcrespo}@gmail.com

Henrique Vicente
Departamento de Química, Escola de Ciências e Tecnologia
Centro de Química de Évora
Universidade de Évora
Évora, Portugal
E-mail: hvicente@uevora.pt

Humberto Chaves
Escola Superior Agrária de Beja
Instituto Politécnico de Beja
Beja
Portugal
E-mail: hc@ipbeja.pt

Jorge Ribeiro, Victor Alves
Escola Superior de Tecnologia e Gestão
ARC4DigiT – Applied Research Center for Digital Transformation – Instituto Politécnico de Viana do Castelo
Viana do Castelo, Portugal
E-mail: jribeiro@estg.ipvc.pt, victoralves.at.vnc@gmail.com

José Neves
Centro Algoritmi
Universidade do Minho
Braga
Portugal
E-mail: jneves@di.uminho.pt

KEYWORDS
Analysis of the Must, Date of the Harvest, Knowledge Discovery in Databases, Data Mining, Decision Trees.

ABSTRACT

The concept of grape ripeness depends not only on the degree of enrichment of the chemical compounds in the grape and the volume of the berries, but also on the possible production purposes. The different types of maturation in individual cases are not sufficient for the decision on the harvest date. Taken together, however, they define oenological maturation times and help to harvest them. However, there are no consistent studies that correlate the chemical parameters obtained from must analysis and oenological maturation due to the non-linearity of these two types of variables. Therefore, this work seeks to create a self-explanatory model that allows for the prediction of ideal harvest time, based on eneological parameters related to practices in new developments in knowledge acquisition and management in relational databases.

INTRODUCTION

The grape's maturation is of greater importance as it influences the type and quality of the wine. The maturation process combines physiological and biochemical mutations, which together with the grape varieties and climatic/geophysical conditions (e.g. light, availability of soil water) make up the difference in the final product (Blouin and Peynaud 2012). The berry's growth is caused by the substances it produces, and mainly by the vine-producing substances that migrate to the berry. In addition, this growth is influenced by the number of seeds, which are responsible for cell proliferation and the formation of grape pulp (Magalhães 2015). In general, there are considered four stages of development of the berry, which are the herbaceous phase, the painter, the maturing and the overmaturing.The phase that corresponds to the herbaceous growth occurs from the grape revenge state to the painter. This period lasts 45 to 65 days. At this stage, the berries have a very pronounced metabolism and development, with chlorophyll being the more obvious pigmentation. There is also an increase in the acid reserve and low sugar contents (Ribéreau-Gayon et al. 2006).

According to Ribéreau-Gayon et al. (2006), the ripening of the grapes begins in the painter phase and should continue until the berries are ripe. This phase will have a periodicity of about 35 to 55 days. During the painter's time the growth of the vine slows down as the berry gains in size and begins to paint. The process of color change can take place from one day to the next, which in the face of the sudden change is called an abrupt phenomenon. In this phase, a very intense enzymatic activity is initiated, which leads to the storage of free sugars, potassium, amino acids and phenolic compounds (Ribéreau-Gayon et al. 2006). If ripening is considered sufficient, it should be stopped because, from that moment, the process cause the grapes to lose qualities. However, it is difficult to determine the exact point in time at which ripening is achieved, once there is only the perception that the berries are ripe. This happens due to the fact that the maturation phase depends on several factors, namely the type of wine to be produced, the grape variety, the degree of earliness, the temperature and the type of soil (Magalhães 2015, Toda 2011).

Some authors defend the existence of four different types of maturation, i.e,, industrial, aromatic, phenolic, and physiological maturation (Blouin and Guimberteau 2000, Ribéreau-Gayon et al. 2011). The first takes into account the maximum concentration of sugars or the minimum concentration of total acidity. It also takes into account the maximum amount of sugar per hectare, the likely alcoholic strength and the final destination of the grapes. Aromatic ripening tends to produce pleasant and desirable aromatic compounds, so it is difficult to determine. Normally, the berries are tested to confirm that they have the desired aromatic composition. In contrast, phenolic ripening reflects the presence of phenolic compounds, namely anthocyanins and tannins, and evaluates their polymeric structures and origins. Finally, the physiological ripening is achieved when the seeds are suitable for carrying out the germination process. These four types of maturation can be aggregated in a type of maturation to which the denomination of oenological maturation is given, considering the optimum harvest time, since it adds the maturation process of all constituents of the berry (i.e., pulp, cranberry). It is defined by the oenologist and

is usually used for a more accurate analysis of the grapes (Magalhães 2015, Ribéreau-Gayon et al. 2006). Setting the harvesting date is a major concern for winegrowers, since it can enhance the quality of the grapes that are produced and, consequently, the wine's quality. It considers several factors, namely (Magalhães 2015), viz.

- Maturity control (i.e., contents of sugars, acids, phenolic and aromatic compounds);
- Meteorological conditions;
- Sanitary status of the grapes;
- Availability of labor;
- Reception capacity of the winery;
- Number of grape varieties; and
- Type of wine (i.e., white, red or pink)

The above shows that the decision-making process regarding the harvesting date is not a simple process. On the one hand, it involves analytical (objective) data, and on the other hand, subjective data, such as the appearance of the berry, its sweetness or the color of the peduncle. In addition, there are no known explicit relations between the variables that control the process, which causes the problem to be addressed using problem solving methodologies that emanated from the areas of Artificial Intelligence, Computer Science and Mathematical Logic.

STATE OF ART

Data analysis is not a new topic, as it has been done for several years, mainly using statistical methods. However, from a very early age, it became clear that the human brain analyzes data and treats information in a different way. In fact, the knowledge acquisition process in humans is done through a learning process, that may involve two distinct stages of information processing, i.e., the inductive form and the deductive one. In the former one, there is case-based learning, that is, patterns and rules are determined from data and experiences. In the later, the rules are used to create new facts. With regard to learning, the relationship between the learning model and its environment should also be highlighted. In this context there are fundamentally three paradigms (Han et al., 2012; Witten et al., 2017), viz.

- Supervised Learning is carried out from a set of cases or examples, i.e., one has prior knowledge of what the output values for one's samples are;
- Reinforcement Learning is a type of machine learning in which the system to be trained for a particular job is self-taught based on its previous experience and results while performing a similar task; and
- Unsupervised Learning, i.e., unmonitored learning is the training of machines using information that is neither classified nor tagged. The task is to group unsorted information into similarities, patterns, and differences without first having to train data.

Technological innovation has offered exponential growth in terms of data storage, both in terms of number of records or complexity. As a result of this effective increase in information, processed by traditional methods, has become increasingly difficult and complex. Thus, applications aimed at the task of knowledge discovery in databases incorporating Data Mining (DM) tools have emerged. The term knowledge discovery in database was used in 1989 as a reference to the broader concept of knowledge search in data, being a process that involves the identification and recognition of patterns or trends in a DataBase (DB), in an autonomous and perhaps automatic way. It involves data selection, data preprocessing, data transformation, data mining and interpretation. Generally, the objectives can be divided into two groups, i.e., forecasting and description. In the case of prediction, which goal is to infer future behavior based on past experiences, whereas in the description one aims to describe in a compact form data sets or variable associations (Han et al. 2012, Figueiredo et al. 2016, Witten et al. 2017).

The simplest definition of Decision Trees (DTs) is to represent a set of rules that follow a hierarchy of classes or values. It expresses a simple conditional logic and, from a graphical point of view, is similar to a tree, being a representation of a set of rules that allow to classify the instances when traversed from the root node to the terminal nodes or leaves. Each node in the tree specifies a test for the instance attributes (variables), and each downstream branch of that node corresponds to one of the possible values for that attribute. An instance is sorted by testing the attribute specified by the root node and then following the branch corresponding to the value of the attribute. The DT induction algorithms use supervised learning, i.e., the correct answers are provided to the system, from a set of examples where each consists of an input vector and an output or response one. The construction of the DTs is done from the data, in a recursive way, subdividing them until obtaining "pure" nodes, in which each node represents only a single class, or the satisfaction of a criterion of stop. The trees generated have the following structure (Han et al. 2012; Witten et al. 2017), viz.

- Pure sheets or nodes – corresponds to classes/objects;
- Internal nodes – specify tests performed on a single attribute giving rise to two or more sub-trees representing possible outputs; and
- Leaf node – correspond to the possible attribute values.

The algorithm ID3 (Quinlan 1999), was one of the pioneers that allowed the development of DTs. Since that time improvements and functionalities have been introduced that have resulted in the appearance of new versions and/or evolutions of this algorithm, as for example the algorithms C4.5 and C5.0. One of the limitations presented by the ID3 algorithm is that it can occur overfitting, which is manifested in the construction of trees with high performance in the training data, but with a low performance in the test data. In the algorithm C4.5 improvements were introduced that allowed to overcome the problem of overfitting through the introduction of tree's pruning. It consists in the reduction of some subtrees, transforming them into leaves, based on the comparison between the errors in this node and the sum of the errors in the nodes that descended from it. In this way, smaller trees can be obtained with greater predictability for new cases (Quinlan 1993). The induction of rules is often associated with DTs once they serve to express the knowledge represented in them. The most discriminant attribute (variable) is tested in the first node of the tree, and the following is tested in subsequent nodes according to their importance for classification. The representation is written in a syntax of type **IF** condition **THEN** action, a production that stands for itself. Rules have two levels, support and confidence. Support refers to the number of cases in which the rule is found, while confidence is the conditional probability of the rule (Han et al. 2012, Nisbet et al. 2017).

As a classic DM applications are usually considered medicine, telecommunications, finance, Internet, fraud detection,

marketing or production. However, some studies demonstrate the applicability of DM tools to other areas such as the production of drinking water (Vicente et al., 2015) or educational contexts (Figueiredo et al., 2016).

There are also some studies in the literature that demonstrate the applicability of DM methods to grape and wine production. Subramanian et al. (2001) present a method aimed at optimizing wine quality based on historical data from the environment and viticulture. The authors used DTs to predict the impact of wine-growing practices on wine quality. The study showed that only 6 out of a total of 20 input variables had a significant impact on the quality of the wine quality.

Another application of data mining tools has been described by Ribeiro et al (2009). The main objective of this study was to develop models for predicting the organoleptic parameters (color, foam, taste and taste) from the chemical parameters of the vinification process. The authors used DTs, Artificial Neural Networks (ANNs) and Linear Regression as DM techniques and achieved very good results with accuracies ranging from 86% to 99%. The ANNs performed better (96%) than the DTs (90%). Cortez et al. (2009) used a dataset of 6497 datasets to predict wine preferences based on physicochemical parameters. Three DM techniques were used under a computationally efficient method that performs the simultaneous selection of variables and models. The Support Vector Machine (SVM) achieved promising results, outperforming Multiple Regression and ANNs.

Hosu et al. (2014) used ANNs to predict the antioxidant activity of wine, based on the total content of phenol, flavonoids, anthocyanins and tannins. The correlation coefficient between outputs and targets had values of more than 0.993 for the training set.These works illustrate the use of various DM algorithms (e.g., DTs, SVMs, ANNs) and illustrate the central role that such tools could play in grape and wine production.

METHODOLOGY

Sampling

In the present work the sampling was carried out with berries, as it allows for a more representative sample of the parcel under study, as it comprises a larger number of vines reflecting different characteristics due to location, such as sun exposure, altitude or humidity (Hidalgo-Togores 2006). Each sample consisted of about 250 berries. Sampling was carried out in three different packages, weekly at 7 am from the beginning of the painting period (late July / early August) to the harvest (September) between 2014 and 2018. It was collected alternately. Berries exposed to the sun and shade, i.e., from within and outside the grapes (Hidalgo-Togores 2006). Grape vines that do not account for most of the parcel were excluded from sampling and diseased grapevines.

In the laboratory the berries undergo a separation process and then counted and weighed. For the extraction of the must from the berries, a household juice breaker was used. Diatomaceous earth was added to the resulting wort which is connected to the suspended particles, facilitating filtration through the Mostonet oenological filter.

Quantification of the Musts Chemical Parameters

The quantification of the physico-chemical parameters of the wines was carried out by Fourier Transform Infrared Spectroscopy (FTIR). It is an indirect analytical method, based on the absorption of infrared radiation by the sample under study. It consists on the detection of special vibrations associated to chemical bonds, allowing to identify and to determine with precision the concentration of the different mixes of the must (Correia 2011, Ferreira et al. 2009). However, the presence of all other matrix components that can absorb radiation at this wavelength will interfere with the quantification process (Correia 2011, Moreira et al. 2002). It is therefore necessary to establish an adequate analytical calibration beforehand to compensate for these interferences. To calibrate the equipment for the physicochemical parameters to be investigated, the results of the quantification using classical analytical methods are used as a reference (Correia 2011, Moreira et al. 2002).

Decision Trees

Given that this work aims to develop a self-explanatory model that allows for the prediction of the ideal harvest date based on eneological parameters, only the chemical parameters of the must have been used as input variables. The DT algorithm used in this study was the J48 as implemented in WEKA (Hall et al. 2009). This J48 implements the 8th revision of the commonly known C4.5 algorithm. A description of this J48 algorithm can be found in Witten et al. (2017). In fact, DTs offer many attractive features, such as a simple interpretation, allowing winemakers to gain insight into the critical physicochemical parameters of must that are most critical to determining the ideal harvest date.

Validation and Evaluation of the Models Performance

There are several methods to estimate the generalizability of a model, such as simple statistics, resubstituting validation, sample-split validation or cross-validation. In the present work it was decided on the sample subdivision validation method, considering the number of records contained in the DB (around one hundred). This method is based on dividing the data into two mutually exclusive subsets, one for generating the model, referred to as the training set, and the other for model validation, called the test set. The cardinality of these subsets is variable and depends, among other factors, on the size of the database (Han et al. 2012; Witten et al. 2017).

After the validation phase it is necessary to evaluate the performance of the model. In the case of classification models (i.e., models in which the output variable is discontinuous), one of the most common tool is the confusion matrix. It is a matrix L×L, (where L represents the number of classes of the output variable), in which the values predicted by the model and the desired values are mapped in the form of a two-value input table. The horizontal entries refer to the desired outputs, while the outputs returned by the model are in the vertical inputs, the cells being filled with the number of instances corresponding to the intersection of the. From the confusion matrix, the sharpness of the model can be calculated as a percentage of the correct answers in relation to the total number of cases inputs (Han et al. 2012).

RESULTS AND DISCUSSION

Characterization of the Must Samples

As already reported in the present study, experimental results were used for some physical-chemical parameters of must samples. Table 1 shows the statistical characterization of the results obtained so far. In terms of coefficients of variation, i.e., 100 × standard deviation/average, the analysis of Table 1 shows that the pH has the lowest coefficient of variation (6.2%), while the alcohol content, malic and potassium range between 17.32% and 18.2 %. For anthocyanins, assimilable nitrogen and tartaric acid, the values vary between 38.4% and 42.76%. Finally, the color intensity is the parameter with the highest coefficient of variation (78.6%). Considering that this metric provides a measure of relative dispersion, i.e., for the variability above the mean, the pH is the parameter with more homogeneous values, being the color intensity values the more distributed in relation to the mean. These results are consistent with the standards reported in the literature, once not only the course of color change is abrupt, but proceeds half walls with a berries' accumulation of free sugars, potassium, amino acids and phenolic compounds (Ribéreau-Gayon et al. 2006).

Table 1: A Statistical Characterization Regarding the Chemical Parameters of the Samples Studied

Parameter	Minimum	Maximum	Average	Standard Deviation	Coefficient of Variation (%)
pH	2.89	4.04	3.38	0.21	6.21
Total acidity (g dm^{-3})	1.06	10.86	5.80	1.68	28.97
Alcohol content (% v/v)	4.62	14.23	10.51	1.82	17.32
Antocyanins (mg dm^{-3})	14.9	332.4	108.3	44.0	40.6
Color intensity (Abs)	0.51	38.27	7.01	5.51	78.60
Assimilable nitrogen (mg dm^{-3})	46.0	369.1	156.3	60.0	38.4
Tartaric acid (g dm^{-3})	0.25	7.11	3.11	1.33	42.76
Malic acid (g dm^{-3})	1.38	3.16	2.00	0.35	17.50
Potassium (mg dm^{-3})	63.3	517.8	398.3	72.7	18.2

A Decision Support System to Forecast the Harvest Date

The way the data is structured shapes up the success and the pursuit of an intelligent analysis of itself. Data preprocessing techniques aim to improve the quality of the data, contributing to more efficient and accurate subsequent phases, namely the mining ones. One of the aspects that must be considered at this stage is the possible disproportion between the number of cases of each class of the output variable, as is the case in the present work, where 82% of the examples recorded in the database referred to the situation of not harvest and only 18% concerned the harvest situation. This imbalance can lead to models with worse performance in the minority class forecasts. The reasons are twofold. The former one derives from the fact that the rules generated for the minority classes are based on fewer examples, and hence more over-adjusted, i.e. the classifiers will tend to learn more rigid limits of the hypothesis under consideration. The later has to do with the fact that the most frequently predicted classes are the majority, and therefore there is a greater probability of incorrectly classifying minority class examples (Han et al. 2012).

There are, fundamentally, two methods to make class distribution more balanced. One of the methods, called under-sampling, is to create a smaller sample of the set of examples from the majority classes. The other, called over-sampling, is to generate cases from the initial cases of the set, in order to increase the number of minority classes cases. These two methods are associated with disadvantages, such as neglecting potentially useful data in the former and, in the latter case, increasing the training set increases the computation time and can lead to overfitting.

In this work, considering the number of samples belonging to each of the classes, in order not to risk discarding potentially useful cases, the over-sampling method was chosen.

The harvest date forecast was set as a classification problem. The induction of rules and decision trees are grounded on data mining techniques for data gathering, allowing for an expressive representation of knowledge, easy to use and understanding.

The decision tree obtained is shown in Figure 1 and begins by testing potassium content at the root node. Samples whose content in this cation is less than or equal to 379.0 mg dm^{-3} are not capable of being harvested. In case the potassium content were higher than 379.0 mg dm^{-3}, the examples were divided according to the malic acid content. Thus, when the concentration of this parameter is higher than 1.77 g dm^{-3}, the grapes are ready to be harvested. When the malic acid concentration is less than or equal to 1.77 g dm^{-3}, the cases are divided according to the pH value. Thus, samples whose pH is less than or equal to 3.53 are not in a position to be harvested, contrary to those whose pH is higher than 3.53 that can be harvested. Table 2 shows the confusion matrix for the developed model, where the values presented are the means for thirty simulations. The acuity of the model, measured in terms of percentage of correct answers in relation to the number of cases presented is 94.8% for the training set, and 93.2% for the test set.

Table 2: Confusion Matrixs Related to the Model Presented for the Prediction of the Harvest Date

Harvest	Training Set		Test Set	
	Yes	No	Yes	No
Yes	48	0	22	0
No	5	43	3	19

The set of rules generated from the decision tree is as follows, viz.

Rules for the **Harvest Class_No**

- **If** Potassium ≤ 379 mg dm^{-3}
  Then → Harvest_No
  (Confidence = 1; Support = 26)

- **If** Potassium > 379 mg dm^{-3}
  And Malic Acid ≤ 1,77 g dm^{-3}
  And pH ≤ 3,53
  Then → Harvest_No
  (Confidence = 1; Support = 17)

Rules for the **Harvest Class_Yes**, viz.

- **If** Potassium > 379 mg dm^{-3}
  And Malic Acid ≤ 1,77 g dm^{-3}
  And pH > 3,53
  Then → Harvest_Yes
  (Confidence = 0,92; Support = 12)

- **If** Potassium > 379 mg dm^{-3}
  And Malic Acid > 1,77 g dm^{-3}
  Then → Harvest_Yes
  (Confidence = 0,90; Support = 41)

The rules obtained are highly supportive, i.e., they apply to a large number of cases, varying between 12 and 41. With respect to confidence, i.e., the percentage of correctness of the rules, where gotten values ranging from 90% to 100%.
Since the output variable presents only two alternatives, it is possible, based on the confusion matrix (Table 2), to calculate the sensitivity and specificity of the model. The first metric translates the proportion of positive cases that were correctly classified as such, while the second one accounts for the proportion of negative cases that were correctly classified as such (Florkowski 2008). The sensitivity of the model was 100% for both the training set and the test set. This result means that the model correctly classifies all positive cases, i.e., cases in which the grapes are ready to be harvested. Regarding the negative cases, the specificity of the model was 89.6% and 86.4%, respectively for the training and test sets. These values allow to affirm that the model has an acceptable performance in the classification of negative cases.

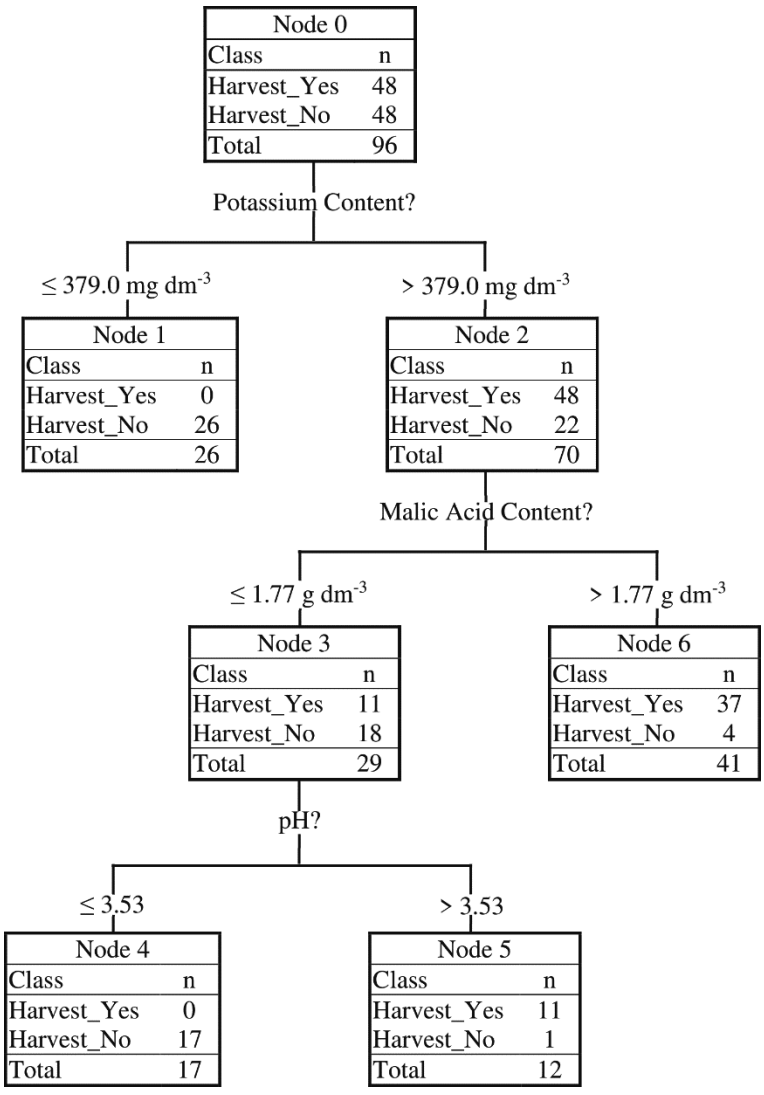


Figure 1: Decision Tree Model for the Prediction of the Harvest Date

CONCLUSIONS

In the present work the evaluation of some physical-chemical parameters of must samples of the variety "*trincadeira*" were carried out, in different maturation stages, in order to develop forecast models for the date of harvest. The results showed that the variability in relation to the average pH, alcohol content, malic acid and potassium is lower than that of anthocyanins, assimilable nitrogen, tartaric acid and color intensity.

Predicting the date of harvesting has proved to be a difficult task since not all of the parameters required for harvesting are fully represented by objective data and sometimes incomplete one. Thus, in this study, in addition to the evaluation of the parameters mentioned above, a Decision Support System based on Decision Trees was presented. The proposed model is able to provide adequate responses, with an overall acuity close to 95%.

Future work suggests the validation of this model using harvest data for the last years. On the other hand, it would be interesting to test if this model can be applied to other varieties besides "*trincadeira*". In addition, new models can be developed to forecast the date of harvest using other computing paradigms such as k-nearest neighbors, artificial neural networks, case-based reasoning, genetic programming, particle swarms or ant colonies, just to name a few.

ACKNOWLEDGEMENTS

This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2019.

REFERENCES

Blouin, J.; and E. Peynaud. 2012. Knowing and Making Wine, 5th Edition, Dunod, Paris, France. (In French).

Blouin, J.; and G. Guimberteau. 2000. *Maturation and Ripeness of the Grapes*. Féret, Bordeaux, France. (In French).

Correia, C. 2011. Infrared Spectroscopy in the Analysis of Musts and Wines. MSc Thesis, University of Aveiro Edition, Aveiro, Portugal. (In Portuguese).

Cortez, P.; A. Cerdeira; F. Almeida; T. Matos; and J. Reis. 2009. "Modeling wine preferences by data mining from physicochemical properties." *Decision Support Systems* 47, 547-533

Ferreira, M.L.; A.M. Costa; N. Ribeiro; T. Simões; and P. Barros. 2009. "Quality Control in FTIR Wine Analysis – Acceptance of Analytical Results." *Ciência e Técnica Vitivinícola* 24, No. 1, 47-53.

Figueiredo, M., L. Esteves; J. Neves; and H. Vicente. 2016. "A data mining approach to study the impact of the methodology followed in chemistry lab classes on the weight attributed by the students to the lab work on learning and motivation." *Chemistry Education Research and Practice* 17, No. 1, 156-171.

Florkowski, C.M. 2008. Sensitivity, Specificity, Receiver-Operating Characteristic (ROC) Curves and Likelihood Ratios: Communicating the Performance of Diagnostic Tests. *The Clinical Biochemist Reviews* 29, Suppl 1, S83-S87.

Hall, M.; E. Frank; G. Holmes; B. Pfahringer; P. Reutemann; and I.H. Witten. 2009. "The WEKA Data Mining Software: An Update." *SIGKDD Exploration* 11, No. 1, 10-18.

Han, J.; M. Kamber; J. Pei. 2012. *Data Mining: Concepts and Techniques*. 3rd edition, Morgan Kaufmann, Waltham, United States of America.

Hidalgo-Togores, J. 2010. The Quality of the Wine from the Vineyard. Mundi-Prensa Libros, S.A., Madrid, Spain. (In Spanish).

Hosu, A.; V.-M. Cristea; and C. Cimpoiu. 2014. "Analysis of total phenolic, flavonoids, anthocyanins and tannins content in Romanian red wines: Prediction of antioxidant activities and classification of wines using artificial neural networks." *Food Chemistry* 150, 113-118.

Magalhães, N. 2015. *Treaty of Viticulture: Vine, Vineyard and Terroir*. 2nd Edition, Esfera Poética, Lisbon, Portugal. (In Portuguese).

Moreira, J.L.; A.M. Marcos; and P.Barros. 2002. "Analysis of Portuguese Wines by Fourier Transform Infrared Spectrometry." *Ciência e Técnica Vitivinícola* 17, No. 1, 27-33.

Nisbet, R.; G. Miner; and K. Yale. 2017. *Handbook ok Statistical Analysis & Data Mining Applications*. 2nd Edition, Academic Press, San Diego, United States of America.

Quinlan, J.R. 1993. *C4.5 Programs for Machine Learning*. Morgan Kaufmann Publishers Inc, San Francisco, United States of America.

Quinlan, J.R. 1999. Simplifying decision trees. *International Journal of Human-Computer Studies* 51, No. 2, 497-510.

Ribeiro, J.; J. Neves; J. Sanchez; M. Delgado; J. Machado; and P. Novais. 2009. "Wine Vinification Prediction using Data Mining Tools". In *Computing and Computational Intelligence*. G. Sirbiladze, A. Sikharulidze, I. Khutsishvili, and T. Lominadz (Eds.). WSEAS Press, 78-85.

Ribereau-Gayon, P.; D. Dubourdieu; B. Donèche; and A. Lonvaud. 2006. *Handbook of Enology: The Microbiology Of The Wine And Vinifications*. Volume 1, 2nd Edition, John Wiley & Sons Ltd, Chichester, United Kindgon.

Subramanian, V.; K. Buck; and D. Block. 2001. "Use of Decision Tree Analysis for Determination of Critical Enological and Viticultural Processing Parameters in Historical Databases." *American Journal of Enology and Viticulture* 52, No. 3, 175-184.

Toda, F. M. 2011. *Keys to Quality Viticulture*. 2nd Edition, Mundi-Prensa Edition, Madrid, Spain. (In Spanish).

Vicente, H.; F. Borralho; C. Couto; G. Gomes; V. Alves; and J. Neves. 2015. "An Adverse Event Reporting and Learning System for Water Sector based on an Extension of the Eindhoven Classification Model." *Water Resources Management* 29, No. 14, 4927-4943.

Witten, I.H.; E. Frank; M.A. Hall; and C.J. Pal. 2017. *Data Mining – Practical Machine Learning Tools and Techniques*. 4th Edition, Morgan Kaufmann, Cambridge, United States of America.

AUTHOR BIOGRAPHIES

BEATRIZ CHARNECA was born in Azaruja, Évora. She obtained her degree in Biotechnology at the School of Sciences and Technology of the University of Évora in 2017. Her current interests are in the area of agriculture, and presently is obtaining a new degree in agronomy, again at the School of Sciences and Technology of the University of Évora.

VANDA SANTOS was born in Évora, Portugal. She obtained her degree in Chemistry at University of Évora, in 2009. She is currently working in a oenological analysis laboratory.

ANA CRESPO was born in Esperança, Portugal. She obtained her degree in Biotechnology at the University of Évora in 2019. Her current interests are in the area of oenology and wine production. She participated in a study to evaluate the knowledge of the customers of a rural tourism company with a vineyard attached regarding the toxicological risk assessment associated to the pesticide use.

HENRIQUE VICENTE was born in S. Martinho do Porto, Portugal and went to the University of Lisbon, where he studied Chemistry and obtained his degrees in 1988. He joined the University of Évora in 1989 and received his Ph.D. in Chemistry in 2005. He is now Auxiliary Professor at the Department of Chemistry at the University of Évora. He is a researcher at the Évora Chemistry Center and his current interests include Water Quality Control, Lakes and Reservoirs Management, Data Mining, Knowledge Discovery from Databases, Knowledge Representation and Reasoning Systems, Evolutionary Intelligence and Intelligent Information Systems.

HUMBERTO CHAVES was born in Caldas da Rainha, Portugal. He is Coordinating Professor in the School of Agriculture, Polytechnic Institute of Beja (IPBeja), Portugal. He holds a PhD (Organic Chemistry – Computational Chemistry Applied to the Study of Mechanisms of Chemical Reactions) in 1998 at the New University of Lisbon, Portugal and Imperial College, London, United Kingdom. His main research interests are Biocombustibles, Waste Recover, Waste Water Treatment, Water Analysis and Safety in Laboratories.

JORGE RIBEIRO is Adjunct Professor in the School of Technology and Management Polytechnic of the Institute of Viana do Castelo (IPVC) – Portugal, since 2006. He holds a PhD (Informatics Engineering) in 2011 at the Department of Electronic and Computation of the University of Santiago de Compostela, Spain. He is affiliated with the Algoritmi – Research and Development Unit of the School of Engineering – University of Minho. His main research interests are Artificial Intelligence, Software Engineering and Knowledge Discovery in Databases and he has been author and co-author of some papers in the field of Data Mining, Software Engineering, Knowledge Representation, Evolutionary Systems and DataMining using Geographic Information Systems.

VICTOR ALVES holds a master's degree in 2011 (Technology and Management of Systems Information at the Institute of Viana do Castelo – Portugal) and an Informatics and Management graduate degree in 1998 (Fernando Pessoa University – Portugal). He is professor of many Information Technology topics in secondary and vocational schools and since 2011 he is invited professor at the School of Technology and Management Polytechnic of the Institute of Viana do Castelo (IPVC), teaching topics related with software development, operating systems, security and governance. His main research interests are artificial intelligence, security and governance of systems and information technology, software engineering and knowledge discovery in databases. In the last decade, and nowadays he is project manager and software developer in many projects in companies and in public institutions.

JOSÉ NEVES is Emeritus Professor of Computer Science at Minho University, Portugal, since 2019. José Neves is the Deputy Director of the Division for Artificial Intelligence. He received his Ph.D. in Computer Science from Heriot Watt University, Scotland, in 1983. His current research interests relate to the areas of Knowledge Representation and Reasoning, Evolutionary Intelligence, Machine Learning, Soft Computing, aiming to construct dynamic virtual worlds of complex symbolic entities that compete against one another in which fitness is judged by one criterion alone, intelligence, here measured in terms of a process of quantification of the quality of their knowledge, leading to the development of new procedures and other metaheuristics and their application in complex tasks of optimization and model inference in distinct areas, namely in the Healthcare Arena and The Law.

OPERATORS' STRUCTURAL KNOWLEDGE EVALUATION AS A FEEDBACK IN COMPUTER-BASED TRAINING IN CONTEXT OF THE NOVEL "ANNA KARENINA"

Victor Dozortsev
AO "Honeywell"
7 Kievskaya str.
121059, Moscow, Russia
E-mail: victor.dozortsev@honeywell.com

Anastasia Mironova
Yandex
16 Lva Tolstogo str.
119021, Moscow, Russia
E-mail: an.mironova.gml@gmail.com

KEYWORDS

Industrial process' operator, computer-based training, structural knowledge, graph theory, coherence, PathFinder technique

ABSTRACT

This paper aims to bring to light how changes in industrial process operators' structural knowledge can be grasped, represented, evaluated and used as a feedback in operators' computer-based training. Under the assumption of similarity of complex multi-connected industrial process and branched literary text, a pilot experiment with the readers of Tolstoy's novel "Anna Karenina" was conducted. The results of the experiment are given and discussed.

INTRODUCTION

Importance of quality training of complex industrial process' (IP) operators can hardly be overestimated in modern world. Industries suffer huge losses, a large portion of which (from 40 to 90% depending on the industry (Johnsen et al. 2017)) occur due to the human factor. Tragedies of recent years have led the professional community to understand the need of focusing on a "systematic approach for managing the human factor" (U.S. Chemical Safety and Hazard Investigation Board 2016). Industrial security challenges, the demographic pitfall of recent decades requires a substantial revision of the operating personnel's training procedures in industries.

One of the important and still unsolved problems that negatively affect quality and fullness of training for beginners, as well as the development and strengthening of skills among professionals, is the lack of valuable feedback to a human operator during the training process. In modern manufacturing, where assets worth billions of dollars are controlled by fewer and fewer number of personnel, computer-based training (CBT) remains the most popular way of training industrial process' (IP) operators. CBT systems are based on high-precision modeling of IP (Dozortsev and Kreidlin 2010) and accurate playback of operators' control environment (Dozortsev 2009). However, even CBT systems cannot fully provide objective feedback during the training process. Until now, the instructor of CBT remains the key source of such feedback, while the pressure of instructor's responsibility and the subjectivity of their assessments significantly narrow the training effectiveness.

It was shown (Mironova 2018) that well-known in cognitive psychology concept of structural knowledge (SK) allows the addition of the necessary automated feedback to the training process. In essence, SK is a system of links of human operator's key concepts about the controlled object. The "richer" and "more reliable" SK, the deeper and more complete the operator understands the structure, principles of operations and the development of possible scenarios of the object's behavior, the higher is professional level of the trainee. The essential diagram of the use of SK as a feedback in personnel training is shown if Fig.1.

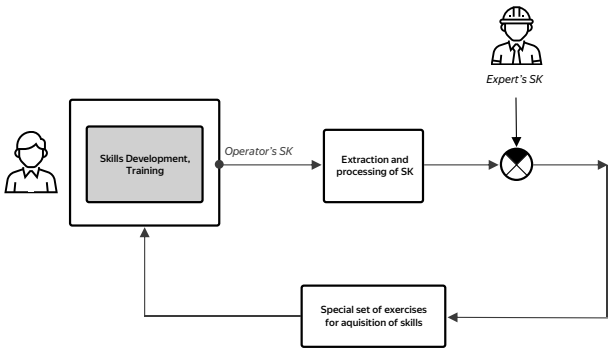


Fig. 1: Training scheme with SK integrated into feedback loop

In the skills development block, traditional training is carried out (for instance, based on CBT), and according to the results information about SK is extracted. After that this extracted SK is compared with the reference, expert SK. Obtained differences (if necessary – in dynamics) are used to correct a special set of exercises aimed at reduction of soft spots in the understanding and IP control skills.

First experiments in the framework of the described approach give us hope of success (Dozortsev et al. 2015), but issues of extracting, representing and evaluating SK in the context of learning to control multi-connected systems remain extremely relevant. First of all, one should mention the complexity of IP organization, which includes many hundreds of measurements, many dozens of control variables, a distributed control system and an extensive system of locks and protections. All these elements, while constantly changing and interacting, behave differently in different

technological situations. This, in fact, explains the objective complexity and expensiveness of operators' training.

Indicated property, along with others (non-reproducibility of technological regimes, openness to material and energy flows) determines the similarity of IP with living organisms noted by some researches. Less often (or not at all) they emphasize the similarity of a complex IP to artifacts of narrative art, for example, literary works contain many characters connected in a dense net of relationships and mutual influences.

An excellent example of such a literary object is Leo Tolstoy's novel "Anna Karenina", a widely known and recognized masterpiece of Russian and world literature of the 19th century. The novel is extremely complex, representing a whole world with many dozens of major and minor characters interacting and changing from chapter to chapter. At the same time, the novel is unusually slender and transparent and is perceived as a whole and realistic universe. It is known that Tolstoy was very proud of the architecture of his work, where "the vaults are brought together, so it's impossible to notice"[1]. Original "Tolstoy's" philosophy unites even seemingly weakly connected characters. Thus, Anna and Levin, the author's favorite protagonists, rotating in different society circles, relentlessly collide in the last part of the book. It is a special, non-linear art structure that makes the novel a rare example of a complex multi-connected system.

Possibly a non-evident idea of the authors of the present work is to test the proposed approach using the novel "Anna Karenina" as an example. Although the expected result is illustrative, its comprehensibility and emotional perception by a wide audience make it possible to reach a more general level of problem analysis, i.e., whether it is possible to meaningfully reproduce the whole structure of a complexly organized object on the basis of existing and newly proposed procedures for extracting, presenting and evaluating SK. Initially, an extremely brief digest of the novel will be given in order to remind the main characters and the ups and downs of the storyline. Then the available tools will be described and the developed formalism of the SK analysis will be presented. Finally, a pilot experiment with the participations of various readers of the novel will be described. Its results will be discussed and goals of future research will be stated.

ABOUT THE NOVEL "ANNA KARENINA"

The action begins at the Moscow home of the Oblonsky princes, where **Stiva** Oblonsky was catch cheating on his wife, **Dolly**. His sister, **Anna**, came from St. Petersburg to reconcile the spouses. Later, Stiva met with a friend of his youth Konstantin **Levin**. Levin came from the village to propose to the 18-year-old princess **Kitty** Shcherbatsky, whom he had long fallen in love with. However, Kitty, didn't defined her own feelings yet, so she refused Levin for the sake of count **Vronsky**, a brilliant young officer.

While meeting his mother, countess **Vronskaya** who came from St. Petersburg, Vronsky saw Anna at the train station for the first time. The meeting was overshadowed by a sad incident – the death of the watchman under the wheels of the train. Anna brought together Stiva and Dolly and then returned to St. Petersburg. Vronsky, who was already in love with her, followed her and found an opportunity to confess to her. In St. Petersburg Anna was increasingly burdened by her husband, a major official Alexey **Karenin**. Her affair with Vronsky was rapidly developing, it became known to the society and Karenin. Karenin required Anna to save the appearance of a happy family life under the threat of her separation from their son, **Seryozha**.

After unsuccessful marriage proposal, Levin met his brother **Nikolay**, who was passionate about communistic ideas, sick and heavily drunk and then returned to his country-house. In the village, where Levin was reading a lot and was writing a book about agriculture, his step-brother Sergey **Kozhnyshev** arrived. They talked a lot about peasants' life. But learning about Kitty's illness after Vronsky left her, Levin realized that he still in love with her. In Moscow, he made a proposal again and got Kitty's consent. After wedding, the young family went to the village, where Levin's brother Koznyshev fell in love with Kitty's friend, **Varenka**, but didn't dare to marry her.

At that time Anna, Vronsky and their newborn daughter had been travelling in Italy. Anna felt happy, but Vronsky yearned for his past times in St. Petersburg. The couple returned to Russia. There Anna clearly felt her rejection from the society, almost everyone avoided meeting her. Even Princess **Betsy** Tverskaya, a relative of Anna and Vronsky, was not ready to have Anna at home. The couple went to Vronsky's country estate and lived a quiet solitary life, awaiting Karenin's decision regarding a divorce. Their relations became more difficult, and Karenin (under the influence of his religious friend, princess **Lidia Ivanovna**) refused to give a divorce.

Being in a whirl, Anna was disappointed in her passionate and selfish love of Vronsky. At the train station, she remembered the man who had been killed by train on the first day of her meeting with Vronsky. She didn't see the chance to unleash a hopeless situation and threw herself under the passing train.

In the most general form the key links between characters of the novel are shown in Fig. 2, reflecting the connection of two central storylines: St. Petersburg line of Anna, Vronsky and Karenin (the bottom part of the graph), and Moscow line of Levin and Kitty (the top part). Main points of intersection pass through Stiva (Anna's brother and Levin's friend at the same time), his wife Dolly (at the same time Kitty's sister), and also Kitty herself, who is in love with Vronsky in the first part of the novel. Intersections give the text a new complexity and depth, the perception of which obviously depends on the experience and sophistication of the reader.

[1] L. Tolstoy. The Letters. 324. To S. Rachinsky. January 27, 1878. Yasnaya Polyana. // L. Tolstoy. Collected works in 22 volumes. M.: Khudozhestvennaya Literatura, 1984. V. 18. P. 820—821.

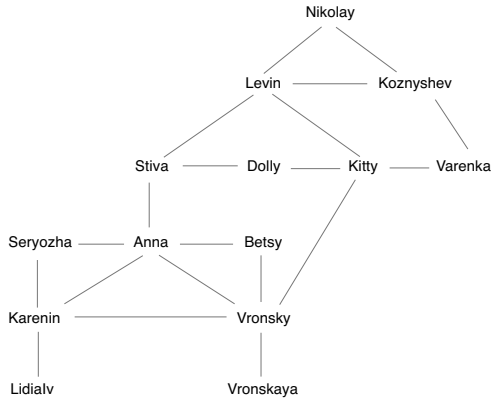


Fig. 2: A set of key links between the novel's characters

TECHNIQUES AND MEASURES OF STRUCTURAL KNOWLEDGE EXTRACTION, REVEALING AND EVALUATION

Handling of structural knowledge (SK) requires special techniques. There are plenty of them that could somehow represent mental models of knowledge. The most commonly used are briefly described in the following section.

Spatial methods such as multidimensional scaling are particularly popular (Petrenko 1993). They are extremely useful on massive datasets, but when one need to conduct more accurate and precise analysis of mental structures they are not efficient enough. Inaccuracy in space reduction and misinterpretation of choosing axes of coordinates affect final structure significantly.

Discrete methods such as hierarchical clustering are also useful and important instrument of analyzing knowledge structures. The soft spot of methods of this type is the strong dependence on clusterization method. It's possible to achieve diverse results at the output depending on what clusterization method was chosen.

Network methods represent knowledge structures in the form of graphs. It is well known that every matrix can be described as a full graph where each pair of nodes is connected by an edge. Therefore, if one has a similarity matrix of concepts, it can be easily transformed into a graph where nodes represent concepts and edges – similarities/connections/impacts between them. The main purpose of network methods is to create the simplest graph describing raw data in the best way. The major investigations were conducted by Roger Schvaneveld (Schvaneveldt 1990).

The main approach of the technique is that to create final graph algorithm chooses only those links between nodes-concepts, which are the shortest paths in graph. To recap a bit, path in graph theory is the consequence of adjacent nodes. Algorithm finds in the graph paths of minimum length, i.e., the shortest transitive pass from one concept to another.

To regulate graph's density and detailed composition PathFinder uses r and q parameters. R-parameter specifies how the weight of the path calculates from weights of edges contained. If $r = 1$, then all components make equal contributions to the total path weight. If r-parameter increases, then components of large magnitude start to make bigger contribution. Therefore, for $r \to \infty$ the heaviest component specifies total path weight. Q-parameter is responsible for the density of final graph. If graph is too dense, then it is inconvenient to interpret and undescriptive. For instance, Fig.3 shows graphs of the same raw datasets, the left one has maximum possible density (each node is connected to another), and the right one is sparse. For $q = 1$ final graph matches exactly initial (all connected to all) network. Increasing brings about density decrease.

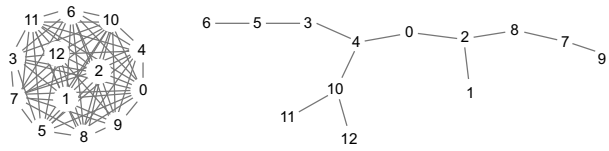


Fig. 3: Graphs of different densities — after processing by PathFinder algorithm

It's worth to mention, that graphs themselves are particularly interesting for close detailed analysis. But if one need to create general instrument of knowledge structure analysis, applicable to various objects and systems from different fields of knowledge, it is important to have a range of quantitative tools. These are measures that one will rely on to get valid feedback in training process. Let's discuss some of quantitative measures based on various graph parameters.

The first measure, coherence, characterize raw matrix itself. Coherence is responsible for consciousness in filling-in matrix. The idea of coherence is based on assumption that if two concepts have similar relationships with the others, then these two concepts should be similar to one another. Investigations (Meyer 2008) shows that extremely low Coherence (less than 0.15) may indicate that the rater did not take the task seriously, or the rater did not understand the concepts very well.

The next and simple one is the number of links in the network. This characteristic shows how sparse and clear structural knowledge is. Experts and professionals usually show precise and accurate networks, whereas, in contrast, newbies show indistinct very dense and often chaotic structures. Note that excessively poor networks may indicate gaps and blind spots in knowledge.

The other important characteristics are diameter or weighted diameter of the graph. In graph theory diameter is the distance between the most remote nodes of the graph. In essence, diameter characterizes breadth and sparseness of the network. Some investigators employ the measure of diameter itself, but the measure of weighted diameter seems much more adequate in the case of describing qualitative characteristics of SK. In effect, diameter itself describes how indirect links are formed in the SK. In other words, how indirect mediated links (and not only straight and short) between concepts are developed.

It seems that rich SK should include long chains of links. It may reflect that certain concepts, although not directly connected, are linked through strong chains of concepts. This does not contradict the presence of direct links in developed SK, but it also emphasizes the presence of long dependencies. The weighted diameter operates not only with the number of nodes in this chain, but also with the total connection strength in it. The high value of the weighted diameter characterizes the strongest mediated dependence of concepts in SK.

The same reasoning goes for the density and weighted density. Just density (which is the proportion of significant links in the SK of all possible links (1)) characterizes the forcing of the SK in the sense of discarding nonessential links and focusing on the essential ones.

$$d = \frac{2l}{n(n-1)} \qquad (1),$$

where n is the number of graph nodes, and l is the number of graph edges.

Since these links may be different in strength, the weighted density indicates the proportion of the total strength of the essential links from the total maximum possible strength of all links (2).

$$d_w = \frac{2}{n(n-1)} \sum_{i=1,j=i+1}^{n} x_{ij} \qquad (2),$$

where n is the number of graph nodes, and x is the weight of a single link.

Note, that the total maximum possible strength of all links is constant for networks with the same number of nodes. As well as density, weighted density reflects the ability of the subject to choose the most significant of all links, and in the case of weighted density, takes into account their strength.

To enhance depictive effect of both coherence measure and weighted diameter measure, the integrated measure of (Coherence)/(Weighted Density) was applied. This measure amplifies properties of both measures and lets us deal with one integrated mark instead of two.

To summarize, all of described measures are interesting not so much as the valuable characteristics of the object itself, but rather as the parameters of the subjective assessment of the object by subjects. Values of these parameters may vary in time, reflecting the evolution of the subjects' views, and may also differ for different subjects (and, of course, for experts), showing a different level of formation of the SK.

PILOT EXPERIMENT WITH THE NOVEL "ANNA KARENINA"

10 participants of different professions, ages, and genders took part in our trial experiment. All of them also had diverse level of expertise in the novel and various ways of knowing the novel (the book itself, movies, theater plays, which is almost equally important as the original text nowadays). Some of testees had rather strong background in literature and liberal arts, others had barely read the novel at school. All participants were received matrices with 14 parameters, each parameter corresponded to a character in the book. When selecting key characters, both main and secondary ones were chosen. Thereby, the final list consisted of the parameters described above, in bold.

The task was to estimate pairwise how strong one parameter is connected to another one. Scale of estimation was from 0 to 5, where 0 stands for "not connected at all" and 5 stands for "connected as much as possible". Participants haven't had any time limits for completing the task. All similarity data were processed using PathFinder technique. Measures described above were calculated for each case.

Each knowledge structure is revealed as unique graph, nevertheless, some structures have something in common. Graphs of the most experienced and educated testees have obviously distinct and clear structures. Figure 4 below illustrates that despite on differences in three "best" structures, all of them imply apparent patterns.

There are chains of directly connected parameters and there are also circles and groups representing clusters of characters. For example, graph of testee 1 includes several circles: "Kitty-Levin-Nikolay-Koznyshev-Varenka" represents close relationships between Levin and his entourage, triangles "Anna-Karenin-Seryozha" and "Anna-Stiva-Dolly" reflect close families' links. At the same time, the key link "Anna-Vronsky" is absorbed into broader groups of characters including Dolly, Kitty and even Levin. Generally, testee 1 inclines to cyclic organization of SK. Moreover, maximum cycle includes 8 characters from 14 presented and contains 5

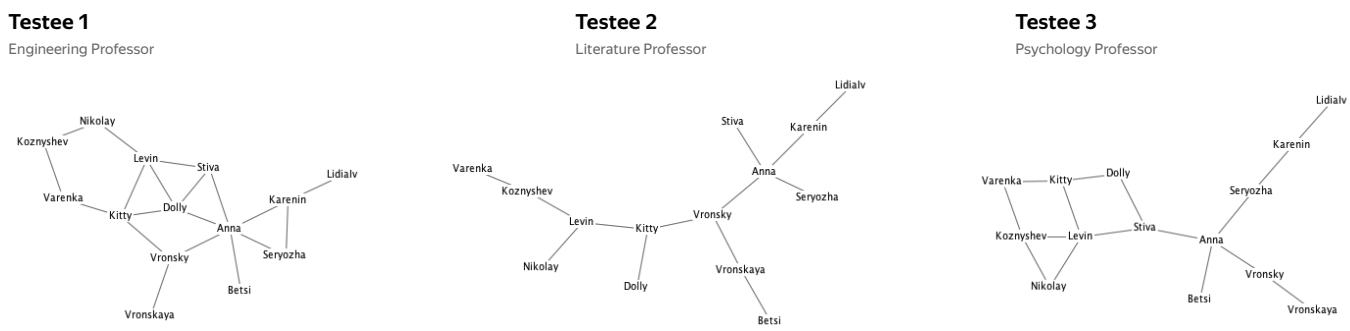


Fig. 4: Examples of "good" knowledge structures

sub-cycles. In comparison with the others "best" testees their graph is denser and consists of stronger links.

Some of such types of circles one can also observe on testee's 3 graph, although their number is much less (the largest cycle combines 7 characters and includes only 3 sub-circles).

Graph number 2 is especially interesting because it belongs to the Literature professor. It is the most sparse and distinct graph of all three. Basically, it represents two chains, connected through Anna and Vronsky (main protagonists), which is perfectly corresponds to the Tolstoy's initial intention of "brining arches together". On one side of that arch one can see the line of Anna, her son, husband and brother, and on the other side, connected through Vronsky and his ex-love Kitty, one can see the line of Levin and his family.

Note, that graph 2 is less dense than the others and doesn't contain cycles at all. For instance, the link between Nikolay and Koznyshev is absent (both are Levin's brothers, but they are not so close by storyline) as well as the link between Karenin and Seryozha (the father is quite indifferent to the boy) and even the link between Stiva and Dolly (husband and wife), who are very close by storyline, but have a deep inherent breakup. Testee's 2 SK obviously reflects not formal, but essential dependencies between characters.

Regarding the graphs that show much worse knowledge and understanding, all of them belong to young people who read the novel at school a couple of years ago. They "more or less remember the storyline and main characters".

Looking at the Figure 5, one can see that structures are more chaotic and dense. For the first of them (testee 8) almost fully cross-connected structure of 7 main characters is relevant. At the same time the rest characters are "linked" to someone by paired links. In other words, the role of secondary characters in the action is underestimated.

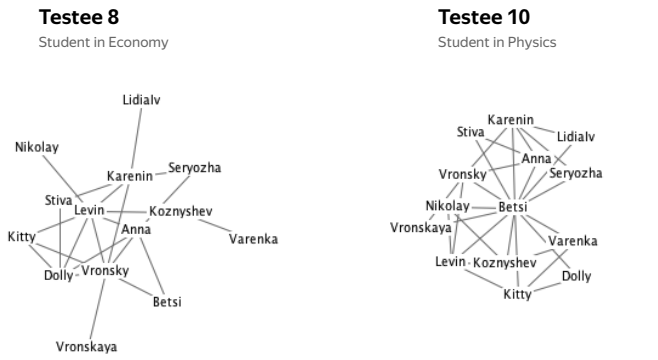


Fig.5: Examples of "poor" knowledge structures

Finally, testee's 10 graph illustrates situation of poor chaotic knowledge. One can see a lot of close links, almost every character is connected to another, that means that testee 10 doesn't have understanding which relationships are important. The center point of the graph becomes Betsy, society woman, who in fact knows everyone, but doesn't have an impact on main storylines and aren't one of the key characters.

These examples illustrate clear difference between "good" and "bad" knowledge structures. But to get that understanding one still need to make fine investigation of each graph. Consequently, let's discuss results of measures described above in application to measured data. Table below illustrates Coherence, Number of Links, Diameter and Weighted Diameter, Density and Weighted Density and also the ratio of Coherence to Density.

Table 1: Graph measures applied to the experiment results

	Coherence	N Links	Diameter	Weighted Diameter	Density	Weighted Density	Coh/Wdens
#1	0.73	19	6	27	0.21	0.97	0.75
#2	0.72	13	7	31	0.14	0.60	1.19
#3	0.66	16	7	31	0.18	0.77	0.86
#4	0.63	18	5	24	0.20	0.88	0.72
#5	0.58	16	5	23	0.18	0.73	0.80
#6	0.57	18	6	26	0.20	0.82	0.69
#7	0.54	20	5	18	0.22	0.98	0.55
#8	0.48	22	4	18	0.24	0.99	0.49
#9	0.48	16	7	22	0.18	0.55	0.87
#10	0.42	28	2	6	0.31	1.23	0.34

Data were sorted by magnitude of coherence and the order perfectly corresponds to our quick analysis of the "best" and the "worst" structures. All three best structures are on the top. One can also notice that better structures possess bigger weighted diameter. That corresponds to the idea of that broad and vast knowledge indicates that the meaning of the object doesn't come to the number of intensively connected principal elements, but also is determined by presence of strong mediated links between parameters that don't interact directly. The Number of Links and Weighted Density testify that indistinct and chaotic structures do have higher values of these measures. For example, the worst case of testee 10 shows extremely high value of Weighted Density – far more then 1, whereas the Literature professor (testee 2) has rather small weighted density. Finally, the integrated measure of (Coherence)/(Weighted Density) also shows correspondence to our reasoning: in general, experts from the top of the table with higher level of expertise have bigger values than the bottom ones.

CONCLUSIONS AND FUTURE TASKS

It is possible that the experiment described above with a well-known literary text may seem far-fetched. However, in addition to the already noted obvious structural similarity of the IP and the Tolstoy's text as massive multiply connected objects, let us point out the following important feature. Readers perceive the novel not directly, but through the prism of various cultural interpretations. Often this perception is

emotional, biased and sometimes even influenced by the self-identification with characters. Paradoxically, an operator too sees not only deterministic “stimulus-response” relations for specific connections, but also the manifestation of the essential interdependencies between elements of a technological object, that become visible to him due to his extensive professional experience. In this sense, subtleties of estimating SK educed in literature experiment can help in building reliable feedback in operators’ training.

Authors are far from thinking about offering teachers to use the described methodology to identify “weak” points in the perception of the Tolstoy’s text and to assign individual tasks regarding rereading certain chapters of the book, although the teacher equipped with measurements described above will indeed be better prepared to meaningful conversation with students.

In contrast, for the instructor of the computer-based training, individual operator graphs (especially in comparison with his own) are a rich practical resource for improving knowledge and skills of the trainees.

SK is an important instrument of analyzing understanding of an object or system studied. That is why the investigation of its organization and characteristics can help significantly improve existing training processes. Analyzing in real time changes in SK of the trained operator will provide valuable feedback, which can either automatically causes essential set of exercises or provide the instructor with additional information about trainee’s progress.

Present work demonstrates only first steps of the further research of SK organization and its appliance to current operators’ training process. However, first outcomes assure us that chosen set of measures do specify qualities of the knowledge structure and can be used in future studies.

REFERENCES

Dozortsev V., Computer-based Training Systems for Technological Processes Operators)". Moscow: Sinteg. 2009. (in Russian)

Dozortsev V., Kreidlin Eu. "State-of-the-Art Automated Process Simulation Systems." *Automation and Remote Control*. 2010. Vol. 71. No. 9. Pp. 1955-1963.

Dozortsev V., Nazin V., Oboznov A., Gutsykova S., Mironova A. "Evaluation of the Process Operator Knowledge Formation Resulting from Computer-Based Training" *Proceedings of the ECEC'2015-EUROMEDIA'2015-FUBUTEC'2015 Conference*. Lisbon, Portugal. April 27-29, 2015. Pp. 118-123.

Johnsen S., Kilskar S., Fossum K. "Missing focus on Human Factors – organizational and cognitive ergonomics – in the safety management for the petroleum industry." *Journal of Risk and Reliability*. 2017. Vol 231(4). Pp. 400-410.

Meyer B. “The effects of computer-elicited structural and group knowledge on complex problem solving performance” Ph.D. Thesis, TU Berlin. 2008.

Mironova A. "Methods of estimation of Structural Knowledge and its application to operators’ training." *Automation in Industry*. 2018. Vol. 6. Pp. 24-31. (in Russian)

Petrenko V. "Meaning as a unit of conscience". 1993. *Journal of Russian and East-European Psycholog*y. Vol. 2. Pp. 3–29.

Schvaneveldt R. "PathFinder Associative Networks: Studies in Knowledge Organization." New-Jersey: ABLEX Publishing Corporation. 1990.

U.S. Chemical Safety and Hazard Investigation Board. "Drilling rig explosion and fire at the Macondo well. Investigation report volume 3" *Report no. 2010-10-I-OS.* 20 April 2016. Washington, DC: U.S Chemical Safety and Hazard Investigation Board.

AUTHORS BIOGRAPHIES

VICTOR M. DOZORTSEV was educated in Russia and got Ph.D. degree in control engineering in 1987. He received his doctorate (Dr. Sc.) after defending in 1999 the thesis dedicated to design, development and application of operator training simulators. In 1976 – 2005 he worked in the Institute of Control Sciences of Russian Academy of Sciences (Moscow, Russia) as chief of a research group. In 2005 he joined the Russian branch of Honeywell Corp. as Advanced Solutions Leader. His research interests include system analysis, mathematical modeling, advanced process control, cognitive engineering, computer-based training and decision-making support. He has professorship in some technical universities of Moscow.

ANASTASIA MIRONOVA was born in Moscow, Russia and studied Applied Physics and Mathematics in Moscow Institute and Physics and Technology. After getting her Master Degree she has been working at Honeywell Corp., where she dealt with technological processes simulation. After that she moved to Yandex for designing applications’ interfaces where she is working ever since. While working she is writing her PhD thesis on operators’ knowledge evaluation.

DECISION MANAGEMENT SIMULATION

Definition of Strategies based on Simulation and Design of Experiments

António Abreu*, José Requeijo**, J. M. F. Calado***, Ana Dias****
*ISEL- Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa,
Uninova Institute, Centre of Technology and Systems, Portugal
E-mail: ajfa@dem.isel.ipl.pt
**UNIDEMI/SEM, Systems Engineering and Management, Portugal
E-mail: jfgr@fct.unl.pt
*** ISEL- Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa,
IDMEC-LAETA-IST-UL Lisboa, Portugal
E-mail: jcalado@dem.isel.ipl.pt
**** ISEL- Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, Portugal
E-mail: asdias@dem.isel.pt

KEYWORDS
Design of Experiments (DOE), Simulation, Lean management, Strategic management.

ABSTRACT

Nowadays, even fairly small companies have to use more and more efficient working methodologies such as the "transnational" ones. The market may still be local or regional, but the competition is global. Companies, to be competitive, need to market their products at acceptable prices, in the right time and with a higher level of quality. According to some authors, the company's survival strategy involves the development of methodologies that are capable of designing, developing and / or improving the processes so that they are more efficient, with a low cost and in a shorter time. This article begins by discussing the relevance of the theme to the competitiveness of companies. Next, it is discussed how the Design of Experiments associated with simulation tools can contribute to the definition of operational strategies. Finally, an illustrative example is presented, and the main conclusions are drawn.

INTRODUCTION

Currently, at the macroeconomic level, it is possible to identify a set of variables that influence the operational strategies developed by managers, such as the energy crisis associated with the constant increase of oil prices and the emergence of new trading powers, as is the case of China and India, which have created new threats to the European industry (Martins et al. 2018). To be competitive, companies must develop capacities that enable them to respond quickly to market needs. Nowadays, it is possible to identify several variables that influence the development of production processes such as market pressures for quality improvement, reduction of production time and costs, and increased agility in production (Camarinha-Matos et al. 2004; Abreu and Calado 2017). On the other hand, the product life cycle is shorter and shorter, which means that the pace of design and/or development of new products is accelerating. The most frequent introduction in the market of new products with shorter time periods has been, in recent times, the survival strategy of some companies to win new customers and respond to the diversity of options available in the market. Therefore, the search for excellence in the creation of more efficient processes has become an operational survival strategy for companies competing in a globalized word (Abreu and Urze 2016; Abreu et al. 2018; Requeijo et al. 2014).

One of the goals of the twenty-first century organizations is to produce consistent products and robust to noise, i.e., the processes should present a good capacity and are aligned with the technical specifications and exhibit reduced variability. Reducing variability, i.e. reducing the common causes of variation due to the various sources of variation that are always present in the production processes, is basically achieved through two interventions. The first refers to more significant reductions, through the intervention of top management in the change of some of the resources allocated to the processes, such as new equipment, more qualified suppliers, more effective training of the organization's employees. A second, when it is not possible to resort to new financial resources, is to reduce some of the variability by optimizing the levels of factors (controllable variables) present in the productive processes.

These are the principles of the approach suggested in this paper, Design of Experiments (DOE) (Pereira and Requeijo 2012; Montgomery 2017; Taguchi 1986; Peace 1993). According to some authors (Harrison et al. 2007; Altiok and Melamed 2010; Klimov and Merkureyv 2008), the increasing use of simulation techniques is related to the fact that through simulation it is possible to imitate the operation of the real system in detail and predict its behaviour even when new events are introduced in the model, answering questions such as: What-If.

In addition, through the simulation, it is also possible to perform analyses on systems that do not yet exist, which allows the development of more efficient systems before any physical alteration has been initiated.

This paper discusses the advantages of DOE in articulation with simulation tools, to identify the most important variables in relation to a given objective and how they interact with one another and with the other elements of the system.

FUNDAMENTALS OF DESIGN OF EXPERIMENTS

The Design of Experiments (DOE) is a methodology that allows us to select the best combination of factors levels in order to optimize the value (s) of the quality characteristic (s) under study, both in terms of its mean value such as the reduction of variability, i.e., allows to determine which controllable factors affect certain quality characteristics and which are the best levels of these factors in order to increase the resistance of the product to the noise factors, thus satisfying the requirements of the various stakeholders in the performance of an organization.

Any Design of Experiments requires, prior to its implementation, a systematic approach so that its implementation leads to positive results. Generally, such an approach should include the following points [7,8]: Constitution of the research team; Clear definition of the objectives of the experiments; Background analysis; Selection of answers, i.e., choice of quality features as well as the respective measurement methods; Selection of controllable factors to be tested and their levels and, if feasible, choice of noise factors and their corresponding levels; Prior analysis of possible interactions between factors, understanding that there is an interaction between two factors when the effect of one factor on the response depends on the level of the other factor; Identification of the factors that will remain constant during the experimentation; Definition of the number of experiments to be performed and, depending on this decision and the number of factors and levels, plan the experiment using the most appropriate matrix, i.e. define the experimental layout; Definition the number of replications (number of times that a particular experiment is repeated); Execution of the experiments (tests) randomly.

The choice of the design of experiment is a function of the number of factors and the levels of those factors. Usually, fractioned planning is used, i.e. design in which only part of the total possible combinations is studied, which allows a significant cost reduction. In this article, only a full factorial design is presented, with two-level factors, referred to as 2k, where k represents the total number of factors.

The 2k Factorial Design

The full factorial design includes all possible combinations of the factors' levels. The simplest factorial design is one in which each factor is studied only at two levels. The generic representation of this type of design is 2k, where 2 is the number of levels of each factor and k is the number of factors included in the design. A complete factorial with two factors, A and B, where each is tested at two levels, will therefore be a 22, which requires 4 experiments to study all possible combinations of levels of the two factors.

The planning matrix of this design is shown in Table 1. To identify which effects, factors and interaction, influence significantly the response, is used the statistical technique ANOVA (Analysis of Variance). For this, it is necessary to determine the variations due to each factor/interaction related with the corresponding application of the proposed approach.

The variation of factor X (SS - Sum of squares) is determined as follows:

Table 1: 2^2 Factorial Design Planning Matrix

Standard Order	*A*	*B*	Response
(1)	–	–	y_{11} K y_{1n}
a	+	–	y_{21} K y_{2n}
b	–	+	y_{31} K y_{3n}
ab	+	+	y_{41} K y_{4n}

$$SS_X = \frac{[(\sum y)_{X^+} - (\sum y)_{X^-}]^2}{2^k n} \quad (1)$$

Where: n-number of replicates

The effect of the factor (the mean effect of a factor is the change in response caused by a change in the level of that factor) is given by:

$$\text{Effect of Factor } X = \frac{(\sum y)_{X^+} - (\sum y)_{X^-}}{2^{k-1} n} \quad (2)$$

The contrast of a factor or interaction is calculated from the response values of the various replicates of the experiment. For example, the contrast of the factor *A* is given by

$$\text{Contrast of } A = -\sum_{j=1}^{n} y_{1j} + \sum_{j=1}^{n} y_{2j} - \sum_{j=1}^{n} y_{3j} + \sum_{j=1}^{n} y_{4j} \quad (3)$$

The ANOVA table for the factorial design 2^2 is given as a function of the main factors and the interaction, as shown in Table 2.

Table 2 - ANOVA Table for 2^2 Factorial Design

Source of Variation	SS_X	*Deg. of Freedom* ν_X	*Mean Square* MS_X	F_0
A	SS_A	1	SS_A/ν_A	MS_A/MS_{Error}
B	SS_B	1	SS_B/ν_B	MS_B/MS_{Error}
AB	SS_{AB}	1	SS_{AB}/ν_{AB}	MS_{AB}/MS_{Error}
Error	SS_{Error}	$(2^k n-1)-3$	SS_{Error}/ν_{Error}	
Total	SS_T	$(2^k n-1)$		

The total variation (SS_T) and the residual variation are given as follows:

$$SS_T = \sum_{i=1}^{2^k} \sum_{j=1}^{n} y_{ij}^2 - \frac{\left(\sum_{i=1}^{2^k} \sum_{j=1}^{n} y_{ij}\right)^2}{2^k n} \quad (4)$$

$$SS_{Error} = SS_T - \sum SS_X \quad (5)$$

The best levels of factors are determined by analysing the mean response values, for the high level (+) and for the low level (–), of the significant factors/interaction. Thus, one should choose the level that optimizes the response, e.g., if the response (quality characteristic) is of the type the higher the better, the best level is one that maximizes the value of the responses. When studying the best combination of factors levels of a factorial design, it is convenient to analyse the response surfaces of the significant interactions.

DOE Assumptions

The validation of the analysis of the results of a design of experiments using the ANOVA application requires the verification of certain assumptions, without which all the analysis can be biased. In addition to applying the correct mathematical model, we analyse the errors, or residues, for each value of the experimentation. It should always be checked whether such assumptions are true, namely if the errors are independent and are normally distributed with null mean and constant variance σ^2. Residual analysis allows to make such confirmation. The residues are obtained by the difference between the values observed and the corresponding values expected or estimated by the model, defined by the following equation:

$$e_{ij} = y_{ij} - \hat{y}_{ij} = y_{ij} - \bar{Y}_{i.} \quad (6)$$

If the assumptions of Normality and homogeneity of variance are not verified, the original data should be transformed and the analysis of variance of the transformed data should be performed. There are several ways to proceed with this transformation, such as the empirical method and the transformation of Box and Cox [7,8].

Contribution of Factors

Taguchi [9,10] attaches great importance to the percentage contribution of a factor or an interaction to the Total Variation, a concept that is used in the analysis of variance as a complement to the test F. The evaluation of the significance of a certain factor X is carried out by the comparison between the Mean Square of the factor and the Residual Variance, that is:

$$\boldsymbol{F_0 = \frac{MS_X}{MS_{Error}} = \frac{SS_X}{(g.l.)_X \times MS_{Error}}} \quad (7)$$

If the factor *X* is not significant, its mean variation MS_X will be of the same order of magnitude of the residual variation. Thus, Taguchi defines the contribution percentage of a factor or an interaction by the following equation:

$$\boldsymbol{\rho_X = \frac{SS_X - (g.l.)_X \times MS_{Error}}{SS_T} \times 100} \quad (8)$$

In the above equation, ρ_X is the percentage contribution of the factor *X*, SS_X the variation of factor *X*, $(g.l.)_X$ the degrees of freedom of the factor *X*, MS_{Error} the residual variance and SS_T the total variation.

The contributions of significant factors are usually presented in the ANOVA table. The value of the error contribution is of practical interest, since a high value of the error may mean that there were problems in the application of the methodology, such as, for example, exclusion of potentially important factors from experimental planning, errors made during the execution of experiments or poor analysis of the results. After analysis of variance, the mean responses for each level of factors and / or significant interactions are examined and the levels leading to the pursuit of the previously defined objectives are chosen.

APPLICATION EXAMPLE

To illustrate the advantages of using DOE and simulation to understand the performance of the systems, as measured by one or more relevant variables, when one or more controllable variables or even non-controllable variables are observed, the case described in section 3.1 is considered.

Simulation scenario

A manufacturing cell consists of two serial workstations, a machining station and an inspection station. At the machining station, the semi-finished products are manufactured and sent to the inspection station. Transportation times between the two seasons are negligible. After the inspection the products are sent for shipment. However, it was found in the inspection process that 10% of the products did not meet the customer's requirements and returned to the machining centre for reprocessing.

Consider the following operational data: The products arrive at the machining station with a Takt Time of 1 minute/unit; The machining time follows a uniform distribution in which the minimum and maximum durations are UNIF (0.65; 0.70) minutes; The inspection time follows a uniform distribution in which the minimum and maximum durations are UNIF (0.75; 0.80) minutes. According to historical data the available maintenance times associated with the machining centre are as follows: MTBF – "Mean Time Between Failures", was estimated to be a random variable that can be described by an exponential distribution, EXPO (360), with a mean of 360 minutes; The equipment repair time, "Down time", was estimated as a random variable following a uniform distribution in which the minimum and maximum durations are UNIF (8; 12) minutes.

Due to the market pressure, the company needs to reduce lead time, i.e. the total time to carry out the machining and inspection process and thus increase its competitiveness. After several analyses, three improvement strategies are considered. According to the Maintenance perspective, with the introduction of a new maintenance management policy, there are strong expectations that it is possible to increase the time interval between equipment malfunctions, that is, it was defined as an expected result as a random variable which can be described by an exponential distribution with a mean of 400 minutes and the equipment repair time, "Down time", was defined as an expected result as being a random variable that follows a uniform distribution in which the minimum and maximum duration is UNIF (6; 9) minutes.

In terms of quality, through the implementation of a quality improvement program the expectations would be to obtain a reduction in the rejected product rate of 5% and thus reduce "Lead Time". Finally, in the production perspective, through better production management it would be possible to obtain significant improvements in the production / inspection process in relation to the current scenario. The machining time would then follow a uniform distribution in which the minimum and maximum durations are (0.55; 0.60) minutes and the inspection time would then follow a uniform distribution in which the minimum and maximum durations are (0, 60; 0.70) minutes.

Thus, to identify the improvement actions that should be implemented, and its degree of urgency, that allow to reduce

the “Lead Time”, a simulation model was built in the Arena software (www.arenasimulation.com) as illustrated in Figure 1 and experiment planning was applied, where five control variables (factors) were considered as shown in Table 3 to analyse their impact on the response variable "Lead Time".

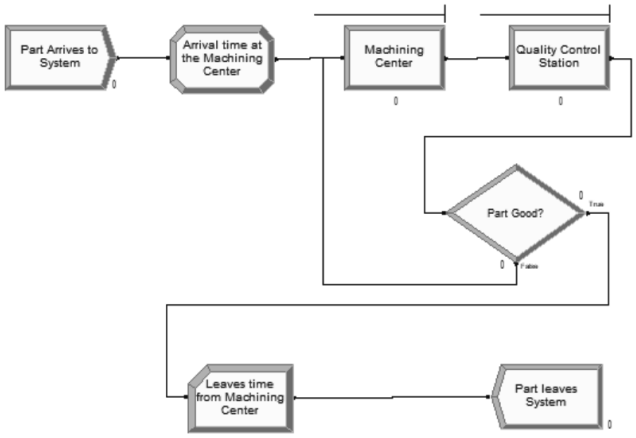


Figure 1: Arena Simulation Model

Table 3: Factors to Consider in the Analysis of the Response Variable "Lead Time"

FACTOR	DESCRIPTION	CURRENT LEVEL (-1)	PROPOSED LEVEL (+1)
A	Time between equipment malfunctions	EXPO (360) min	EXPO (400) min
B	Equipment repair time	UNIF (8,12) min	UNIF (6, 9) min
C	Rejected products	10%	5%
D	Processing time	UNIF (0.65; 0.70) min	UNIF (0.55; 0.60) min
E	Inspection time	UNIF (0.70; 0.80) min	UNIF (0.60; 0.70) min

The simulation time used was 160 hours (approximately 20 days of production) with a warm-up period to ensure the system is loaded and three replications were performed for each experiment.

Analysis of results

The value for each replica of the experiments was obtained by simulation, as mentioned in section 3.1. To find the best combination of levels of the factors that improve the response (Lead time) the ANOVA statistical technique was applied. Thus, the first step consists of the construction of the condensed ANOVA and to verify the assumptions of the ANOVA.

Since the Normality and homogeneity of variance are not verified, the experimental data had to be transformed using the Box and Cox transformation. Statistica software was used to perform this task. Table 4 shows the condensed ANOVA of the complete factorial 25 of the transformed data, referring to “Lead Time”.

The best levels of factors are determined taking into account the significant effects. The mean values of the transformed data of the significant factors are presented in Table 5 for the low (- 1) and high (+ 1) levels. Figure 2 shows the response surfaces of the significant interactions.

Analysing the results presented in Table 5 and Figure 2, Table 6 shows the best levels of factors for the study performed.

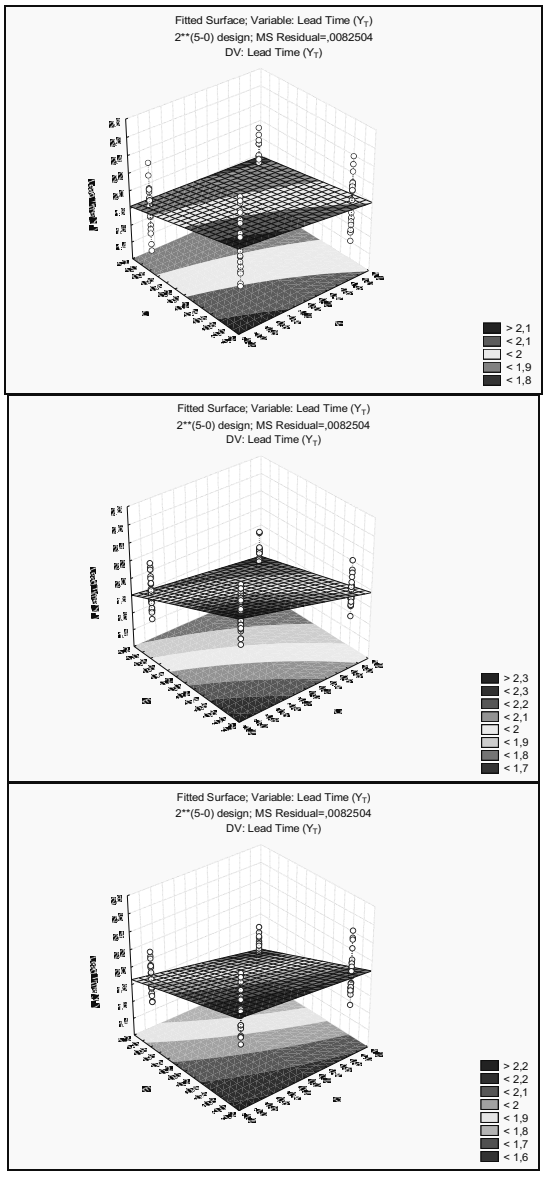


Figure 2: Response Surfaces of Significant Interactions for “Lead Time” (Factorial 2^5) – AC, CE and DE

Table 4: Condensed ANOVA of Factorial 2^5 for “Lead Time”

	ANOVA; Var.:Lead Time (Y_T); R-sqr=0.90573; Adj:0.89706 (Full Factorial 2^5 (n=3)) 2**(5-0) design; MS Residual=,0082504 DV: Lead Time (Y_T)					
Variation source	SS_X	ν_X	MS_X	F_0	p	ρ (%)
A	0.189510	1	0.189510	22.9697	0.000007	2.38%
B	1.067038	1	1.067038	129.3312	0.000000	13.91%
C	1.346961	1	1.346961	163.2594	0.000000	17.58%
D	0.775235	1	0.775235	93.9630	0.000000	10.07%
E	3.305693	1	3.305693	400.6691	0.000000	43.31%
AC	0.043639	1	0.043639	5.2893	0.023851	0.46%
CE	0.114611	1	0.114611	13.8915	0.000344	1.40%
DE	0.053526	1	0.053526	6.4876	0.012621	0.59%
Error	0.717787	87	0.008250			10.30%
Total	7.613999	95				100%

Table 5: Mean Values of Transformed Data for "Lead Time" (Factorial 2^5)

Factor	A	B	C	D	E
Level -1	1.9836	2.04459	2.05762	2.0290	2.12473
Level +1	1.8947	1.83374	1.82071	1.8493	1.75360

Table 6: Best Levels of the Factors for the "Lead Time" (Factorial 2^5)

A	B	C	D	E
1	1	1	1	1

CONCLUSIONS

Companies must be lucid, otherwise it can be fatal today, that in order to be competitive the strategy of survival must be based on methodologies that are capable of identifying the actions that must be carried out in order to provide, through efficient processes, innovative products and high quality. Thus, the definition of operational strategies based on techniques such as those presented in this article is more rigorous, avoiding in this way the subjectivity of analysis.

In this context, it was discussed how the experiments planning supported in simulation processes can be used in the definition of operational strategies in an organizational context.

However, one aspect that may compromise the robustness of the improvements resulting from the implementation of the operational plan, once designed, is related to the initial improvement expectations that would be expected to obtain (future state) in each of the actions (factors) identified.

ACKNOWLEDGEMENTS

This work was partially supported by FCT, through IDMEC, under LAETA, project UID/EMS/50022/2019.

REFERENCES

Abreu, A. and J. M. F. Calado 2017. "A fuzzy logic model to evaluate the lean level of an organization". *International Journal of Artificial Intelligence and Applications (IJAIA)*, Vol. 8, No. 5, 59-75.

Abreu, A. and P. Urze 2016. "System thinking shaping innovation ecosystems". *Open Engineering*, 6 (1), 418-425.

Abreu, A.; J. Requeijo; J. M. F. Calado; and A. Dias 2018. "Control Charts to Support Trust Monitoring in Dynamic Logistics Networks". In *Collaborative Networks of Cognitive Systems (PRO-VE 2018)*, L. M. Camarinha-Matos, H. Afsarmanesh and Y. Rezgui (Eds.). IFIP Advances in Information and Communication Technology, Vol. 534, Springer Nature Switzerland AG 2018, 499–511.

Altiok, T. and B. Melamed 2010. *Simulation Modelling and Analysis with Arena*. Elsevier.

Camarinha-Matos, L. M.; H. Afsarmanesh; and A. Abreu 2004. "Targeting major new trends". In *Collaborative Networked Organizations: A research agenda for emerging business models*, L. M. Camarinha-Matos and H. Afsarmanesh (Eds.). Kluwer Academic Publishers. ISBN: 978-1-4020-7823-1.

Harrison, J. R.; Z. Lin; G. R. Carroll; and K. M. Carley 2007. "Simulation modelling in organizational and management research". *Academy of Management Review*, 32(4), 1229-1245.

Klimov, R. and Y. Merkuryev 2008. "Simulation model for supply chain reliability evaluation". *Technological and Economic Development of Economy*, 14(3), 300-311.

Martins, José; A. Abreu; and João Calado 2018. "The need to develop a culture of innovation in a globalization context". In *Globalization*, George Yungchih Wang (Ed.). IntechOpen, ISBN 978-953-51-6088-5, DOI: 10.5772/intechopen.81266, 1-17.

Montgomery, D. C. 2017. *Design and analysis of experiments*. 9th Edition, John Wiley & Sons, ISBN: 978-1-119-11347-8.

Peace, G. S. 1993. *Taguchi Methods: A Hands-On Approach to Quality Engineering*. Addison-Wesley Publishing Company, New York.

Pereira, Z. L. and J. G. Requeijo 2012. *Qualidade: Planeamento e Controlo Estatístico de Processos*. Fundação da FCT-UNL, Lisboa.

Requeijo, José; A. Abreu; and Ana Matosa 2014. "Statistical Process Control for a Limited Amount of Data". In *Proceedings of ICORES 2014 – 3rd International Conference on Operations Research and Enterprise Systems* (Angers, France, March 6-8). ESEO, 190-195.

StatSoft, Inc. 2013. *Electronic Statistics Textbook*. Tulsa, OK: StatSoft, http://www.statsoft.com/textbook/.

Taguchi, G. 1986. *Introduction to Quality Engineering*. UNIPUB, White Plains, New York.

WEB REFERENCES

www.arenasimulation.com

BIOGRAPHIES

ANTÓNIO ABREU (born, 1967). He concluded his PhD in 2007 in Industrial Engineering at the New University of Lisbon (Lisbon, Portugal) and he is currently professor of Industrial Engineering in the Polytechnic Institute of Lisbon – Instituto Superior de Engenharia de Lisboa (ISEL), where he now holds assistant professor position. He is member of several national and international associations, e.g. he is co-founder of SOCOLNET (Society of Collaborative Networks), member of IFAC – International Federation of Automatic Control at TC5.3 Enterprise Integration & Networking, member of ISO/TC 258 - Project, programme and portfolio management, and member of INSTICC – Institute for Systems and Technologies of Information, Control and Communication. As researcher, he has been involved in several European research projects, organization and program committees of several national and international conferences. He has more than 100 publications in books, journals and conferences proceedings in the area of collaborative networked organisations, logistics, project management, quality management, open-innovation, lean management. (ajfa@dem.isel.ipl.pt; https://www.isel.pt/docentes/antonio-joao-pina-da-costa-feliciano-abreu).

JOSÉ REQUEIJO has received a Ph.D. in Industrial Engineering from the New University of Lisbon – Portugal (UNL). He is now an Assistant Professor at the Department of Mechanical and Industrial Engineering of the FCT-UNL. He has extensive teaching experience in several areas of Quality Engineering. He is an effective member of the UNIDEMI research center. As a researcher, he has participated in research projects, as well as in industrial engineering congresses (Quality area). He is the author of papers in international journals and conferences. (jfgr@fct.unl.pt; https://www.demi.fct.unl.pt/pessoas/docentes/jose-fernando-gomes-requeijo).

J. M. F. CALADO (born, 1962). He got the PhD degree in Control Engineering at the City University (London, United Kingdom) and did the Habilition Exam in Electrical Engineering at the Universidade da Beira Interior (Covilhã, Portugal) in 2009. Since 1998, he has been with the Polytechnic Institute of Lisbon – Instituto Superior de Engenharia de Lisboa (ISEL) in a permanent position as Associate Professor and since May 2009 he became Full Professor in the Mechanical Engineering Department – Control Systems Group. He is currently the Head of the Mechanical Engineering Department and the Coordinator of the MSc in Maintenance Engineering and the Coordinator of the MSc in Engineering and Industrial Management. He is Member of IFAC Technical Committee on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS, Senior Member of IEEE – Institute of Electrical and Electronic Engineers, Member of Society of Collaborative Networks (SoColNet), Member of The Portuguese Association for Automatic Control, Member of The Portuguese Society of Robotics and Fellow Member of the Portuguese Association of Qualified Engineers. He is author of more than 150 scientific publications including books' chapters, journal papers and papers in proceedings of national and international conferences and he has been serving as reviewer or several international journals and international conferences. (jcalado@dem.isel.ipl.pt; https://www.isel.pt/docentes/joao-manuel-ferreira-calado).

ANA DIAS was born in Lisbon, Portugal where she obtained her degree in Mechanical Engineering in 2001 by the Instituto Superior Técnico (IST), and then her PhD in Engineering and Industrial Management by the University of Beira Interior (UBI), Covilhã, Portugal in 2015. She is a former industrial collaborator in quality, occupational health and reception/testing of equipment, from 2001 to 2008. Since then she is Professor at Instituto Superior de Engenharia de Lisboa (ISEL). (asdias@dem.isel.pt; https://www.isel.pt/docentes/ana-sofia-martins-da-eira-dias).

SITUATIONAL INFORMATION SYSTEM FOR DECISION-MAKING SUPPORT REGARDING INDUSTRY-NATURAL COMPLEXES

Alexander Fridman and Andrey Oleynik
Institute for Informatics and Mathematical Modelling
Kola Science Centre of the Russian Academy of Sciences
24A Fersman str., 184209, Apatity Murmansk reg.,
Russia
E-mail: fridman@iimm.ru

KEYWORDS
industrial-natural complex, decision-making support, subject domain conceptual model, situational modelling, cognitive categorization of situations, decisions coordination.

ABSTRACT

This paper introduces a superstruction upon a geographic information system (GIS) built to create a situational information system (SIS) aimed at comparative static and dynamic analysis of user-formed alternative structures for complex hierarchical spatial objects, namely industry-natural complexes (INCs). The main distinguishing features of SIS are in supporting a domain model open for operative modification, combining different forms of representation for both data and expert knowledge as well as in cognitive processing of situations and development of intelligent coordination procedures and planning of interactions among several decision makers responsible for functioning of INC's components. A hierarchical subject domain conceptual model (SDCM) is used as the heart of SIS to maintain every stage of modelling. This model provides a formal definition of situations occurred within the object under investigation, their categorization and synthesis, thus maintaining multicriteria analysis of decisions on changing/preserving the current INC's structure. Unified processing of data coming from simulation modules of INC's structural components, embedded GIS and expert system (ES) constitutes the main difference between SIS and its prototypes.
Such a system ought to become core software for decision making in modern situation centres (SCs) intended for spatial modelling and forecasting at different organisational levels.

BASICS OF THE SDCM

Quite a few years ago (Fridman et al. 2004) we introduced our integrated SDCM to the audience of this conference. In this article, we would like to remind foundations of the developed situational approach and brief our advances in using SDCM to model INCs.
SDCM comprises elements of a subject domain and relations among them formalising object structures and cause-effect links, which are essential within frames of the current study of an INC. Three types of elements are allowed in the model: *objects* (components of the INC), *processes* and *resources* (data). At forming the INC conceptual model, some alternate structures of the object under investigation are input into the model by using OR-relations in objects decomposition and resources generation. A specific extension of the situational approach (Pospelov 1986) on the problem of INC investigation based on the SDCM was proposed to compare alternative object trees (Fridman et al. 1998).
In the SIS, *objects* form a hierarchy, reflecting the organizational links between them. In the course of building a model, objects are associated with an electronic map (GIS) in such a way as to ensure a one-to-one correspondence between the conceptual and geographical representation of the INC. Relations among model elements are formalized by using *resources* that simulate material and informational interactions among these elements and represent time series of data. *Processes* describe the nature of resources conversions within model elements. In order to implement multi-model development, a number of types of processes are allowed. Currently, these are calculated modules of a fairly arbitrary structure (generally described by difference equations), user-defined functions or sets of production rules stored and processed in a specialized ES of the SIS. The type of a process is set by assigning to it an executor who defines the computer implementation of the process. Executors are also assigned to resources external to the INC under study, their values in relation to time can be generated by using user-defined functions, external databases or ES rules.
Technologically, the SIS is designed to model organizational systems in the paradigm of the structural approach to building an open domain model. It has been proven (Fridman et al. 1998) that the SIS allows to investigate stratified, multilevel and organizational hierarchies, i.e. all main types of hierarchies considered in (Mesarovic et al. 1970).
Main ideas and features of the proposed approach to situational modelling of INCs are described in detail in a number of our publications (Fridman 2016, 2017, 2018, and others referred to here). In this article, they will be overviewed at a qualitative level, without mathematical formulations. Mathematical structures of the SIS's model elements and subsystems that are necessary for its studying, as well as details and proofs of the statements presented below are contained in the referred publications. The most complete description of the SIS can be found in (Fridman 2015).

SYNOPSIS OF SITUATIONAL MODELLING

To reduce complexity of a model, decomposition techniques are often used to split modelling tasks into several simpler subtasks. In organizational systems (INCs belong to this class), the model usually becomes hierarchical. Set-theoretical techniques to investigate such models within the

traditional optimization approach started by fundamental works of M. Mesarovic with colleagues and resulted in appearance of General Systems Theory (Mesarovic and Takahara 1975) and Theory of Hierarchical Multilevel Systems (Mesarovic et al. 1970) where the researched object is modelled by a formal system. An alternative approach called Systems Simulation is based on works of Shannon (Shannon 1975). One more direction to generalize the classical control theory was proposed by D.A. Pospelov (Pospelov 1986). In his situational control, the object under investigation is modelled by a set (a network, in general) of formal systems, and specific analysis of situations is carried out to choose one of these formal systems relevant to the current situation. However, the situational control gives no formal definition of a situation and does not consider the hierarchical description of the object. Another technique to describe hierarchies was grounded by conceptual design of databases after J. Martin and J. Ullman and developed to Structured Analysis (SADT) by D. Ross; see, for instance (Dickover et al. 1977). In SADT, the model of an object is designed by its functional decomposition upon one or several parameters. Nowadays, the mentioned approaches have originated numerous theories and models including UML, Situational Method Engineering (Henderson-Sellers and Ralyté 2010; Gericke, et al. 2009), Situation Awareness Analysis and Measurement (Endsley 2015), DF&RM and JDL models (Reiter 2001), Situation Calculus (Steinberg 2009), etc. Unlike these theories and models dealing with general problems and techniques of data fusion and processing, the described below situational information system is focused on software support of HSOs situational analysis and simulation. So, the terminology we use further on is applicable to this area only. Details can be found in our papers included in the list of references below.

The situational approach to studying an INC is implemented in the SIS as follows.

An elementary form of representing information in the SIS is a *fact* containing information about the current values of a certain resource. An *initial situation* is entered by a user as a finite set of facts. The SIS ES analyzes the SDCM and the initial situation and extends the latter to the *complete situation*. Each complete situation is displayed by a connected fragment of the SDCM, which may include alternative variants of possible structures for the INS under study. A *sufficient situation* is one of admissible (comprising no alternatives) options for implementation of the corresponding complete situation. Sufficient situations can be compared with each other statically according to specified quality criteria (as a result of classifying situations) and dynamically by simulation. A *scenario* is a sequence of sufficient situations generated from the same complete situation; it sets a specific modelling variant.

Thus, the SIS integrates different forms of knowledge representation (calculated data, graphic characteristics of INC elements, expert knowledge) and solves the problem to research and predict behaviour of INCs as multilevel multicomponent spatial objects.

Classification and Generalization of Situations

To classify situations in the pilot version of the SIS (Fridman et al. 2004; Fridman et al. 1998), we proposed a numerical *generalized performance criterion* (GPC) for any element of an INC as a weighted sum of scalar (partial) criteria with importance coefficients that are inversely proportional to tolerances for deviations of the particular criteria from their nominal value, which corresponds to the common sense: the more important is a scalar criterion for a decision maker (DM), the smaller is the tolerance advisable to assign to the deviations of this criterion. Advantages of the proposed generalized criterion are as follows: it is equal to unity when the values of all its arguments (partial criteria) are on the verge of tolerances and do not exceed unity if all arguments lie within the tolerances; for its formation, the DM must specify only semantically understandable variables, namely nominal values of partial criteria and tolerances for their deviations; to find the root cause of inefficient functioning of the INC, it is sufficient to identify the farthest objects in the object hierarchy (from the root object of the studied model fragment; in the SIS such an object is called the *object of decision-making*, ODM, since in fact a DM is situated on this object), which has a GPC value significantly higher than unity. Calculating the values of this criterion up the tree of objects from leaf objects to the ODM allows for an unambiguous classification of sufficient situations by assigning to one class those in which the same partial criterion makes the minimal contribution into the generalized criterion, that is, this partial criterion is satisfied to the maximum degree. Within one class, it is natural to consider a sufficient situation better that another one, if its value of the generalized criterion is smaller.

However, if it is necessary to transfer an INC to another class of situations, it is required to offer the DM a structure from the new class, which is minimally different from the current situation, so as not to introduce unnecessary disturbances into the real system, but still close enough to the best structure of the desired class. The generalized criterion presented above is not directly suitable for solving such a problem, since its values depend not only on the structure of the situation, but also on the values of the resources. Therefore, in (Fridman 2018) a cognitive method for classifying situations in the SIS was developed, that depends less on the values of resources. It is based on a new semantic metric of proximity of situations. This metric uses the Tversky's ratio model (TRM, Tversky 1977) at each level of the object hierarchy, and the hierarchy of the situations description is taken into account by normalizing the values of the TRM with weighting factors inversely proportional to the level number. By varying values of these factors, we can consider expert estimations for importance of the levels, but in the general case, obviously, the contribution of lower level objects into the proposed new hierarchical Tversky's model (HTM) should decrease with increasing level numbers.

With the help of the HTM, it is also possible to generalize descriptions of situations in the SIS. Such generalization includes two main stages: the search for common signs of situations that fall into one class for each studied SDCM fragment, and the search for occurrences of situations in situations of higher levels (the level number here is set by the level of the ODM), which is done by means of Graph Matching (for example, (Sambhoos et al. 2010)). The results of the generalization are formed in the form of rules in the SIS ES (Fridman et al. 2004; Fridman et al. 1998), which include both positive and negative examples.

Then, effectiveness of alternatives provided in the SDCM is evaluated. The wider is the set of classes of situations comprising various options of a certain alternative, the more effective is this alternative. Then the alternatives are ordered by this criterion, and a DM is primarily presented to evaluate the most effective alternatives. The opposite statement is also true: an alternative is ineffective for a given ODA, if none of the available choices translates sufficient situations into another class. In the course of generalizing situations, it is advisable to identify in advance the set of properties of the most effective alternatives. The results of evaluation of effectiveness for different alternatives are formalized in the form of rules of the SIS ES (in situational control, they are referred to as logical-transformational rules (Pospelov 1986) and govern the process of classifying situations).

Coordination and Planning of Interactions among Components of an INC

As already noted, the problem of coordination arises when there are several DMs responsible for the functioning of the various components of one INC. To solve it, it is necessary to find conditions under which all DMs reach their local goals and at the same time solve the top-level task, for the solution of which the entire system was created. Traditionally, this problem is considered for a two-level system, since the possibility of generalization to a larger number of levels is obvious (Mesarovic et al. 1970). Coordination has two aspects: coordination of tasks of lower-level DMs relative to the task of the Coordinator (the upper-level DM) and coordination with respect to the global goal. Based on the principle of predicting interactions (Mesarovic et al. 1970) for the SIS, we found the necessary co-ordination conditions for both these aspects by analyzing gradients (increments for the discrete formulation of the problem) of the above-presented GPC on nominal values of partial criteria (Fridman and Fridman 2010a, 2010b, 2012, 2013). In (Fridman 2016, 2017) the same approach was used to coordinate interactions within a collective of intelligent dynamic systems (IDSs, Osipov 2010). IDSs are fundamentally different from “classical” dynamic systems in that they are implemented using databases and knowledge bases and allow for presence of non-numeric components in the state vector. However, the principle of predicting interactions turned out to be suitable in this case too. Instead of the criterion gradients, it is necessary to analyze the inclusion ratio for state vectors of the involved IDSs.
Using the same approach and the concept of N-attainability (Osipov 2010), it became possible to solve the planning problem for both the SIS and an IDSs’ collective, that is, to propose algorithms for choosing controls that transfer the system from an initial state to the target one (Fridman 2016). Planning is a necessary step prior to any control.

Situational Modelling and SCs

Creating a situation centre today is a necessary condition for improving the effectiveness of management activities at different organisational levels starting from corporations and state-owned companies and finishing at the level of countries or their unions, e.g., EU SITCEN, NATO SITCEN. SCs are complex organisational and technical structures based on information-analytical systems, monitoring tools, visualisation and display means as well as on predictive models and patterns of development. Their ultimate goals are as follows: increasing efficiency and quality of management decisions, preventing emergencies or minimising losses due to prompt and proper addressing of compensatory measures.
Russia and its predecessor U.S.S.R. also had a fair experience in projecting and using SCc (for instance, see (Larichev 1994; Gelovani et al. 1980; Gelovani and Dubovsky 1990); a good review is available in (Zatsarinny and Shabanov 2015)).
The perennial problem of modern SCs seems to result from their uniqueness and software incompatibility which prevents automation and coordination of their actions.
In our opinion, this problem can be handled by structural unification of models used in various SCc, and the above-introduced SIS provides a fair basis for such unification. Naturally, different secrecy aspects are tightly related to this task, but there exist a lot of areas, e.g., logistics, technical safety, etc., that can be tackled apart from confident data.
Lately, our research group conducts a project funded by the Russian Foundation for Basic Researches (grant 18-29-03022 “Development of Techniques and Technologies for Intelligent Situational Analysis to Provide Information and Analytical Support of a Network of Cognitive Situation Centers in the Arctic Zone of Russian Federation”). The project is aimed at developing technologies and methods of intellectual analysis of situations for information and analytical support of the system (network) of cognitive situational centres within the Arctic zone of the Russian Federation (Oleynik et al. 2017, 2018). The relevance of the research is determined by the need to include in the composition of computer analysis and predictive modelling tools used in situational centres, tools for operational cognitive classification of situations and automated selection of predictive models that are most appropriate for a particular class of situations. Classification is proposed to be carried out within the framework of a generalized interdisciplinary conceptual space (for example, see (Rosch 1973; Gardenfors and Lohndorf 2013; Decock and Douven 2014)) based on quality dimensions that characterize complex modelling objects from the point of view of various subject domains. The project shall result in creating methods for formation of an interdisciplinary conceptual space and the technology of its use for situational modelling, both for rapid assessment of the current state and forecasting development options for complex spatial systems in the Russian Arctic. A prototype of a single conceptual space and methods to assess situational awareness will be developed using the example of a generalized space of situational modelling of industrial-natural systems and freshwater ecosystems of the Russian Arctic, taking into account their mutual influence and risks of non-standard situations caused by large-scale atmospheric phenomena that are characteristic for these territories. The scientific value of this result consists in intellectualization of systems for modelling complex objects in a dynamic environment in order to increase the information security and efficiency of decision-making on managing objects in various conditions.
At present, the concept of situation(al) awareness (SA), see, for example, (Endsley 2015), describes the most general principles of preparing and processing information for implementing a situational approach in dynamic subject domains. Inadequate or incorrect SA is considered to be one

of the main factors associated with accidents caused by the "human factor". This is especially evident in highly dynamic subject domains, however, to our mind, it ought to be taken into account also for the situational modelling of dynamic hierarchical spatial complexes of the INC's class we consider here, where the period of decision-making is long enough, but other aspects of SA are quite significant.
The formal definition of SA is divided into three segments: perception of elements in the environment, comprehending of the situation and predicting future states of the system.
Main principles of SA proposed by the author of this approach, M. Endsley (for instance, (Endsley 2015; Lundberg 2015)), are specified in the SIS as follows:
perception is modelled by setting an initial situation, the comprehension is defined by formation of the relevant complete situation and determination of the organizational level due to solve the current problem, prediction is implemented by simulation experiments;
the purpose of a simulation is set by the DM by selecting a desired class of situations during the simulation;
specificity of the available information is taken into account by choosing an initial situation and automatic rejection of unpromising alternatives;
DMs' preferences and expectations are formalized by selection of the dominant partial criterion in the corresponding GPC and by composition of permissible alternatives stipulated in the INC model at its creation.
For quantitative assessment of SA in the SMS, we propose to use ratios between set powers of input resources of the area of responsibility of a DM, which change without her/his participation, output resources controlled by this DM, and their share in the total number of resources essential for functioning of the entire INC. These relations allow an objective assessment of importance for decisions of this DM and take into account this importance when searching for a balance of interests of all DMs affecting characteristics of the INC. The introduced ideas will be detailed and published in near future.

CONCLUSION

The article describes the results obtained in the field of development of methods and technology of situational modelling of complex spatial dynamic systems, which, in particular, include industrial-natural complexes. The basic principles of the situational modelling system, focused on software support for situational analysis and modelling of these systems, are outlined. Compared to the previous publication (Fridman et al. 2004), the following aspects of the novelty of the completed developments are briefly presented. Cognitive methods of generalization and criteria for classifying situations as well as conditions for their applicability are described. It is noted that the cognitive method of situations classification, based on the proposed semantic hierarchical metric of proximity of situations, allows to take into account expert assessments of importance for levels of the hierarchical model of an industrial-natural complex. Two aspects of tasks coordination solved by various decision makers in their managed parts of an industrial-natural complex, and a way to assess their situational awareness are considered.

FUTURE RESEARCH

Generally speaking, meanwhile we see two ways to use the SIS.
First way is in designing a general SIS to study a certain area for different goals. The number of goals and the space of the area here is theoretically limited with achievable data only. The more goals are touched in the general SIS, the better, because it is not easy to foresee importance of certain factors for this or that goal. Applicability of the SIS in this implementation is substantiated with its natural capability to accumulate different parameters and functions of an INC by their spatial binding. This general SIS could provide solving the following tasks:

- exploration of operation laws, interconnections and importance of INC parameters for situational modelling of the INC and its subsystems;
- generation of "rapid prototypes" for commercial decision making support systems (DMSSs);
- creation an intelligent environment for integrating and/or separating different applications in regional management;
- coordination of interactions among DMs responsible for different parts of an INC and planning the behaviour of the entire system.

Such a general SIS would objectify decision making, decrease undesirable consequences of these decisions and ground proper measures for minimizing negative impacts to the INC under investigation.

The second way to use SISs seems to be in generation of specialized situational DMSSs by separating them from the general one. Within the SIS, that can be done fairly easy (Fridman 2015). These problem-oriented DMSSs would be capable to make decisions on a limited set of typical (repetitive) situations. Such DMSSs would provide a DM with an algorithmic support to ground altering (or not altering) the structure of her/his subordinated object and constitute an alternative for an expert council in this sense. Automated generation of specialized DMSSs would significantly decrease expenses for their creation and, which is more important, would objectify their efficiency within their domains since this efficiency could be preliminarily tested within the general SIS by simulating the metasystem for these domains.
In shorter perspective, we plan to conduct our researches in the following areas:

- synthesis of control strategies for INCs in a generalized state space including heterogeneous variables (numeric, symbolic, logical, fuzzy, etc.);
- situations processing for abductive tasks (what is necessary to do in order to obtain a certain result);
- revealing correlations between the metric of the GPC and results of situations classification;
- learning laws and connections between types of INC decomposition and sensitivity of the model;
- improving methods to estimate indices of INCs safety and reliability for underdetermined data;
- automation of building the SDCM by means of intelligence techniques (Artemieva and Fridman 2018; Artemieva et al. 2012);
- incorporating concepts of situation awareness and conceptual spaces into our model. Our goal is to obtain justifiable quantitative estimates of SA for every DM involved in management within one INC in order to find a

reasonable compromise among their interests and prevent possible conflicts.

Besides investigation of INCs, the introduced approach looks prospective for estimating SA in supply chain management and development of virtual enterprises.

ACKNOWLEDGEMENTS

This research was supported in part by the Russian Foundation for Basic Researches (grants 16-29-12901, 18-29-03022, 18-07-00132, and 18-01-00076).

REFERENCES

Artemieva, I. and A.Ya. Fridman. 2018. "Ontologies in the Automation Problem for Situational Modelling." In *2018 3rd Russian-Pacific Conference on Computer Technology and Applications* (RPC), 48-53.

Artemieva, I.; A. Zuenko and A. Fridman. 2012. "Algebraic Approach to Information Fusion in Ontology-Based Modelling Systems." *International Journal of Computing*, 11, No.1, 55-63.

Decock, L. and I. Douven. 2014. "What is Graded Membership?" *Noûs.* 48, 653-682.

Dickover, M.E.; C.L. McGowan; and D.T. Ross. 1977. "Software design using SADT." In *ACM Annual Conference*, 125-133.

Endsley, M.R. 2015. Final Reflections: "Situation Awareness Models and Measures". *Journal of Cognitive Engineering and Decision Making* 9, No.1, 101–111.

Fridman, A. and O. Fridman. 2010. "Gradient Coordination Technique for Controlling Hierarchical and Network Systems". *Systems Research Forum* 4, No.2, 121-136.

Fridman, A. 2016. "Planning and Coordination in Hierarchies of Intelligent Dynamic Systems". *TELKOMNIKA* 14, No.4, December, 1408-1416.

Fridman, A. 2018. "Cognitive Categorization in Hierarchical Systems under Situational Control". In *Advances in Intelligent Systems Research* 158, Atlantis Press, 43-50.

Fridman, A. and O. Fridman. 2010. "Incremental Coordination in Collaborative Networks". In *Proceedings of International Congress on Ultra Modern Telecommunications and Control* Systems (ICUMT-2010), October 18-20, Moscow, Russia, 649-654.

Fridman, A. and O. Fridman. 2012. "Combining Neural Networks and Incremental Techniques for Coordination in System of Systems". In *Cybernetics and Systems 2012: Proceedings of Twentieth European Meeting on Cybernetics and Systems Research* (EMCSR 2012). Vienna, Austria, 203-207.

Fridman, A.; A. Oleynik; and O. Fridman. 2004. "Knowledge Integrating in Situative Modelling System for Nature-Technical Complexes". In: *Proceedings of the 2004 European Simulation and Modelling Conference* (ESMc2004), Paris, France, October 25-27, 25-29.

Fridman, A.; A. Oleynik; and V. Putilov. 1998. "GIS-based Simulation System for State Diagnostics of Non-stationary Spatial Objects". In: *Proceedings of 12th European Simulation Multiconference* (ESM'98), Manchester, UK, June 16-18, 1998, vol.1, pp.146-150.

Fridman, A.Ya. 2015. *Situational structural control in industry-natural complexes. Methods and models*. Saarbrucken, Germany: LAP Lambert Academic Publishing. (In Russian).

Fridman, A.Ya. 2017. "SEMS-Based Control in Locally Organized Hierarchical Structures of Robots Collectives". In: *Smart Electromechanical Systems: The Central Nervous System*. A.E. Gorodetskiy and V.G. Kurbanov (Eds.) Series: Studies in Systems, Decision and Control 95, 31-50. 1st ed. Springer.

Fridman, O. and A. Fridman. 2013. "Decreasing Dissemination of Disturbances within Network Systems by Neural Networks". *TELKOMNIKA* 11, No.9, September, 4942-4948.

Gardenfors, P. and S. Lohndorf. 2013. "What is a domain? Dimensional structures versus meronomic relations". *Cognitive Linguistics* 24, No.3, 437-456.

Gelovani, V.A. and S.V. Dubovsky. 1990. "Global modelling of the potential world system". *International Political Science Review* 11, No.2, 207-218.

Gelovani, V.A.; V.B. Britkov; and V.V. Yurchenko. 1980. "An Interactive Modelling System for Analysis of Alternative Decisions". In *Decision Support Systems: Issues and Challenges. IIASA Proceedings* 11, 149-151.

Gericke, A. et al. 2009 "Situational method engineering for governance, risk and compliance information systems". In *4th Int. Conf. on Design Science Research in Information Systems and Technology*, article no. 24. New York, NY, USA: ACM Press.

Henderson-Sellers, B. and J. Ralyté. 2010. "Situational method engineering: state-of-the-art review". *Universal Computer Science* 16, No.3, 424-478.

Larichev, O.I. 1994. "Interactive Decision Support Systems for Top Decision Makers". In *Policy Analysis: Tools for Critical Choice by Top Decision Makers.* Y. Dror (Ed.). United Nations.

Lundberg, J. 2015. "Situation Awareness Systems, States and Processes: A holistic framework". in *Theoretical Issues in Ergonomics Science*.

Mesarovic, M. and Y. Takahara. 1975. *General Systems Theory: Mathematical Foundations*. Acad. Press, New-York, San Francisco and London.

Mesarovic, M.; D. Macko; and Y. Takahara. 1970. *Theory of Hierarchical Multilevel Systems*. Acad. Press, New-York and London.

Oleynik, A. et al. 2017. "Solutions for System Analysis and Information Support of the Various Activities in the Arctic". *Czech Polar Reports* 7, No.2, 271-279.

Oleynik, A., A. Fridman and A. Masloboev. 2018. "Informational and analytical support of the network of intelligent situational centers in Russian Arctic". In: *IT&MathAZ 2018 Information Technologies and Mathematical Modeling for Efficient Development of Arctic Zone. Proceedings of the International Research Workshop on Information Technologies and Mathematical Modeling for Efficient Development of Arctic Zone*. Yekaterinburg, Russia, April 19-21, 57-64.

Osipov, G. "Intelligent dynamic systems". 2010. *Scientific and Technical Information Processing* 37, No.5 (December), 259-264.

Pospelov, D.A. 1986. *Situational Control: Theory and Practice*. Batelle Memorial Institute, Columbus, OH.

Reiter, R. 2001. *Knowledge in action: logical foundations for specifying and implementing dynamical systems*. MIT Press.

Rosch, E.H. 1973. "Natural categories". *Cognitive Psychology* 4, No.3, 328-350.

Sambhoos, K.; R. Nagi; M. Sudit; and A. Stotz. 2010. "Enhancements to High Level Data Fusion using Graph Matching and State Space Search". *Information Fusion* 11, No.4, 351-364.

Shannon, R.E. 1975. *System simulation: the art and science*. Prentice-Hall.

Steinberg, A.N. 2009. "Foundations of situation and threat assessment". In *Handbook of Multisensor Data Fusion*, M.E. Liggins, D.L. Hall and J. Llinas (Eds.), chapter 18. London: CRC Press.

Tversky, A. "Features of similarity". 1977. *Psychological Review* 84, No.4, 327-352.

Zatsarinny, A.A. and A.P. Shabanov. 2015. *Technology of informational operation support of organizational systems on the basis of situation centres*. Torus Press, Moscow. (In Russian).

RELIABILITY AND QUALITY IN MANUFACTURING

William Conley
Austin E. Cofrin School of Business
University of Wisconsin at Green Bay
Green Bay, Wisconsin 54311-7001
U.S.A.
Conleyw@uwgb.edu

KEYWORDS
Statistical optimization, market planning, cost control, industrial engineering, profits, complex systems

ABSTRACT

Reliability and quality apply to many fields. Quality products that can be made within budget limitations where market research has determined there is a sufficient demand at a profitable price are all important. Also, the actual factory floor where the manufacturing takes place has to be organized and run (by industrial and other engineers and supervisors) so that the correct amounts of products are produced to meet customers demands in a timely fashion. Additionally, reliability will be addressed from the perspective of how many backup components should be available to take over if a primary component fails in a subsystem of the product which is being produced.

Three of these problems are dealt with here in a multivariate nonlinear setting and statistical optimization is used to guide their solutions.

INTRODUCTION

First, a market study in ten markets is done to estimate the demand for ten products a company may produce and statistical optimization (multi stage Monte Carlo optimization – MSMCO) stimulation is done to guide the company decision makers on how much (if any) to produce for these products.

Secondly, a specific quality in design of a product is looked at to optimize how many back-up components to include to have a reliable chance that a certain subsystem will work.

Lastly, a planning of production for a factory that produces 25 output products or components from 25 input items is studied and optimized to meet the 25 current product demands from the company's varied customers. Again, statistical optimization simulation will be used. The solution technique (MSMCO) will also be explained and illustrated.

MARKET RESEARCH

A company plans to make and sell its ten products in the coming time period. However, it wants to be pretty sure of what percentage of its potential customers will buy these products.

It would like to get each of the ten percentages to within 2% to 6% with 95% confidence and only spend $100,000 or less on the survey. It plans to conduct telephone interviews. The cost per interview for products one through ten are 25, 30, 18, 16, 10, 35, 27, 21, 14 and 28 dollars respectively.

The standard normal proportion formula $z = (p\square - p)/\sqrt{(pq)/n}$ can be rearranged and solved for a sample size formula of $n = \left(\frac{z}{2B}\right)^2$ where z is the appropriate standard normal value for the level of confidence desired and B is the bound (how close to the true proportion is required). Therefore, the company's optimization problem becomes

Minimize $f(B_1, B_2, \ldots B_{10}) = \sum_{1=1}^{10} B_i$ subject to $0 < B_i \leq .10$ for i =1, 2,… 10 and

$\sum_{i=1}^{10} C_i \left(\frac{z}{2Bi}\right)^2 \leq$ 100,000 dollars

with z = 1.96 from the standard normal distribution for the 95% goal and the C_i's are the ten interview costs just mentioned. Note that even though they don't want to go above 6% error they put the upper bound at .10 just in case something unusual happens. Therefore, a 20 stage multi stage Monte Carlo optimization (MSMCO) simulation drawing 1000 sets of 10 B_i's in each stage is done in the range 0 to .10.

The printout reals the sample sizes N_i and the estimation error B_i resulting from the phone calls

$B_1 = 0.0472$ $B_2 = 0.0507$
$N_1 = 431.7$ $N_2 = 373.8$
$B_3 = 0.0428$ $B_4 = 0.0411$
$N_3 = 523.4$ $N_4 = 567.9$
$B_5 = 0.0351$ $B_6 = 0.0538$
$N_5 = 777.9$ $N_6 = 331.3$
$B_7 = 0.0493$ $B_8 = 0.0452$
$N_7 = 395.1$ $N_8 = 469.6$
$B_9 = 0.0394$ $B_{10} = 0.0496$
$N_9 = 619.3$ $N_{10} = 389.7$

The average error was .04543 and the total cost was 100,000 dollars. Note that real value arithmetic was used so the ten sample sizes (the N_is) should probably be rounded to the nearest whole number of phone calls to make.

This might make the total cost off by a few dollars; however, the management officer should not mind given that this statistical study (like all surveys) will have a little bit of error in them anyway.

RELIABILITY

Company two makes a number of subassemblies for products where reliability is an issue. The particular one they are working on now has 14 components all arranged in a series. Therefore, if any one of them fails the whole subassembly will not work.

The failure probabilities for each of the 14 components are .015i for i = 1, 2, . . . 14 and their costs are 14-.5i dollars for i = 1, 2, . . . 14.

Therefore, the question is how many backup components should be attached to each of the 14 components so that the system has at least a 90% chance of working and the total cost is minimized. Therefore, their desire to minimize

$$\text{Total Cost} = \sum_{i=1}^{14} (14-.5i)\, x_i$$

subject to product

$$\prod_{i=1}^{14} (1-(.015i)^{x_i}) \geq .90$$

and the x_is are the number of backup components plus one for i = 1, 2, . . . 14. The x_i's are the backups + 1 for the actual components.

A 15 stage MSMCO statistical optimization simulation drawing 10,000 sample feasible solutions at each stage went across the feasible solution space to a minimum total cost of 324 dollars and the following number of backups + 1 (for the actual component also) to use to guarantee at least 90% success.

TC = 324.00000
$x_1=1,\ x_2=2,\ x_3=2,\ x_4=2,\ x_5=2,\ x_6=2,\ x_7=2$
$x_8=3,\ x_9=3,\ x_{10}=2,\ x_{11}=3,\ x_{12}=3,\ x_{13}=3,\ x_{14}=3$

Therefore, component one has no backup, whereas components 2, 3, 4, 5, 6, 7, and 10 have one backup, the (2s) and the rest of them have 2 backups (the 3s). This should be acceptable for the current application as this subassembly only requires a 90% success rate.

However, another application requires this subassembly to work 99% of the time. Therefore, the company will make two of these assemblies and attach them together using one of them as a backup. Therefore, the probability of success then will be $(1-.1^2) = 1-.01 = .99$ but it will cost twice as much 2 x 324 = 648 dollars, unless they get quantity discounts.

Also, if .999 success rate was required $(1-.1^3) = 1-.001 = .999$ would entail putting three of these assemblies together with two of them as backups and a cost of 3x324 = 972 dollars.

Some commercial airplanes have more than a million parts and components in them. Many of them are backups that will work if the primary device fails, as safety and performance are paramount.

In those types of cases switching to double or triple or extended precision in the optimization computer program may be advisable as more and more product calculations with small numbers are required.

FACTORY PRODUCTION OPTIMIZATION

A factory takes in tons of twenty-five input compounds and by processing them in their factory in various combinations turns them into twenty-five valuable products they can sell to their customers.

Currently, their customers have ordered the following number of units of products one through twenty-five in the pattern

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

56715	231800	2643026	1020815	2391873
1574224	999774	1929429	1108265	443064
1719470	1337913	11588	252658	49706
1508500	77950	816209	31334	83445
1697778	245929	96441	1026064	70690

all in units

So the question for the factory managers and engineers is how many tons of compounds 1, 2, 3 . . . 25 should be sent into processing to meet these customer demands if the processing yield equations relating how the input compounds combine to produce units of the products to be sold are

$$x_i^4 + x_{i+1}^3 + x_{i+2}^2 \cdot x_{i+3} = C_j$$

for i = 1, 2, 3, . . . 22

and

$$x_{23}^4 + x_1 \cdot x_2 \cdot x_{23} \cdot x_{25} = C_{23}$$

$$x_{24}^4 + x_2 \cdot x_9 \cdot x_{21} \cdot x_{24} = C_{24}$$

$$x_{25}^4 + x_6 \cdot x_{12} \cdot x_{15} \cdot x_{23} = C_{25}$$

subject to $1 \le x_i \le 40$ tons for all the input compounds i=1, 2, 3, … .25 where again the c_j's are for j = 1, 2, . . . 25 are the customers' orders in units.

Therefore, they write a statistical optimization solution using multi stage Monte Carlo optimization (MSMCO) simulation to minimize

$$f(x_1, x_2, x_3 \ldots x_{25}) = \sum_{j=1}^{25} \left| L_j - C_j \right|$$

where L_j and C_j are the left- and right-hand sides of equations j = 1, 2, 3, . . . 25. It is easier and preferred to have the inputs be whole numbers of tons (although they can do fractions if they have to). So there are about $40^{25} \cong 1 \times 10^{40}$ feasible solutions.

THE SOLUTION SIMULATION

The solution plan is to draw 300,000 feasible solutions (in the 1 to 40 ton regions) in each of 30 stages of ever moving and ever decreasing in size 25 dimensional "rectangles" in pursuit of the optimal (always storing the best answer so far, lowest total equation errors) at every point in this simulation. This was done twice using real arithmetic and the total errors of 190,835 and 338,753 are unacceptably large.

Therefore, the 25 x_i values for each of these two sub-simulations are compared coordinate-wise and any ones that are close together are rounded to the nearest whole number and pinned down at that exact number for the rest of the simulation. X_{21} was 36.03 in the first pass and x_{21} was 36.05 in the second pass so x_{21} is fixed at the nearest integer x_{21} = 36 for the rest of the simulation. This also changes the real value into whole numbers which are preferred.

Then two more 30 stages (drawing 300,000 sample answers at each stage) tries are done and total equation errors are now down to 6885.673 and 86459.266.

Then three more pairs like these first two pairs are done (with more pinning down of the x_i's that are quite close in value). These total equation errors were

1924.614 and 1238.664
782.923 636.197
.125 .141

Then one more 30 stage 300,000 sample at each stage is done and the answer is

0.000 = total equation error

x_1=4.00	x_2=19.00	x_3=40.00	x_4=31.00	x_5=39.00
x_6= 35.00	x_7=31.00	x_8=37.00	x_9=32.00	x_{10}=25.00
x_{11}=36.00	x_{12}=34.00	x_{13}=5.00	x_{14}=22.00	x_{15}=3.00
x_{16}=35.00	x_{17}=15.00	x_{18}=30.00	x_{19}=5.00	x_{20}=13.00
x_{21}=36.00	x_{22}=22.00	x_{23}=17.00	x_{24}=26.00	x_{25}=10.00

Therefore, the answer is to use 4 tons of compound 1 and 19 tons of compounds 2 . . . x_5 = 39 tons of compound 5 . . . down to x_{25} = 10 tons of compound 25 to process and make into 56,715 units of product 1 and 231,800 units of product 2 . . . down to 70690 units of product 25 that the customers want.

This was a hypothetical problem solved with multi stage Monte Carlo optimization (MSMCO). Real world input output analysis in factories, chemical plants, oil refineries, etc. are complex and important. Linear algebra and linear programming will work on some of these problems.

However, input and output analysis in the real world of business, economics and science can be nonlinear. When it is, this type of statistical optimization simulation can be useful to solve the problem exactly or produce a useful approximate solution.

STATISTICAL OPTIMIZATION

Multi stage Monte Carlo optimization (MSMCO) or statistical optimization is partially illustrated geometrically and statistically in Figure 1. Every optimization (whether it is a maximization problem (profit, output, etc.) or a minimization problem (less cost or pollution or a system of equations (minimizing total error to zero)) has a sampling distribution of all of the feasible solutions. Then it is just a matter of designing and implementing a computer simulation program to cross the feasible solution space (distribution) to the minimum on the far left hand side (see Figure 1) or a maximum on the far right hand side (see Figure 1) to find the optimal or a useful approximate solution.

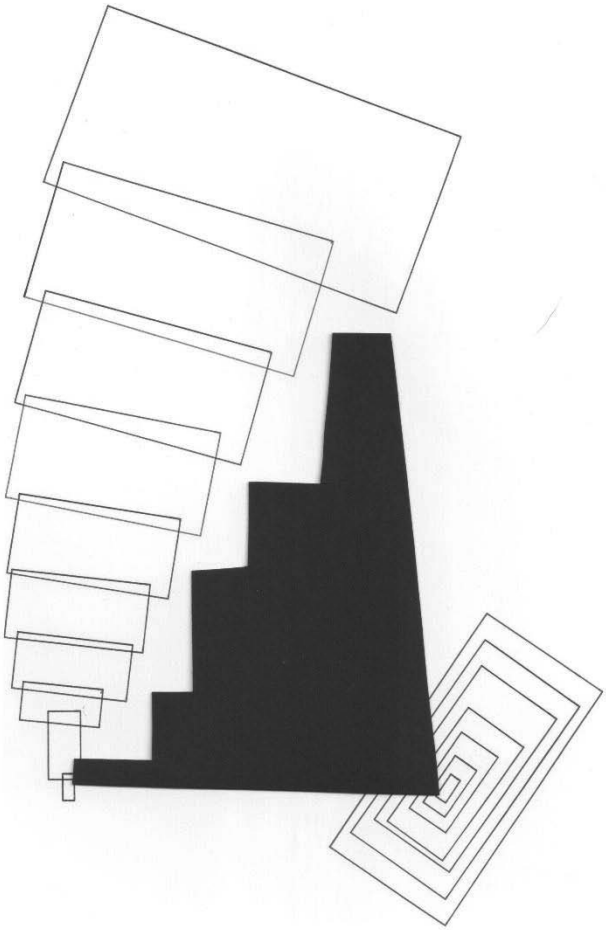


Figure 1: A Ten Stage Search for the Minimum and a Seven Stage Search for the Maximum

The 25 equation and 25 variable system worked on in the previous section had 40^{25} or about $1.125899907 \times 10^{40}$ feasible solutions. Therefore, a complete search is out of the question even with today's powerful 21st century computers.

However, our multi stage Monte Carlo optimization (MSMCO) "random" search learned from its sampling mistakes and remembered its best answers (always centering its focused search around the "best answer so far") and closed in geometrically and found the true optimal after only drawing $11 \times 30 \times 300000 = 99{,}000{,}000$ of the about 1×10^{40} feasible solutions.
Whatever the shape and size of the feasible solution space MSMCO (statistical optimization) will "study" the patterns and trails available in the sampling distribution to find its way to the optimal or a useful approximation as long as a fairly powerful computer is available for use.

The author ran this 99,000,000 sample focused search on the computer in his office in a few minutes of computer run time. As computers get more and more powerful and cheaper and ubiquitous this should become more and more viable a solution technique. This statistical optimization simulation can be written in almost any computer language.

CONCLUSION

Presented here were three nonlinear multivariate optimization problems that were solved using statistical optimization (multi stage Monte Carlo optimization MSMCO).

The first one involved a market survey of potential customers for ten products where accurate estimates were required within a constrained budget limitation.

The second one involved making a device where backup components could take over (more costs) if the primary ones failed to work or wore out.

The third example was a large system of nonlinear equations in an input output analysis system in a factory production setting where customers' orders are to be met in a timely fashion.

Further reading on general statistical work and linear and nonlinear optimization can be found in (Anderson 2003), (Anderson, et al 1999), (Black 2014), (Conley 2013), (Conley 2014), (Conley 2015), (Conley 2016), (Hayter 2002), (Keller and Warrack 2003), and (Verna, et al 2018).

The new I.B.M. Summit Super Computer (Smith and Loehrke 2018) can do 2×10^{17} calculations per second leading to the belief that computer simulation techniques can be very useful in economics and all areas of business and business logistics.

REFERENCES

Anderson, T. W. 2003. *Multivariate Statistical Analysis*, 3rd edition, Wiley and Sons, New York.

Anderson, D. R., Sweeney, D. J., Williams, T. A. 1999. *Statistics for Business and Economics*, 7th edition. Southwestern College Publishing, Cincinnati, Ohio.

Black, K. 2014. *Business Statistics for Contemporary Decision Making*, 8th edition. John Wiley & Sons, New York.

Conley, W. C. 2013. Multivariate inventory and selected problems using multi stage monte Carlo optimization. *Proceedings of the 2013 Industrial Simulation Conference* ISC2013, Ghent University, Ghent, Belgium, May 22-24, Ostend, Belgium: EUROSIS, pp. 208-216.

Conley, W. C. 2014. Machine and engine adjustment. *Proceedings of the 23014 European Simulation and Modelling Conference* ESM2014, University of Porto, Porto, Portugal, October 22-24. Ostend, Belgium: EUROSIS, pp. 321-326.

Conley, W. C. 2015. The other application area of survey sampling. *Proceedings of the 2015 Industrial Simulation Conference ISC2015*, Valencia, Spain. Ostend, Belgium: EUROSIS, pp. 10-14..

Conley, W. 2016. Statistical optimization applied to chemical yield equations. *Proceedings of the 2016 Industrial Simulation Conference ISC2016*, Universitatea Politechnica, Bucharest, Romania. Ostend, Belgium: EUROSIS, pp. 145-148.

Hayter, A. J. 2002. *Probability and Statistics for Engineers and Scientists*, 2nd Edition, Duxbury Press, Pacific Grove, California.

Keller, G. and Warrack, B. 2003. *Statistics for Management and Economics*. 6th Edition. Thompson Brooks/Cole, Pacific Grove, California.

Smith, M. and Loehrke, J. 2018. "20000 trillion calculations per second," *USA Today Weekend Edition*, June 22, 2018, Arlington, Virginia, p. 1.

Verna, A., Jago, S., Lopez, J. Chen, J., and Bracht, P. 2018. Multi Scale Simulation and Optimization. *Proceedings of the 16th International Industrial Simulation Conference ISC 2018,* Ponta Delgata, Portugal. Ostend, Belgium: EUROSIS, pp. 13-15.

BIOGRAPHY

WILLIAM CONLEY received a B.A. in mathematics (with honors) from Albion College in 1970, an M.A. in theoretical mathematics from Western Michigan University in 1971, a 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 five books and more than 230 publications world-wide. He also gave a speech and presentation on his research at the 7th International Genetic Algorithms Conference held in the United States in July 1997. 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. He was also named to Omicron Delta Epsilon (ODE) the national economics honorary and Kappa Mu Epsilon (KME) the national mathematics honorary.

MANPOWER MANAGEMENT

Manpower planning using simulation and heuristic optimization

Oussama Mazari-Abdessameud
Johan Van Kerckhoven
Filip Van Utterbeeck
Royal Military Academy, Department of Mathematics
email: mazari.abdessameud.oussama@gmail.com

Marie-Anne Guerry
Vrije Universiteit Brussel,
Department of Business Technology and Operations

KEYWORDS
Manpower planning, Entity based simulation, Heuristic optimization, Military organization

ABSTRACT

This article presents the use of a Generic Manpower Simulation Engine (GMSE) coupled with a heuristic optimization algorithm to tackle manpower planning problems. Manpower planning aims to meet the organization's demand for human resources. The GMSE is an entity based simulator that mimics the evolution of the manpower according to a set of policies. We propose a heuristic optimization algorithm to find the optimal policies that result in a limited deviation from the organization's target. We applied our approach to the case study of a military organization.

INTRODUCTION

Manpower planning is an important task for any organization. It aims to "Determine the number of personnel and their skills that best meet the future operational requirements of an enterprise"(Gass 1991). Wang (2005) expresses the effective manpower planning in a military context as "there will continue to be sufficient people with the required competencies to deliver the capability output required by the Government at affordable cost".

Manpower planning aims to support decisions regarding human resources (HR) policies (such as recruitment, promotion and job positions assignments). These policies have an impact on the manpower in the short term as well as the long term. In order to make a suitable choice, HR managers resort to manpower planning models and techniques. Hall (2009) distinguishes three categories of existing manpower planning models and techniques, namely Markov models, simulation models, and optimization approaches. Simulation models were widely used in the manpower planning context. We refer to Jnitova et al. (2017) that review over thirty articles of simulation in manpower planning. Both heuristic and exact optimization approaches were employed for manpower planning. Cai et al. (2013), Hall and Fu (2015) and Mazari Abdessameud et al. (2018) use exact optimization approaches to tackle manpower planning problems. On the other hand, Showalter et al. (1977), Fowler et al. (2008) and Nearchou and Lagodimos (2013) employ heuristic optimization approaches for similar problems. In this article, we present a simulation model in combination with an optimization technique to tackle manpower planning problems. We use a Generic Manpower Simulation Engine (GMSE) for the evaluation of HR policies in a heuristic optimization algorithm. For each manpower state, we want to have the number of personnel in that state as close as possible to the targeted number.

The proposed approach was applied to a real case study, being the Belgian defense manpower. Therefore, the next section describes the military manpower and its specificities. The section thereafter presents the GMSE and how it can be used to simulate a manpower evolution through time. We then illustrate the use of the GMSE with the heuristic optimization algorithms (simulated annealing and genetic algorithm). The article concludes with presenting and discussing the obtained results for the case study.

MILITARY MANPOWER DESCRIPTION

The military manpower consists of soldiers serving for a military organization. Each soldier has a set of characteristics such as military rank, age, acquired competencies, and job position. The specificity of the military manpower system is the hierarchical nature that restricts recruitment to the lowest hierarchy level (Wang 2005, Hall 2009). In other words, soldiers are recruited as basic trainees and they receive promotions and transitions in the military organization towards higher levels. This implies the need for robust planning to meet future manpower requirements. In order to fulfill the personnel demand for the different levels in the hierarchy, soldiers have to make transitions through the hierarchical structure. Those transitions consist generally of rank promotions or job position changes.

The characteristics of the soldier define the state to which they belong. A state is defined based on the characteristics of the soldiers. For example, the state officer means the rank of the soldier is either Lieutenant, Captain, Major or Colonel. A soldier can belong to several states at the same time as they are not necessarily mu-

tually exclusive. For instance, a soldier can be an officer, aged over 40 and can belong to the air force.
The manpower planning objectives are usually defined based on states. The organization targets certain manpower distributions over the different states. For example, the organization could target 3200 soldiers in the state officer, 1500 soldiers in the air force and only 30% aged over 40 years.

GENERIC MANPOWER SIMULATION ENGINE

The GMSE is a decision making aid tool that helps the HR managers predict the manpower evolution under certain HR management policies (Van Kerckhoven et al. 2018). The HR manager defines as input to the GMSE a set of policies (recruitment, promotions, retirement...) and runs a simulation to observe the predicted evolution of the manpower within a period of time. The obtained results can be compared to the target manpower composition. The GMSE is a generic simulation tool and consists of a variety of features. We focus in this section on the exploited features in our study case.
In order to approach a realistic behavior of the manpower, we use an entity based simulation. This choice is motivated by the ability to define the individual behavior of the manpower members. The entities evolve within the simulation according to the defined policies and to their characteristics and individual behavior. We distinguish three main actions that an entity can perform:

1. **Enter the system:** An entity is added into the simulation with its specific characteristics. It can be created by the initialization process or the recruitment one. The initialization process produces a set of entities to mimic the actual organization situation. The characteristics of the entities are loaded from an HR database. On the other hand, the recruitment process is a periodical operation that produces entities according to the recruitment policy. The recruitment policy defines per cycle the number and the characteristics of the generated entities.

2. **Evolve in the system:** Each entity evolves throughout the simulation and changes its characteristics. The basic entity evolution is aging which updates the age of the entity. Also, the entity evolves when performing transitions. Transitions are periodical operations that change a subset of the entity characteristics. An entity is eligible for a transition if it satisfies the required conditions which could be linked to any entity characteristic. Furthermore, the HR manager has to define for each transition a probability that an eligible entity undergoes this transition.

3. **Leave the system:** The GMSE defines two ways to leave the system: retirement and attrition. The retirement is the natural way to leave the system at the end of a career. It occurs periodically for entities satisfying the retirement conditions regarding the age and career length. The attrition is the unnatural way to leave the system which includes voluntary resignation and forced resignation (for instance for medical reasons). Attrition is defined as a probabilistic event occurring each cycle. The attrition probabilities could be determined using historical databases of the organization. Also, attrition is possible when failing certain transitions (e.g. in case of not passing a test).

HEURISTIC OPTIMIZATION ALGORITHMS

The HR manager can use the GMSE to forecast the impact of different HR management policies on the manpower of his organization. However, the organization typically defines an HR target to be satisfied. The target is a distribution of the personnel over the different manpower states. Due to the stochastic behavior of the manpower, the organization defines an acceptance domain. So the target is a number of soldiers in each state with a tolerated rate of deviation. The HR manager has to find the optimal policies to satisfy this target. Therefore, we illustrate in this section heuristic optimization algorithms to find adequate policies satisfying the organization's target.
The proposed heuristic optimization algorithms use the GMSE in the evaluation of the solutions. Figure 1 illustrates how the GMSE is used in the heuristic optimization algorithms. In this article, we explore two algorithms, namely simulated annealing, and a genetic algorithm. The solution(s) fitness evaluation process is the same for all explored algorithms. The searched solutions are recruitment policies (how much we recruit on which recruitment type).
Solution(s) fitness evaluation: consists of two components, the GMSE and the compare to the target block. The first component simulates the evolution of the manpower depending on the policies received from the generate solution(s) block. The second component compares the simulation results to the target and computes the fitness of the solution. Table 1 shows a layout of the computed fitness where the score is a function of the target unfulfillment rate and time. Also for each state, we note the cumulative surplus and cumulative deficiency throughout the simulation run. As the simulation is stochastic, this operation (simulation + fitness evaluation) is repeated several times. The final fitness of a solution is the mean of the generated fitnesses. The surpluses and deficiencies measures are used together with the direct dependencies (described below) to generate better solutions.
Direct dependencies: For large organizations such as

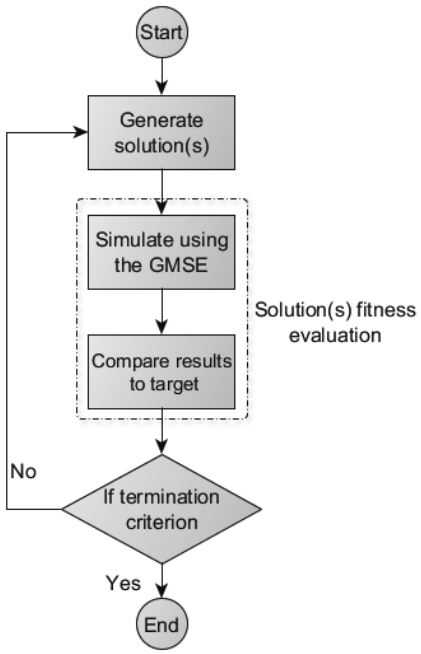


Figure 1: Heuristic optimization algorithms flowchart

Table 1: Solution fitness layout

Score	f(unfulfillment rate, time)	
State 1	Surplus 1 (S_1)	Deficiency 1 (D_1)
State 2	Surplus 2 (S_2)	Deficiency 2 (D_2)
...	...	...
State n	Surplus n (S_n)	Deficiency n (D_n)

the military, the manpower system is complex and every management action has an impact on a large part of the organization. However, certain actions have a first-order impact on specific manpower states. For example in the military organization, recruiting for a non-commissioned officers (NCO) class, impacts the population in the officers class through NCO getting promotions to officers. But it has a first-order impact on the number of the population in that class. We define this impact as a direct dependency between recruiting for the NCO class and the population in the NCO class.

Optimization Algorithms: Having defined the Solution(s) fitness evaluation and the direct dependencies, we illustrate the developed algorithms:

1. **Simulated annealing:** The first explored algorithm is simulated annealing. We start with an arbitrary solution that will be tuned using the direct dependencies and the fitness. After evaluating the generated solution, it is accepted based on its score in the fitness table following a simulated annealing principle. For the accepted solution, we randomly chose a subset of parameters of the solution to tune. For each selected parameter, if its direct dependency state has a larger cumulative surplus than cumulative deficiency, we reduce the parameter by a random amount. In the inverse case, we augment it. In order to consider the system complexity which does not always obey only to the direct dependencies, we include a random diversification scheme. We repeat this whole process until the termination criterion is reached. We fix the termination criterion as a number of iterations.

2. **Genetic Algorithm:** For the genetic algorithm, we generate an initial random population of solutions. After evaluating all the generated solutions, we perform a crossover of the best solution so far (based on the score) with all the other solutions in the population. The crossover takes two parents to generate one child. It considers for each parameter of the solution a linear extrapolation between the two parents. We use in the extrapolation the fitness of the directly correlated target with the considered parameter. For each parent, we use the dominating fitness component which is either the surplus component or the deficiency one. Figure 2 shows two situations of the fitness of the parents and how the crossover extrapolates to find the child's parameter value. If the parameter values for the two parents and the child are X_1, X_2, X_c respectively, we write the crossover function as:

$$X_c = X_1 - (\frac{X_1 - X_2}{\alpha_1 Y_1 - \alpha_2 Y_2})\alpha_1 Y_1$$

Where:

$$\alpha_i = \begin{cases} 1 & S_i \geq D_i \\ -1 & \text{else.} \end{cases}$$

$$Y_i = \begin{cases} S_i & S_i \geq D_i \\ D_i & \text{else.} \end{cases}$$

The crossovers generate a new population that can be mutated with a random minor mutation. The mutation does not exploit any form of fitness as it helps to diversify the explored solutions. We continue to evaluate the fitness of populations and generate new ones until we reach the termination criterion fixed as a number of generations.

3. **Simulated annealing with a crossover function:** We combine in this approach the two previous solution approaches. We perform simulated annealing as explained before with introducing a crossover of the current solution with a random solution. We generate following a probability of $f(iteration^{-1})$ a random solution. This generated solution is introduced with the current one in a crossover. The best of the three solutions is retained for the next iteration. The crossover function accelerates the process of getting to a zone with higher quality solutions.

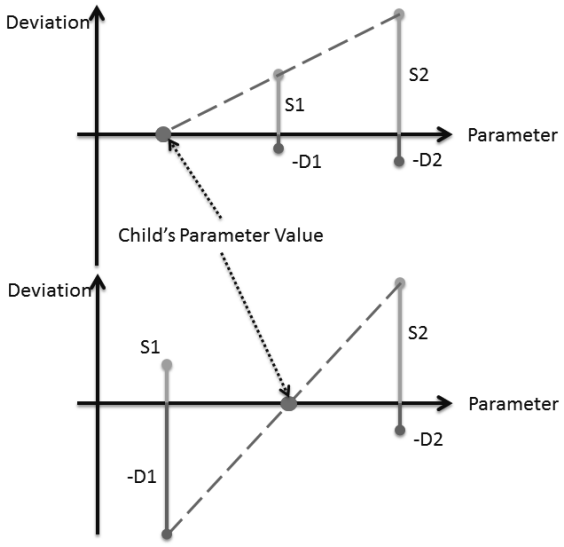


Figure 2: Linear extrapolation between two parents to find the new child's parameter value - two different cases

RESULTS AND DISCUSSION

GMSE and the case study

The GMSE is applied to military manpower and specifically to the Belgian defense manpower. We perform a simulation for a time span of 30 years with a starting population of 27700 soldiers. We consider seven main characteristics of the soldiers and 120 possible transitions in the organization. Also, there exist 10 different recruitment types for the organization. The attrition rates are derived from a historical database of the organization. We perform a simulation with an arbitrary set of HR management policies. The HR policies regard the annual recruitment for each recruitment type, promotion probabilities from one rank to the other and the probability to successfully perform certain transitions. Generally, a simulation run for 30 years with an annual population between 25000 and 30000 takes about one minute on an Intel(R) Core(TM) i7 CPU 950 3.07GHz 307GHz. Figure 3 represents the manpower obtained by the simulation. The graph shows a break down of the manpower by the rank class, an important soldiers' characteristic. We notice that the total manpower count goes over 30000 soldiers before reaching the end of the simulation. In fact, this count is far from the organization's target (Table 2). Also, the distribution of the manpower over the states does not respect the target distribution.

In order to close the gap between the target distributions and the simulation results, the HR manager can alter the different simulated policies. However, the number of simulated policies parameters could be very large (in this case: 120 transition parameters and the different

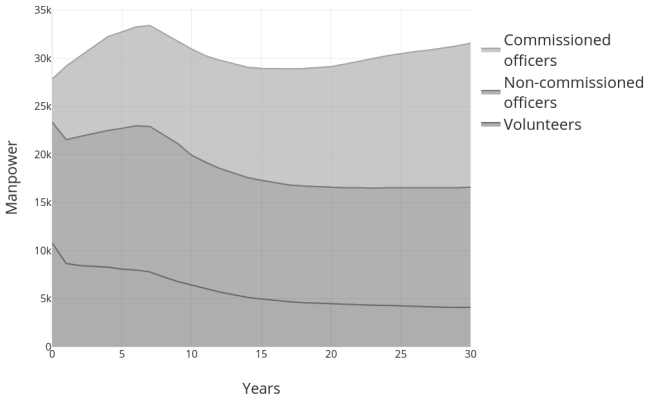


Figure 3: Simulation result for the simulation with arbitrary choice of policies - break down of the population by rank class

annual recruitments) and also the simulation takes a while. To avoid these heavy tasks for the HR manager, we apply a heuristic optimization approach to reach the target distributions.

Optimization results for the case study

We applied our three proposed algorithms to the case study described in the previous subsection. The main target distribution of our case study is the distribution of the manpower over the rank classes which is illustrated in table 2. In fact, we do not target any specific distribution for the first seven years of the simulation. Our target is due seven years after the start of the simulation. The target value has its acceptance domain set to a deviation of 10%.

Table 2: Target manpower distribution over rank classes

	Target value from year 0 to 7	Target value from year 7 to 30
Officers	No value	7400
NCO	No value	10500
Volunteers	No value	6000
Total population	No value	23900

The three algorithms give almost similar results. However, the simulated annealing with a crossover function performs somewhat better than the others. Figure 4 shows the evolution of the score of the best-retained solution through the execution of the three algorithms. The simulated annealing with a crossover function had the best score. We notice that the genetic algorithm gets to better solutions faster than the other two, but after a certain level, it stabilizes. For the simulated annealing

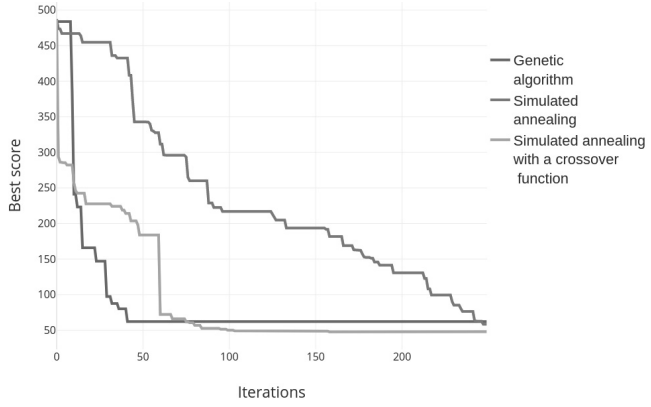


Figure 4: Best solution's score evolution for the three proposed heuristic search algorithms

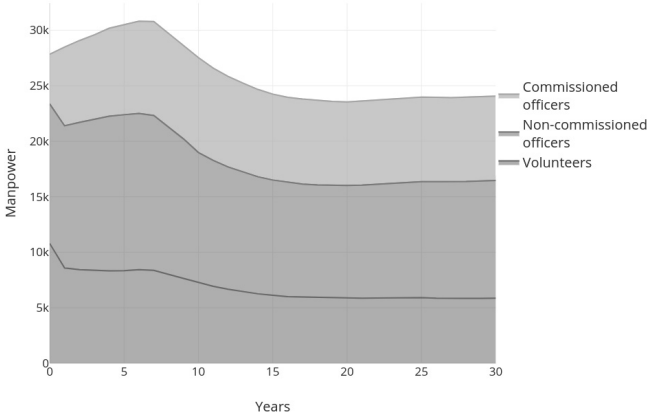


Figure 5: Result of the best solution obtained through the optimization - break down of the population by rank class

with a crossover, it gains quick advancement in the best solution score comparing to the simulated annealing due to the random crossovers. However, it keeps improving the solution better than the genetic algorithm owing to the simulated annealing principle.

Figure 5 illustrates the simulation results of the best-found solution. The graph shows a break down of the population by the rank class. Figure 6 shows the deviations from the target by state and the allowable deviation of 10%. We notice that the target was partially satisfied. For the officers class, we can see the deviation maintained in the allowable domain except for the two first years which presents a miner unfulfilled target. However, for the volunteers and the NCO, we start with an important deviation which gets reduced and stabilizes in the accepted domain. The deviations from the targets cannot be stable through the years of simulation because of the stochastic behavior of the entities. The unfulfilled part (years 7-12) is due to the initial manpower which was very far from the target.

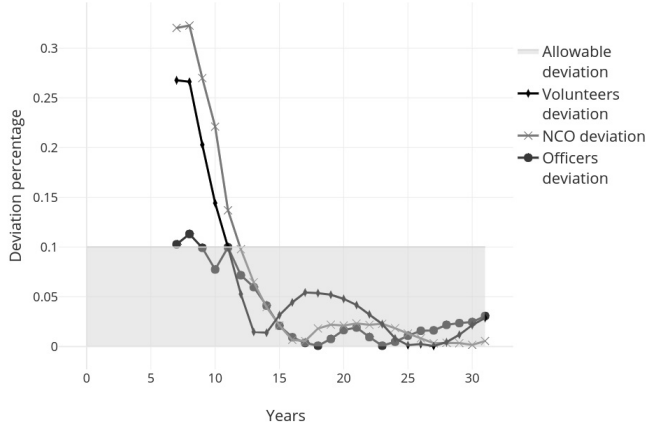


Figure 6: Deviation percentage from the target by state

CONCLUSION

This article presents a Generic Manpower Simulation Engine (GMSE) augmented with heuristic optimization algorithms to tackle manpower planning problems. The GMSE is a simulation tool that, given an initial population of an organization and its management policies, predicts the evolution of the manpower in the future. It uses an entity based approach to mimic the behavior of the manpower members. Furthermore, we applied heuristic optimization algorithms to help the HR manager find the optimal HR management policies that enable reaching the organization's HR target.

We applied our methodology to a case study using real-world data from the Belgian Defense. The optimization results show that the HR manager could rely on the developed algorithms to define the organization's HR management policies. However, the set of HR management policies considered in the algorithms is still limited.

In future work, we want to enlarge the set of considered policies in the optimization part. This will give more freedom to the optimizer and may result in better solutions. Also, instead of considering only the direct dependencies, we want to use a more advanced dependency function to take into account more complex impacts of HR management policies. Furthermore, we need to apply our approach to other case studies to ensure the robustness of it.

REFERENCES

Yilin Cai, Zizhen Zhang, Songshan Guo, Hu Qin, and Andrew Lim. A tree-based tabu search algorithm for the manpower allocation problem with time windows and job-teaming constraints. *IJCAI International Joint Conference on Artificial Intelligence*, pages 496–502, 2013.

John W Fowler, Pornsarun Wirojanagud, and Esma S Gel. Heuristics for workforce planning with worker

differences. *European Journal of Operational Research*, 190:724–740, 2008.

Saul I. Gass. Military manpower planning models. *Computers & Operations Research*, 18:65–73, 1991.

Andrew O Hall. *Simulating and optimizing:Military manpower modeling and mountain range options*. PhD thesis, 2009.

Andrew O. Hall and Michael C. Fu. Optimal army officer force profiles. *Optimization Letters*, 9(8):1769–1785, 2015.

Victoria Jnitova, Sondoss Elsawah, and Michael Ryan. Review of simulation models in military workforce planning and management context. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 14(4):447–463, 2017.

Oussama Mazari Abdessameud, Filip Van Utterbeeck, Johan Van Kerckhoven, and Marie-Anne Guerry. Military Manpower Planning - Towards Simultaneous Optimization of Statutory and Competence Logics using Population based Approaches. In *Proceedings of the 7th International Conference on Operations Research and Enterprise Systems*, number Icores, pages 178–185, 2018.

Andreas C Nearchou and Athanasios G. Lagodimos. Heuristic solutions for the economic manpower shift planning problem. *European J of Industrial Engineering*, 7(December), 2013.

M. J. Showalter, L. J. Krajewski, and L. P. Ritzman. Manpower allocation in U. S. postal facilities: A heuristic approach. *Computers & Operations Research*, 4:257–269, 1977.

Johan Van Kerckhoven, Oussama Mazari-abdessameud, and Filip Van Utterbeeck. Simulation and long term (military) manpower planning : a custom generic simulation tool. In *The Tenth International Conference on Advances in System Simulation*, 2018.

Jun Wang. A review of operations research applications in workforce planning and potential modeling of military training. *DSTO Systems Sciences Laboratory, Edinburgh Australia*, 2005.

MODELLING FIRE FIGHTERS ACTIONS IN A FOREST FIRE SIMULATION ALGORITHM

Yves Dumond
Laboratoire LISTIC, Université Savoie Mont Blanc
Campus Scientifique
F-73376 Le Bourget-du-Lac Cedex
e-mail: yves.dumond@univ-smb.fr

KEYWORDS
Forest fire simulation, vector-based approach, raster-based approach, fire fighters actions, geographical information system.

ABSTRACT

We introduce in this paper an algorithm for forest fire spread modelling which involves actions carried on by fire fighters. The calculation of fire propagation is performed on a graph mapped onto a 2-D grid of cells representing the landscape. The propagating vertices are either centers or corners of the cells whereas vertices belonging to the fire final contour can be located in any points. The fire rate of spread along the different edges of the graph are calculated in 3-D cells approximating the relief. The corresponding calculations are made with the proviso that starting from a given ignition point, the fire evolves to an elliptic shape. The global fire propagation is then obtained by means of the Dijkstra's shortest path algorithm. We conclude on the fact that taking into account fire fighters actions is indispensable when forest fire simulation is used in operational frameworks.

INTRODUCTION

In many parts of the world, forest fires are one of the worst natural disasters. Indeed, this curse can deeply modify the ecosystems and lead to species extinction. Each year, 3.5 million square kilometres are burnt on the planet, 12 thousand of which in the European Union. On this last point, it is worth noting that if the problem was until now limited to the Mediterranean basin, Central and Northern Europe are now threatened. As a matter of fact, Germany and Scandinavia had to face important difficulties during summer 2018.
In this context, the development of software tools dedicated to forest fires fight management is a key issue. In particular, simulation can be central to fight management since it allows means dispatching to be anticipated. This technology is currently implemented in North America where the level of reliability it has reached makes it possible its use in operational frameworks (Finney, 2004), (Peterson et al., 2009), (Tymstra et al., 2009), (Noonan-Wright et al., 2011). Elsewhere in the world, it seems that its use remains somewhat lower. Nevertheless, huge amounts of research works are dedicated to this topic worldwide.

The work presented in this paper is the result of a collaboration between the University Savoie Mont Blanc and a Fire Department in the South of France (SDIS 06). First, we developed a software tool dedicated to fight management (Dumond, 2015). Present works are focusing on forest fire simulation with operational purposes. In particular, this implies that computation durations must be short enough to allow results to be exploited in due time.

FOREST FIRE FIGHT MANAGEMENT

Tactical situations

The cornerstone of forest fire fight management is usually referred to as *tactical situation*. This concept is generally implemented using a geographical information system (GIS) with different base maps such as geological survey maps or aerial photographs. An example of tactical situation is given hereafter (Fig. 1):

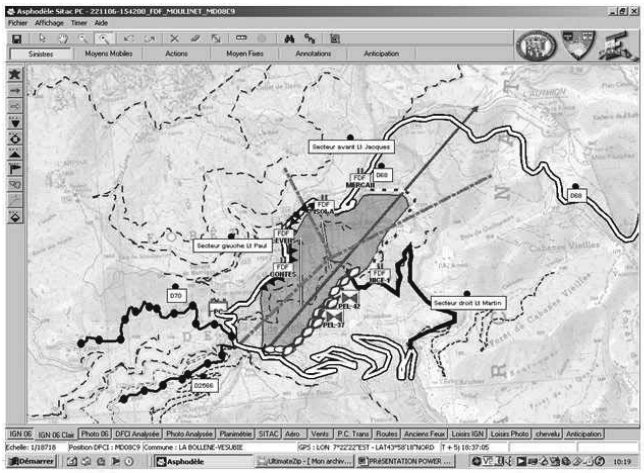


Figure 1: A tactical situation (Courtesy of SDIS 06)

Such graphics encompass all relevant information required to manage the fight against the fire:

- The morphological characteristics of the areas of operations, in particular the description of all man-made items, e.g. high voltage lines that can be a serious threat to aircraft, the location of fire hydrants which allow vehicles to be refilled

and the layout of roads and forest tracks that makes fire spots accessible.

- All the characteristics the fire, in particular the perimeters concerned and the axes of propagation.

- The location of the fire-fighting units engaged on the field.

- The nature and the location of all the actions performed by fire-fighters, both past and present.

Fire fighters actions

Many works have been dedicated to means allocation for wildland fire suppression (Griffith et al., 2017). Such activities are to be conducted in environments which are both dynamic and uncertain with the risk of heavy damage in case of errors in the implemented strategy.

The corresponding models fall within the scope of mathematical optimization and decision support theory (Ntaimo et al., 2013). They can be either deterministic or stochastic. Furthermore, they are by nature related to simulation algorithms (Hu and Sun, 2007), (Hu and Ntaimo, 2009). Following the same objective, other works focus on genetic algorithms (Homchaudhuri et al., 2013)

Among the numerous kinds of actions which can be undertaken by fire fighters, we pay attention in this paper to the following ones which are those taken into account in the algorithm introduced later in the text (Fig. 2):

- A *base line* represent a high concentration of fighting units formed with the intention to attack the fire, generally on its head. Here, the objective is to stop the fire.

- As the name suggests, a *flank attack* consists in fighting the fire on its side. This requires less means than a base line. In this case, the objective is to contain the propagation of the fire.

- *Droppings* from aircraft, i.e. planes or helicopters, can either concern water or *retardant*, i.e. water mixed with a chemical product whose effect is to slow down fire progression.

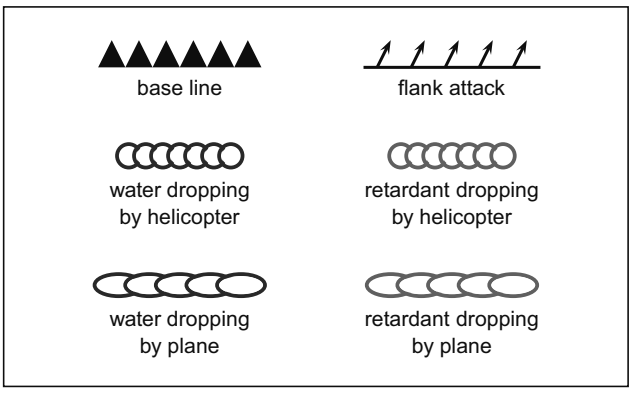


Figure 2: Symbols denoting fire fighters actions

The actions defined above are supposed to have the same effect on the propagation of the fire: all of them define broken lines, under the form of a list of segments which cannot be crossed by the fire. Therefore, in the rest of the paper, they will be referred to as *stop lines*.

FOREST FIRE MODELLING

For the past 50 years, a lot of research works have been dedicated to forest fire simulation: see for instance (Sullivan, 2009) for a very comprehensive description. It is customary to distinguish between three overall approaches: *physical models* (Richards, 2005) which reason in terms of heat transfers according to different local parameters (fuel bed, temperature, wind, slope, etc.); *empirical models* (FCFDG, 1992) which establish statistical laws with respect to fire rates of spread under the basis of thousands of field measurements; *semi-empirical models* (Rothermel, 1972), (Lopes et al., 2002), (Vakalis, 2004), (Finney, 2004), (Peterson et al., 2009), (Tymstra et al., 2009), (Kalabokidis et al., 2013), which appear as a compromise between the previous models, i.e. statistical laws are used for the elaboration of formulae which in turn provide fire rates of spread as a function of local parameters.

For the time being, semi-empirical models are recognized as the most efficient ones since they provide a suitable balance between calculation efficiency and accuracy of the fire contours obtained. These models can be implemented using two different approaches:

- *Vector-based algorithms* (Finney, 2004) assume that starting from an ignition at a given point and without external influence, the fire evolves to a specific shape, generally an ellipse or a double-ellipse (it is worth noting that some authors use ovoids or tear drops). Thus, at time t the global fire contour is discretized by a set of points which are regarded as independent ignition points. At time $t + \delta t$, the envelope curve (Glasa and Halada, 2007) of the local shapes provides the new fire contour.
In this approach, a delicate point is the definition of the discretization steps, i.e. respectively the distance chosen between the different ignition points and the elapsed time (i.e. δt) between two successive envelope curves. On this last point, it is worth noting that the characteristics of the ellipses are determined according to the physical properties of the ignition points which are those of the cells concerned. Thus, provided δt is long enough, some ellipses will, due to their growth, straddle neighbouring cells which may have other physical properties. This is clearly a cause of distortion in the calculation.

- *Raster-based algorithms* (Hernándes Encinas et al., 2007), (Peterson et al., 2009), (Tymstra et al., 2009) rely primarily on the division of the geographic space into a grid of cells which generally have a square form. However, some authors use hexagonal cells (Trunfio et al., 2011). Fire growth is thus computed over a graph whose vertices are generally the centers of the cells constituting the underlying grids. The center of each cell is connected to the center of other cells by a set of edges. Different types of neighbourhood can be defined as

a function of the number and of the direction of the outgoing edges. In any case, the graph comprises a set of arbitrarily defined edges. Here, the main cause of approximation is that these edges do not precisely overlap the highest fire rates of spread. Hence, the final contour is generally underestimated.

It is of huge importance to note that, in practice, the vector-based and the raster-based approaches are mixed together in order to elaborate hybrid algorithms. Indeed, on the one hand, vector-based algorithms cannot ignore the fact that local parameters are provided by GIS under the assumption that the landscape is divided into cells and on the other hand, shapes are often used in raster-based algorithms as a way of getting realistic rates of spread. The algorithm presented in this paper actually falls within the category of hybrid algorithms.

THE PROPOSED MODEL

Available ground data: 2-D grid and local winds

The first hypothesis is the existence of a 2-D grid representing the landscape which is made of cells of size 25m $\times$ 25m. For each cell, the underlying GIS provides the *geographic coordinates*, the *altitude*, the *slope* and the *aspect*, namely the orientation of the cell with respect to the points of the compass. The vegetation is not considered here since this study concerns Mediterranean shrub areas for which fuel bed properties are supposed to be constant.

Wind is a parameter of huge importance in fire spread modelling. We need to separate the levels at which wind is considered. Upper-air wind, referred to as *laminar wind* is constant. On the contrary, close to the ground, the wind is subject to the influence of the relief. This results in important differences from cell to cell, both in direction and intensity. This justifies the term of *local wind*. Therefore, we have identified a set of dominant winds in the areas concerned. For each of them, and for several orientations within given ranges and different intensities, a dedicated wind map has been produced using the WindNinja software system (Butler et al., 2014). Then, in operational context, it is enough to load the corresponding wind map as a function of the current meteorological situation. This allows a local wind to be available for each cell.

Fire spread calculation over a graph

Considering the 2-D grid mapped onto the landscape, we define a directed graph $\mathcal{G}$ which contains three types of vertex:

- *Propagating vertices* which are considered as potential ignition points. Hence, fire spreads from one to another according to a given neighbourhood. In constrast with what is usually done, the vertices considered here are either *centers* or *corners* of cells of $\mathcal{G}$. This allows the number of outgoing edges of a given vertex to be increased and therefore to get a better level of discretization in the calculation of fire contours. In the different figures, they are represented by the symbol $\circledast$.

- *Final vertices* which by construction belong to the final contour determined by the algorithm. These vertices can be any points of the grid. In the different figures, they are represented by the symbol $\otimes$.

- Vertices which are necessary for the calculation of the propagation but whose status remains undetermined, at least temporarily. They are represented in the different figures by the symbol $\odot$.

In $\mathcal{G}$, the out-neighbourhood of a propagating vertex is defined as a function of its type, i.e. "center" or "corner". The Northeast quarter of the out-neighbourhood of a vertex of each type is given below (Fig. 3):

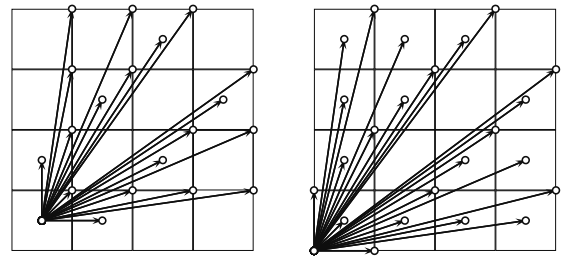


Figure 3: Out-neighbourhood of vertices

The complete out-neighborhood of a propagating vertex a, which can be easily deduced from Figure 3 by symmetry, is denoted by $\mathcal{N}(a)$.

Moreover, an ignition date can be calculated for every vertex of $\mathcal{G}$. Thus, for any vertex a, we denote $a.ignition$ the corresponding ignition date.

Fire spread duration along a segment

Consider two vertices a_i and a_j of $\mathcal{G}$ under the assumption that a_i is a propagating vertex. The edge a_ia_j consists of a list of segments, each of them crossing a different cell (Fig. 4):

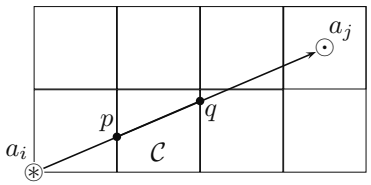


Figure 4: Edge linking two propagating vertices

It is clear that the local parameters which determine the fire rate of spread vary from cell to cell. Hence, the latter is different on each segment and, as a consequence, must be subject to a specific calculation. For instance, the points p and q define the segment of a_ia_j that crosses the cell $\mathcal{C}$. We show hereafter how the fire rate of spread along pq can be calculated.

To ensure precision, this rate of spread must be calculated in a 3-D cell in order to stay as close as possible to the relief of the landscape. As stated above, the cell $\mathcal{C}$ has both an orientation and a slope. Therefore, the calculations are made in a plane having identical characteristics and denoted by $P_{\mathcal{C}}$. The vertical projections on $P_{\mathcal{C}}$ of the cell $\mathcal{C}$ and of the points p and q are respectively denoted by $\mathcal{C}^\star$, $p^\star$ and $q^\star$.

First, we postulate that the respective spreading durations along the segments pq and $p^\star q^\star$ are identical. Under this assumption, we consider $p^\star$ as both an ignition point and the origin of the following vectors (Fig. 5):

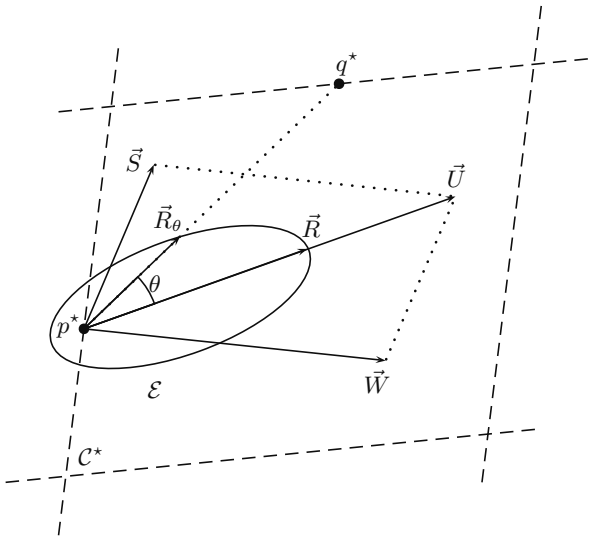


Figure 5: Ellipse construction in the cell $\mathcal{C}^\star$

- $\vec{W}$, the local wind vector which is given by the wind map corresponding to current meteorological conditions.

- $\vec{S}$, the local slope vector whose direction is given by the steepest slope line of $\mathcal{C}^\star$ and whose norm is provided by the following formula (Finney, 2004):

$$|\vec{S}| = 15.275 \times \beta^{-0.0225} \times \tan^2(\alpha) \tag{1}$$

where β is defined by the fuel bed and α is the angle defining the steepest slope. Here, the aim is to model the effect of upward propagation of incandescent gases along the slope. This causes the vegetation upstream of the fire zone to dry out and finally to be ignited. Characterizing this phenomenon under the form of a speed vector allows it to be combined with the effect of the wind. As a consequence, the slope vector is often referred to as an *"equivalent-wind"* vector.

In the present setting, namely for the Mediterranean shrub, we use the expression $10 \times \tan^2(\alpha)$ truncated at the value of 10 $m.s^{-1}$ for slopes steeper than 45 degrees. Beyond this threshold, the increase of the slope gradient does not lead to a rise of the corresponding slope effect in terms of propagation speed.

- $\vec{U}$, the resultant wind-slope vector which is defined by the vectorial sum $\vec{S} + \vec{W}$.

According to (McAlpine et al., 1991), starting from a single ignition point and without external influence, a fire evolves to an elliptic shape. This ellipse, denoted hereafter $\mathcal{E}$, can be characterized as follows (Fig. 5):

- The rear focus of $\mathcal{E}$ is determined by the location of $p^\star$.

- The orientation of the main axis of $\mathcal{E}$ is defined by $\vec{U}$.

- The norm of $\vec{R}$, which is equal to the distance between $p^\star$ and the forward extremity of $\mathcal{E}$, defines the highest possible rate of spread in $\mathcal{C}^\star$. In the present setting, this norm is given by a formula (unpublished work) which has been specifically designed for Mediterranean shrub in the South of France. The three parameters required by the calculation are the temperature in the shade T_s, the soil water content S_w and the norm of $\vec{U}$:

$$|\vec{R}| = 180 \times e^{(0.06 \times T_s)} \times \tanh(\tfrac{100 - S_w}{150}) \times (1 + 2 \times (0.8483 + \tanh(\frac{|\vec{U}|}{30} - 1.25))) \tag{2}$$

- The length-to-breath ratio of $\mathcal{E}$, denoted by LB, is given by the following formula (Alexander, 1985):

$$LB = 1 + 0.0012 \times (2.237 \times |\vec{U}|)^{2.154} \tag{3}$$

The value of LB being determined, the eccentricity ϵ of $\mathcal{E}$, is in turn obtained by:

$$\epsilon = \sqrt{1 - \frac{1}{LB^2}} \tag{4}$$

The characterization of $\mathcal{E}$ being completed, the fire rate of spread in a direction forming an angle θ with the main axis is given by the polar equation of $\mathcal{E}$ (Albini and Chase, 1980):

$$|\vec{R}_\theta| = \frac{(1 - \epsilon)}{(1 - \epsilon * cos\theta)} \times |\vec{R}| \tag{5}$$

Note that the equation (5) supposes that the origin of the coordinate system considered is on the rear focus of $\mathcal{E}$.

Hence, if we assume that θ is the angle between $\vec{R}$ and $\overrightarrow{p^\star q^\star}$, the spreading duration between $p^\star$ and $q^\star$, and consequently between p and q, can immediately be obtained by:

$$\delta spread(p^\star, q^\star) = \frac{|\overrightarrow{p^\star q^\star}|}{|\vec{R}_\theta|} = \delta spread(p, q) \tag{6}$$

The out-neighbourhood of vertices of type *corner* is such that some edges exactly fit the border of cells. In that case, the calculation of the spreading duration is made in the two adjacent cells and the smallest value, i.e. that which corresponds to the highest rate of spread, is retained.

Fire spread along an edge

Now, we turn to the questions raised by fire spread along a given edge, being understood that the presence of stop lines can affect this phenomenon.

So, let a_i et a_j be two vertices of $\mathcal{G}$ with the proviso that a_i is a propagating vertex. We assume that the global duration of the simulation is equal to Δ. We study below the propagation along the edge $a_i a_j$. In some instances, we will be led to introduce new vertices. The different following cases deserve to be considered:

• **Case 1:** we assume that $a_i a_j$ does not intersect any stop line. If $a_i a_j$ is made of a list of n segments $\langle \dots, p_k q_k, \dots \rangle$, then the spreading duration along this edge is defined by:

$$\delta spread(a_i, a_j) = \sum_{k \in 1 \dots n} \delta spread(p_k, q_k) \qquad (7)$$

Moreover, if $a_i.ignition + \delta spread(a_i, a_j) < \Delta$, the fire will actually reach a_j. Thus, a_j is in turn a propagating vertex such that $a_j.ignition = a_i.ignition + \delta spread(a_i, a_j)$ (Fig. 6):

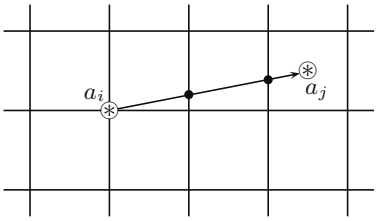


Figure 6: Case 1

• **Case 2:** here again, $a_i a_j$ does not intersect any stop line, but now the hypothesis $a_i.ignition + \delta spread(a_i, a_j) \geq \Delta$ holds. Therefore, the fire will not reach a_j along the edge $a_i a_j$. Thus, we cannot state that a_j is a propagating vertex unless the fire reaches it through another path. So, its status being temporarily undetermined, it is represented in Figure 7 by the symbol $\odot$.
Now, we must pay attention to the ending point of the fire along the edge $a_i a_j$. Using the calculation method described above, we can generate a new vertex, say a_p, such that $a_p.ignition = \Delta$. In particular, this implies that the last segment concerned in $a_i a_j$ will be partially traversed on the prorata of the allowed time.
Finally, note that a_p is a final vertex which is represented in Figure 7 by the symbol $\otimes$.

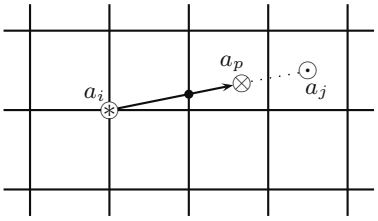


Figure 7: Case 2

• **Case 3a:** we suppose that $a_i a_j$ intersects at least one stop line which is represented in Figure 8 by a double line. Among the different intersections between $a_i a_j$ and one or several stop lines, we consider the one, say a_q, which is the closest to a_i. Moreover, we also assume that the hypothesis $a_i.ignition + \delta spread(a_i, a_q) \leq \Delta$ holds. In other words, the simulation will not be stopped before the fire has reached a_q. What is more, the fire cannot by hypothesis cross the stop line. Therefore, we can state that a_q is a final vertex of $\mathcal{G}$.

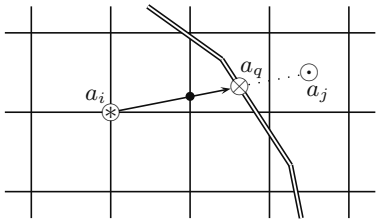


Figure 8: Case 3a

• **Case 3b:** As in case 3a, we suppose that $a_i a_j$ intersects at least one stop line and that a_q has the same status. Nevertheless, here the hypothesis $a_i.ignition + \delta spread(a_i, a_q) > \Delta$ is satisfied. This means that the simulation will stop before the fire has reached a_q. Hence, following the reasoning implemented for Case 2, we create a final vertex along $a_i a_j$, say a_r, such that $a_r.ignition = \Delta$ (Fig. 9):

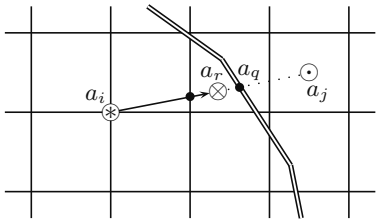


Figure 9: Case 3b

The procedure EDGE

The algorithms 1 and 2 given in the rest of the paper deal with specific sub-sets of $\mathcal{G}$, namely:

- $\mathcal{B}$ which is the set of propagating vertices.

- $\mathcal{V}$ which represents the *vicinity* of $\mathcal{B}$, i.e. the vertices which are not yet propagating vertices but for which at least one ignition time has been calculated.

- $\mathcal{C}$ which contains the final vertices.

Moreover, $\mathcal{L}$ is the set of the stop lines defined in the operations area. By abuse of language, we denote $a_i a_j \cap \mathcal{L}$ the set of all the intersection points between the edge $a_i a_j$ and the elements of $\mathcal{L}$. If necessary, these intersection points will be treated as vertices.

Under these bases, the procedure EDGE defined below handles fire spread along a given edge $a_i a_j$. The different cases dealt with precisely match those detailed previously (the portions of pseudo-code separated by the symbol "|" are mutually exclusive). Furthermore, it is important to note that provided the vertex a_j already occurs in $\mathcal{V}$, if its former ignition time is posterior to the one calculated, then this data is updated. Here, the objective is obviously to select the shortest paths in terms of spreading duration.

Finally, one must mention that the sets $\mathcal{V}$ and $\mathcal{C}$ are normally modified by the procedure EDGE.

Algorithm 1 : Fire spreading along an edge

EDGE$(\mathcal{V}, \mathcal{C}, \mathcal{L}, \Delta, a_i, a_j)$

$\delta \leftarrow \delta spread(a_i, a_j)$

case 1
case $a_i.ignition + \delta < \Delta$ **and** $a_i a_j \cap \mathcal{L} = \emptyset$:
if $a_j \in \mathcal{V}$ **then**
  if $a_i.ignition + \delta < a_j.ignition$ **then**
    $a_j.ignition \leftarrow a_i.ignition + \delta$
  end if
else
  $a_j.ignition \leftarrow a_i.ignition + \delta$
  $\mathcal{V} \leftarrow \mathcal{V} \cup \{a_j\}$
end if
|
case 2
case $a_i.ignition + \delta \geq \Delta$ **and** $a_i a_j \cap \mathcal{L} = \emptyset$:
let $a_p \in a_i a_j$ **such that** $a_p.ignition = \Delta$ **then**
  $\mathcal{C} \leftarrow \mathcal{C} \cup \{a_p\}$
end let
|
case 3.a and 3.b
case $a_i a_j \cap \mathcal{L} \neq \emptyset$:
let $a_q \in a_i a_j$
such that $|a_i a_q| = \min\{|a_i a_l|, a_l \in a_i a_j \cap \mathcal{L}\}$ **then**
  $\delta' \leftarrow \delta spread(a_i, a_q)$
  if $a_i.ignition + \delta' \leq \Delta$ **then**
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{a_q\}$
  else
    let $a_r \in a_i a_j$ **such that** $a_r.ignition = \Delta$ **then**
      $\mathcal{C} \leftarrow \mathcal{C} \cup \{a_r\}$
    end let
  end if
end let

Global propagation algorithm

The global propagation algorithm, specified hereafter, uses the procedure EDGE and is based on the principles of the Dijkstra's shortest path algorithm. It proceeds in four different phases:

- During the initialization phase, the meteorological data, the initial fire contour, the global duration of the simulation and the fire fighters actions are entered manually.
All the centers and the corners of cells occurring inside the initial contour are defined as propagating vertices with an ignition time equal to zero. The set $\mathcal{V}$ and $\mathcal{C}$ are initialized to the empty set.

- The procedure EDGE is then successively invoked for all the vertices of $\mathcal{B}$. This elaborates the initial value of $\mathcal{V}$ and $\mathcal{C}$. Note that in the second case this may be due to the fact that fire fighters actions can very well be defined close to the initial fire contour.

- The main iteration block calculates the fire global propagation and therefore elaborates the resulting contour.
This phase consists of repetitively choosing in $\mathcal{V}$ an element, say d, of minimum value in terms of ignition time. Then, d is considered to become a propagating vertex. It is therefore removed from $\mathcal{V}$ and inserted in $\mathcal{B}$.
Next, for all element of $\mathcal{N}(d)$ which is not in $\mathcal{B}$, the procedure EDGE is invoked which gradually leads to modify $\mathcal{V}$ and $\mathcal{C}$. This sequence is repeated as long as there exists in $\mathcal{V}$ at least one vertex whose ignition time is less than Δ.

- At last, the envelope curve encompassing $\mathcal{C}$, i.e. the final contour, is displayed.

Algorithm 2 : Fire spread global calculation

FIRE PROPAGATION

initialization phase:
$meteorological\_data\_entry$
$\Delta \leftarrow simulation\_global\_duration$
$\mathcal{B} \leftarrow centers\_and\_corners\_in\_the\_initial\_contour$
$\mathcal{L} \leftarrow stop\_lines\_entry$
for all $b \in \mathcal{B}$ **do**
  $b.ignition \leftarrow 0.0$
end for
$\mathcal{V} \leftarrow \emptyset$
$\mathcal{C} \leftarrow \emptyset$

# first calculation of $\mathcal{V}$ and $\mathcal{C}$:
for all $b \in \mathcal{B}$ **do**
  for all $n \in \mathcal{N}(b) \setminus \mathcal{B}$ **do**
    EDGE$(\mathcal{V}, \mathcal{C}, \mathcal{L}, \Delta, b, n)$
  end for
end for

# fire spread global calculation:
while $\min\{a.ignition, a \in \mathcal{V}\} < \Delta$ **do**
  let $d \in \mathcal{V}$
  such that $d.ignition = \min\{e.ignition, e \in \mathcal{V}\}$
  then
    $\mathcal{V} \leftarrow \mathcal{V} \setminus \{d\}$
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{d\}$
    for all $m \in \mathcal{N}(d) \setminus \mathcal{B}$
      EDGE$(\mathcal{V}, \mathcal{C}, \mathcal{L}, \Delta, d, m)$
    end for
  end let
end while

final contour display:
display-envelope-curve$(\mathcal{C})$

IMPLEMENTATION AND ASSESSMENT

The algorithm introduced in this paper has been implemented with the C++ programming language under the GIS GeoConcept. Experimentations have been conducted using a laptop equipped with a processor Core I7 and 16 Gb RAM. Provided the size of the initial contour has a limited size (as for instance in Figure 11), the duration of a simulation does not exceed a minute for a propagation of 3 hours on the ground. This allows various scenarios to be tested in order to facilitate tactical analyses. Obviously, such an approach is possible only because fire fighters actions can be introduced in the algorithm.

Another important aspect is the assessment of the level of correlation between real forest fires and the results of simulation. For that purpose, we used archives related to past fires within a period of 15 years. In that way, a drawback lies in the fact that we only have at our disposal the ignition point and the final contour, i.e. intermediate contours are not represented. Furthermore, the accuracy with which the plots where drawn is uncertain and the same is true for meteorological conditions. Nevertheless, after careful research, some case studies could be carried out. Hence, we made a number of tests of which we give an example below.

So, consider on the left of Figure 10 the contour which is backed by the river bed. Note that the latter has not been crossed by the fire which traversed approximately 3720 meters along its main axis in around 3 hours and 40 minutes.

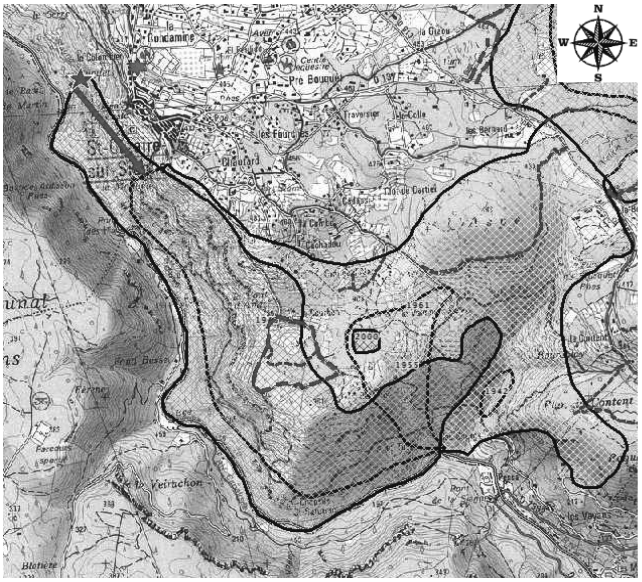


Figure 10: Contour of a past fire (archive extract of SDIS 06)

The corresponding simulation is given in Figure 11. A base line has been introduced in order to represent the protection of the village located at the East of the initial fire contour: although this information is not present in the Figure 10, we know that this action was implemented by the fire fighters. We can note some correlation between the two contours even though the propagation has been slightly underestimated on the East flank. Moreover, the river bed has been defined as a non-flammable area. This is the reason why it has not been crossed during the simulation, as shown in Figure 11.

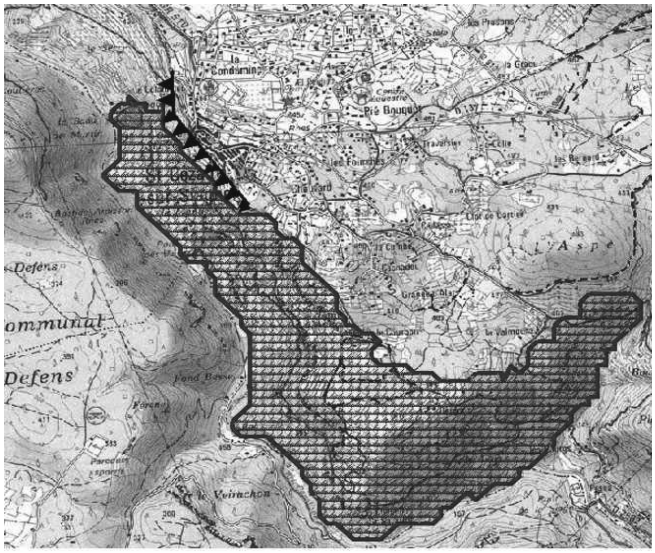


Figure 11: Result of a simulation

CONCLUSION

We have introduced in this paper an algorithm for forest fire simulation which incorporates fire fighters actions. Upon an experimental phase, we have reached the following global conclusions:

- Although they are not perfect, the results obtained already make it possible to extract information which could be useful for operation management.
- As one would expect, the model introduced in this paper is extremely sensitive to any variation of the aerologic data. Therefore, it is of huge importance that wind maps are as accurate as possible.
- The formula defined by equation (2) should be enriched so as to incorporate at least two additional parameters, namely vegetation and hygrometry.

From an operational perspective, this software system could find its place within a comprehensive approach. This would involve informal reasonings carried out by experienced officers simultaneously with maps specifying past fires (as far as they have occurred in the area concerned) and results of computer simulations. In case of converging analyses, reliable conclusions could then be drawn in terms of fire fighting strategy development.

ACKNOWLEDGEMENT

The author wishes to thank:

- The Fire Department of the Alpes-Maritimes (SDIS 06) for its logistic support.
- The Rocky Mountain Research Station (Missoula, Montana) of the US Forest Service which graciously provides the WindNinja software system.

- The reviewers of ISC'19 for refereeing the initial version of this paper and for they wise advices.

- Mr Bryan Lovell whose help has been extremely precious in improving the English style of the text.

BIOGRAPHY

Yves Dumond obtained his PhD in Computer Science from the University of Nice Sophia-Antipolis (France) in 1988. He is currently associate professor at the University Savoie Mont Blanc, where he has worked since 1990. His topics of interest include concurrency theory and forest fire modelling and simulation.

REFERENCES

Albini, F.A. and Chase, C.H. (1980). Fire containment equations for pocket calculators. *Research paper, RP-INT-268*, US Department of Agriculture, Forest service, Intermountain Forest and Range experiment station, Ogden, Utah, USA, 18 p.

Alexander, M.E. (1985). Estimating the length-to-breadth ratio of elliptical forest fire patterns. In *Proceedings of the 8th Conference on Forest and Fire Meteorology*, 29 April-2 May, Detroit, Michigan, USA, pp. 287–304.

Butler, B., Forthofer, J. and Wagenbrenner N. (2014). An update on the WindNinja surface wind modeling tool. *Advances in Forest Fire Research*, Domingos Xavier Viegas Editor, Coimbra University Press, pp. 54–60.

Dumond, Y. (2015). From forest firefighting doctrine to digital battlefield: a case study. *Disaster Prevention and Management*, vol. 24, no. 3, pp. 320–337.

Finney, M.A. (2004). FARSITE: fire area simulator-model development and evaluation. *Research paper RMRS-RP-4 Revised*, US Department of Agriculture, Forest Service, Rocky Mountain Research Station, Ogden, Utah, USA, 52 p.

Forestry Canada Fire Danger Group (1992). Development and structure of the Canadian Forest Fire Behaviour Prediction System. *Information Report ST-X-3*, Ottawa, Canada, 63 p.

Glasa, J., and Halada, L. (2007). Enveloppe theory and its application for a forest fire front evolution. *Journal of Applied Mathematics, Statistics and Informatics*, vol. 3, no. 1, pp. 27–37.

Griffith, J.D., Kochenderfer, M.J., Moss, R.J., Mišic, V.V. Gupta, V. and Bertsimas, D. (2017). Automated dynamic resource allocation for wildfire suppression. *MIT Lincoln Laboratory Journal*, vol. 22, no. 2, pp. 38–52.

Hernándes Encinas, A., Hernándes Encinas, L., Hoya White, S., Martín del Rey., A., and Rodríguez Sánchez, G. (2007). Simulation of forest fire fronts using cellular automata. *Advances in Engineering Software*, no. 38, pp. 372–378.

Homchaudhuri, B., Kumar, M., Cohen, K. (2013). Genetic algorithm based simulation-optimization for fighting wildfires. *International Journal of Computational Methods*, vol. 10, no. 6, 1350035, 28 p.

Hu, X., and Ntaimo, L. (2009). Integrated simulation and optimization for wildfire containment. *ACM Transactions on Modelling and Computer Simulation*, vol. 19, issue 4, pp. 1–29.

Hu, X., and Sun, Y. (2007). Agent-based modeling and simulation of wildland fire suppression. In *Proceedings of the 2007 Winter Simulation Conference*, 9–13 December, Washington, USA, pp. 1275–1283.

Kalabokidis, K., Athanasis, N., Gagliardi, F., Karayiannis, F., Palaiologou, P., Parastatidis, S. and Vasilakos, C. (2013). Virtual Fire: a web-based GIS platform for forest fire control. *Ecological Informatics*, no. 16, pp. 62-69.

Lopes, A.M.G., Cruz, M.G., and Viegas, D.X. (2002). Firestation – an integrated software system for the numerical simulation of fire spread on complex topography. *Environmental Modelling & Software*, no. 17, pp. 269–285.

McAlpine, R.S., Lawson, B.D., Taylor, E. (1991). Fire spread across a slope. In *Proceedings of the 11th Conference on Fire and Forest Meteorology*, 16–19 April 1991, Missoula, Montana, USA, pp. 218–225.

Noonan-Wright, E.K., Opperman, T.S., Finney, M.A., Zimmerman, G.T., Seli, R.C., Elenz, L.M., Calzin, D.E., and Fielder, J.R. (2011). Developing the US Wildland Fire Decision Support System, *Journal of Combustion*, vol. 2011, 14 p.

Ntaimo, L., Gallego-Arrubla, J.A., Gan, J., Stripling, C., Young, J., Spencer, T. (2013). A simulation and stochastic integer programming approach to wildfire initial attack planning, *Forest Science*, vol. 59, issue 1, pp. 105-117.

Peterson, S.H., Morais, M.E., Carlson, J.M., Dennison, P.E., Roberts, D.A., Moritz, M.A., and Weise, D.R. (2009). Using HFire for spatial modeling of fire in shrublands. *Research paper PSW-RP-259*, US Department of Agriculture, Forest Service, Pacific Southwest Research Station, Riverside, California, USA, 44 p.

Richards, G.D. (2005). An elliptical growth model of forest fire fronts and its numerical solution. *International Journal for Numerical Methods in engineering*, vol. 30, no. 6, pp. 1163–1179.

Rothermel, R.C. (1972). A mathematical model for predicting fire spread in wildland fuel. *General Technical Report INT-115*, US Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station, Ogden, Utah, USA, 79 p.

Sullivan, A. (2009). Wildland surface fire spread modeling, 1990-2007. 1: Physical and quasi-physical models. 2: Empirical and quasi-empirical models. 3: Simulation and mathematical analogue models. *International Journal of Wildland Fire*, vol. 18, no. 4, pp. 349–403.

Trunfio, G.A., D'Ambrosio, D., Rongo R., Spataro W. and Di Gregorio S. (2011), A New Algorithm for Simulating Wildfire Spread through Cellular Automata. *ACM Transactions on Modeling and Computer Simulation*, vol. 22, pp. 1–26.

Tymstra, C., Bryce, R.W., Wotton, B.M. and Armitage, O.B. (2009), Development and structure of Prometheus: the Canadian wildland fire growth simulation model. *Information Report NOR-X-417*, Natural Resources Canada, Canadian Forestry Service, Northern Forestry Centre, Edmonton, Alberta, Canada, 88 p.

Vakalis, D., Sarimveis, H., Kiranoudis, C., Alexandridis, A., and Bafas, G. (2004). A GIS based operational system for wildland fire crisis management I. Mathematical modelling and simulation. *Applied Mathematical Modelling*, vol. 28, Issue 4, pp. 389–410.

Flexible and Scalable Field Service with Condition Base Maintenance Models for Simulating Large Scale Scenarios

Gabriel G. Castañé
Helmut Simonis
Kenneth N. Brown
Insight centre for data analytics
Department of Computer Science
University College Cork
Cork, Ireland
{gabriel.castane, helmut.simonis,
ken.brown}@insight-centre.org

Cemalettin Ozturk
Yiqing Lin
United Technologies
Research Center
East Hartford, CT U.S.A
{OzturkC, LinY}@utrc.utc.com

Mark Antunes
United Technologies
Corporation
Winipeg, Manitoba
Canada
mark.antunes@otis.com

KEYWORDS
Discrete event simulation, Computer performance, Computer software, Operations research, Workforce Scheduling

ABSTRACT

Field service planning and scheduling research focuses on providing solutions to scenarios in which personnel travel between multiple locations carrying out different tasks specific to the field where they work. However, studying "what-if" activities in these scenarios is challenging for researchers, who have to deal with the impact on client service agreements, preferences of technicians when tasks are assigned, or unexpected faulty units. Therefore, simulation techniques are widely used to experiment and understand some critical aspects of workforce scheduling. Unfortunately, to mimic reality a high number of parameters are required within simulation models, raising scalability issues when the number of technicians, activities, and units increase over certain limits.
This work[1] presents a flexible simulation model to address field service planning and scheduling in large scale scenarios. The evaluation proves that the model executes in reasonable time for environments that consist of hundreds of thousands of locations in which technicians must perform diverse numbers of activities.

INTRODUCTION

Field service technician scheduling, an extension of the vehicle routing problem (Dantzig & Ramser, 1959), is a practical requirement for companies providing on-site services. Deliveries, visits, and maintenance and repair of installations are significant contributors to revenues and costs, and so more cost-efficient solutions promise major financial savings. The main objectives of the problem include minimising transportation costs, lateness penalties, and overtime costs, and reacting efficiently to unexpected call-outs. The core of the problem is assigning workers or technicians to tasks and routes, and is the focus of current research (Cheong et al., 2015; Kuo et al., 2014; Petrakis et al., 2012). Recent developments by UPS, focusing mainly on the routing part, are expected to save the company hundreds of millions of euros in coming years (Holland et al., 2017).
The field service maintenance problem can be modelled as geographic area with a set of units to be maintained, a set of locations for those units, a set of roads connecting those locations, and set of tasks to be performed on the different units. Each technician must complete a partial tour each day, starting at a particular location (home or depot), visiting a number of locations, performing the tasks for which they are qualified, and returning to their start location. These partial tours are scheduled in advance, accounting for travel times, task execution times, client service agreements, and daily working time limits. Violations of the service agreements, or requirements for additional hours, incurs additional costs, including payment penalties and overtime payments. These schedules are subject to deviations, which include variation on service times due to unpredictable factors, travel time delays, and the appearance of unplanned tasks, or client callbacks, that must be completed within short deadlines
The field service maintenance problem can be modelled as geographic area with a set of units to be maintained, a set of locations for those units, a set of roads connecting those locations, and set of tasks to be performed on the different units. Each technician must complete a partial tour each day, starting at a particular location (home or depot), visiting a number of locations, performing the tasks for which they are qualified, and returning to their start location. These partial tours are scheduled in advance, accounting for travel times, task execution times, client service agreements, and daily working time limits. Violations of the service agreements, or requirements for additional hours, incurs additional costs,

[1]This material is based upon works supported by United Technologies Corporation under UCC Collaboration Project and by Science Foundation Ireland under Grant No. 12/RC/2289 which is co-funded under the European Regional Development Fund.

including payment penalties and overtime payments. These schedules are subject to deviations, which include variation on service times due to unpredictable factors, travel time delays, and the appearance of unplanned tasks, or client callbacks, that must be completed within short deadlines.

The process of incorporating such deviations into already scheduled workplans is expensive, time consuming and usually inefficient. Poor decisions can cause significant increase in costs. This has led companies and academic researchers to produce detailed bespoke simulation models to study the impact of these deviations. However, the number of locations and technicians can vary from small numbers for small to medium enterprises to hundreds of thousands of locations and large fleets of technicians for larger companies. The size of the resulting problems can pose significant scaling challenges for simulation models and associated scheduling and routing software, and is the focus of current research.

In this paper, a flexible and scalable simulation model is proposed to enable the study of large scenarios for the field service maintenance problem. The main goals of our research are to provide models that allow users to simulate large diverse scenarios in reasonable time; to allow for flexibility in the workforce size, task allocation and routing; and to incorporate callbacks into the models which create unplanned tasks requiring altered but cost-efficient schedules. The main advantages of the proposed simulation model are listed below.

- Scalability: High scalability to support experiments with hundreds of thousands of locations and technicians.
- Flexibility: The models can be extended to support new tasks, service units, skills or capabilities.
- Low cost: the use of simulation models allows many more scenarios to be evaluated than incurring the cost of evaluating changes to the real environment.
- Performance: The running time of the models is sufficiently low to allow for simulation to be included in online scheduling, to evaluate proposed reactions to callbacks as they happen.
- Level independence: The models can be implemented in platforms independent of the technology, using, for example, bespoke tools in a low level programming language, or implemented as models for third-party simulation platforms.
- What-if analysis: Hypothetical situations which expand or contract the number of locations, units and technicians can be studied without cost to the company.
- Sandbox environments: The simulation models allow different decision procedures and policies to be studied without harming service agreements with customers.

In the results section, we show the performance impact of scaling up the number of technicians and locations in our models implemented using discrete event simulation and agent-based simulation.

MOTIVATION AND RELATED WORK

The work we present here is applicable for companies and public bodies that maintain safety critical units or emergency systems. Such deployments can involve well above tens of thousands of activities over wide and diverse areas. For example, (Antunes et al., 2018) present experiments for 5 depots in 3 administrative regions, handling 18 routes for 1,252 customer sites, and a total of 79,454 activities to be scheduled. There, it can be seen that solutions entail an underlying complexity that requires either abstracting the models into a high level objective function, or prunning the number of elements to be studied per experiment, thus allowing the executions to produce the results in a reasonable amount time.

Hence a key objective of our work is to build flexible fine-grained models that can scale to sufficiently large sizes that can represent real commercial deployments.

Low computational costs and cloud computing technologies have boosted the production of bespoke and general software tools developed with the goal of easing the production of work schedules, contingency plans, and the analysis of mobile workforce management. Multiple companies offer their clients production services to perform such analysis. Foremost among these in mobile workforce planning are IFS, ClickSoftware, VERISAE, ORTEC, QUINTIQ, Insiris, Verisae, vWork, or AEROMARK. However, these have features that impede industry from adapting these tools. Foremost among these are high cost of licensing, lack of flexibility to model specific problems, hidden APIs inaccessible to the client to adapt the model to their needs, or company security conflicts with tools that operate in the Cloud. As consequence, most companies develop bespoke solutions in spite of the time and economic costs associated with these.

Multiple techniques have been studied in the field service domain and simulation is widely adopted for it capability to mimic reality. For simulation experts, the main aim is to design and decompose the scenarios into a set of virtual elements which encompass all the relevant. Foremost among these elements in the field service area are technicians, locations, clients, tasks, travels, units, instance specific data, and the behavioural features of virtual elements and their interactions with other model elements. Thence, researchers and industry designed different models tailored to the input data and their needs supported by simulation frameworks. This goes from simple cases, e.g. workforce scheduling in a specific fast food company (Hueter & Swart, 1998), to more general food production scenarios, but which sacrifice details in order to achieve generality (Weng et al., 2017). (Jahangirian et al., 2010) give an extensive summary of a variety of different business application simulation case studies.

According to (Negahban & Smith, 2014), only a 96.5% of papers in simulation "*present models and studies addressing a narrow set of manufacturing problems or single-use applications of discrete event simulation*". Only a number of

Table I
HIGH LEVEL SIMULATION FRAMEWORKS SUPPORTING PROCESS MODELING, GIS, TO CREATE MODELS ON FIELD SERVICE

Name	Organisation	Language	Last Release	Description	Reference
DESMO-J	University of Hamburg	Java 8	Mar 2017	Java extension for discrete event simulation models	http://desmoj.sourceforge.net/
SSJ	University of Montreal	Java 8	Oct 2018	Java library for stochastic simulation and programming discrete-event simulations	http://simul.iro.umontreal.ca/
JaamSim	JaamSim Software Inc.	Java 8	Sep 2018	JaamSim is a Java based discrete-event simulation environment	http://jaamsim.com/
JSL	M. Rossetti	Java 8	April 2012	The JSL (a Java Simulation Library) is a simulation library for Java. Grade based project	http://sites.uark.edu/ rossetti/java-simulation-library/
Sim.JS	Maneesh Varshney	JavaScript	Jun 2011	JavaScripts based DES	http://simjs.com/
SimPy	Team SimPy	Python	Nov 2016	SimPy is a process-based discrete-event simulation framework based on standard Python	http://simpy.readthedocs.io/
r-simmer	Iaki Ucar, Bart Smeets	R	Jan 2019	process-oriented and trajectory-based Discrete-Event Simulation (DES) package for R	http://r-simmer.org/
SIMPROCESS	CACI Inc.	Bespoke	Sep 2018	SIMPROCESS integrates Process mapping, hierarchical event-driven simulation, and activity-based costing (ABC) into a single tool	http://simprocess.com/
ExtendSim	Imagine That!	Bespoke/Portable to DLL	Oct 2018	ExtendSim enables you to model continuous, discrete event, discrete rate, agent-based systems	http://extendsim.com/
Arena	Rockwell Automation	SIMAN	Dec2016	Flow Chart Modeling and event simulation	http://www.arenasimulation.com/
Simul8	Simul8 Corporation	bespoke/visual logic	Mar 2016	Process flowchart and simulations	http://www.simul8.com/
AnyLogic	AnyLogic	Bespoke/Java	Jun 2018	Agent Based, System Dynamic and Discrete Event Modeling	http://www.anylogic.com

authors describe the issues associated with the development of generic simulation models that are reusable for wider applications (Anglani et al., 2002; Huang et al., 2007; Park, 2005)
An exploration on the available simulation tools have been also performed. Special emphasis was taken on the different types of features that they support to build models as well as the capabilities to create models simulating processes to support Field Service research. The results are shown in Table I.

DESIGN OF SCALABLE SIMULATION MODEL

The design has been developed to address generality and scalability by using Unified Modeling Language - UML - 2.5 (UML, 2017). Therefore, it can be implemented in any language or by using a specific simulation framework.
Figure 1 depicts a class diagram in which the attributes and classes have been omitted in this paper for clarity. Moreover, in this section class names are shown highlighted in **bold**, and the relations between classes are presented as *emphasized* text.
The **SimManager** class represents an *abstraction* of a **REST API**. Restful services enable the possibility of coupling another system to manage the simulations - simulation optimisation loop - or complex architectures in which simulation is provided as a service. Within this class, time events and inputs required to start a simulation execution are managed. Inputs must be complying with the data model described in the Figure2 and developed in JSON format.

A **City** is *composed of* **Building**, and this can be one of both, **Depot** or **BuildingClient**. The latter can be seen as a set of **Units** that must be maintained, installed, or repaired.

In order to consider a set of activities regarding Condition Base Maintenance over a number of sensorized units, a degradation model is included into each **Unit**. These can be **DiscreteUnit**, if the degradation is function of number of uses - e.g. vending machines, photocopy machines; or **ContinuousUnit**, in case the degradation is due to the time that it is working, such as the case of a turbine or bulbs. Moreover, if a finer grain required, this degradation can be executed in a **Component** bases of a given **Unit**. The utilisation of each **Unit** and a lack of maintenance might end with a **Callback**, that is a task *created on failure*.

A **Route** is an *assignation* of Technician Units. A **trans-**

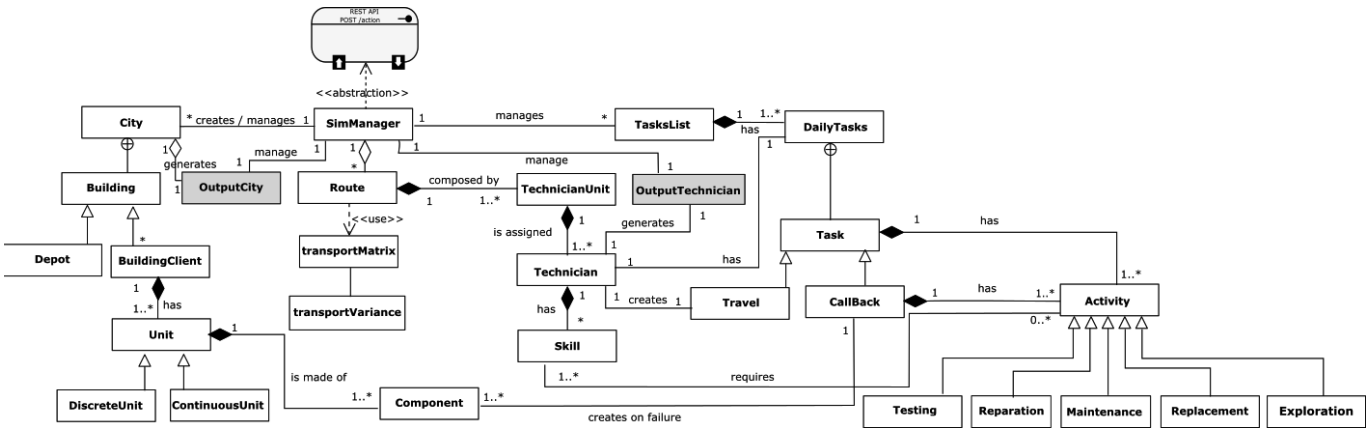


Figure 1. High level overview of class diagram of the proposed simulation model for Field Service and Workforce Scheduling problem.

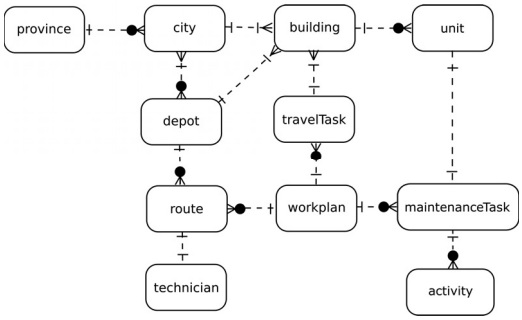


Figure 2. Data model for the exchange of information / initialization of the simulation.

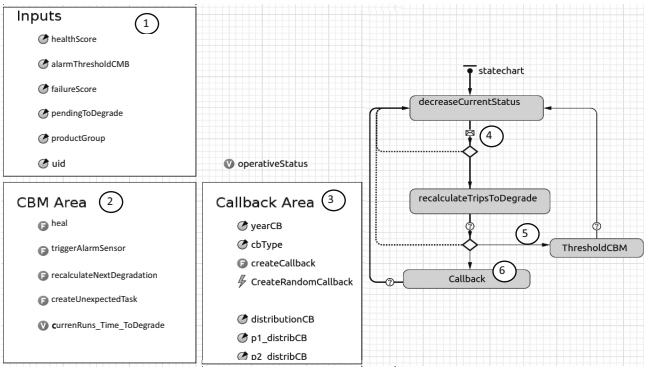


Figure 3. Agent based implementation for units, their degradation, sensorisation alarms, and the generation of unexpected tasks due to failures.

portMatrix has been chosen as a mechanism to define the distance between different geographical locations in which a **transportVariance** can be added due to external factors, such as: traffic issues or weather conditions. Moreover, a **Technician** *is assigned to* a **TechnicianUnit**, and each one of these *has* as set of **Skills**, used as a modifier of the time variance required to perfom a **Task**. These are encompassed into a **TaskList** that must be executed by a **Technician**, and **DailyTasks** are the group of **Task** selected as a part of **TaskList** to be executed in a specific day. In this class, methods associated to the selection and order in which the tasks are executed based on their constraints, due date, or agreements with clients can be considered, thence a relation with **Technician** is required.

A **Task** that must be performed in a location, is a composition of **Activities**, that can be: **Testing**, **Reparation**, **Maintenance**, **Replacement**, and **Exploration**. Moreover, there are another two types of tasks. One that it is created on failure due to the lack of maintenance, or due to an unexpected error - **Callbacks** - and **Travel** that are the tasks that represent a **TechnicianUnit** going from a location to another.

OutputCity and **OutputTechnician** have been highlighted to be outputs of the simulation process. Their responsibility are to collect statuses in the case of **Units**. and final locations and overtime in case of **Technician** class.

IMPLEMENTATION

The implementation of the diagrams shown in the design are partially done using the AnyLogic framework (Borshchev, 2013) and Java. This allows a seamless integration with external Java code from the AnyLogic engine, and it supports hybrid simulation methods, such as agent based and discrete event simulation modelling logic in the same model. However, diagrams described can be used on any other framework or language, therefore allowing the reader to replicate the simulation model and the results with independence to the language.

Units, Callbacks, and CBMs

Figure 3 depicts the behaviour of a generic unit/component. It consists on the following:

1) **Input parameters** where the data encompassed in the inputs allow to identify the unit, component, and to describe its behaviour. Parameters such as *healthScore*, *alarmThresholdCBM* and failureScore are used to control the degradation and either to trigger failures, or to enable an alarm in a sensorised component;
2) **CBM Area** encompass functions to calculate the degradation, the healing mechanism, and creation of a failures in case an sensor alarm is unattended are described in this area;
3) **Callback Area** in which random failures are described as a number of failures per year, and a distribution and types of callbacks that are triggered, loaded as inputs to reproduce the behavior of real units.

Moreover, the right part for the Figure shows a flowchart describing the ABS part of the Units/Components. Here, each utilisation in case of **DiscreteUnits** or periodically for **ContinuousUnits** decrease the status (4) of the variable in (2) *currentRuns_Time_ToDegrade*. In case this variable reaches 0, a new value is assigned and a point is decreased into the *healthScore*. If this reaches the *ThresholdCBM*, it generates an alarm in the corresponding sensor to be considered by the Technician in the next schedule (5). If it is not a sensorised unit/component, or if a previous alarm (5) has not be considered, a callback (6) is triggered and the variable *operativeStatus* is turned into *not operative*. Otherwise, the flow newly returned to *decreaseCurrentStatus* state.

Technicians and task executions

Technicians and tasks executions are composed by two parts, an agent based modelling - Figure 4 - and a process modeling part - Figure 5. In the left part of Figure 4 a set of inputs are collected that describe start and end times, coordinates where to start and suppose to end, and if the

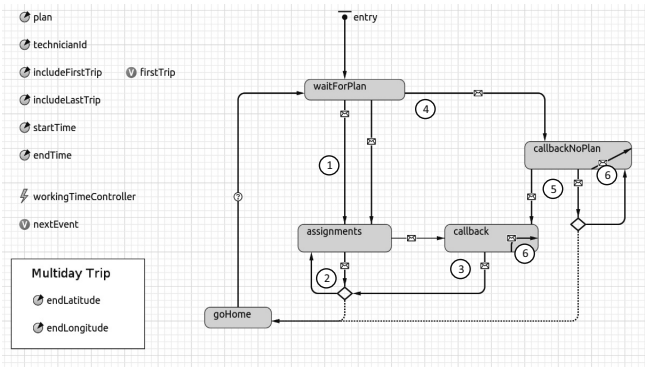


Figure 4. Agents based implementation describing technicians flow to attend the tasks

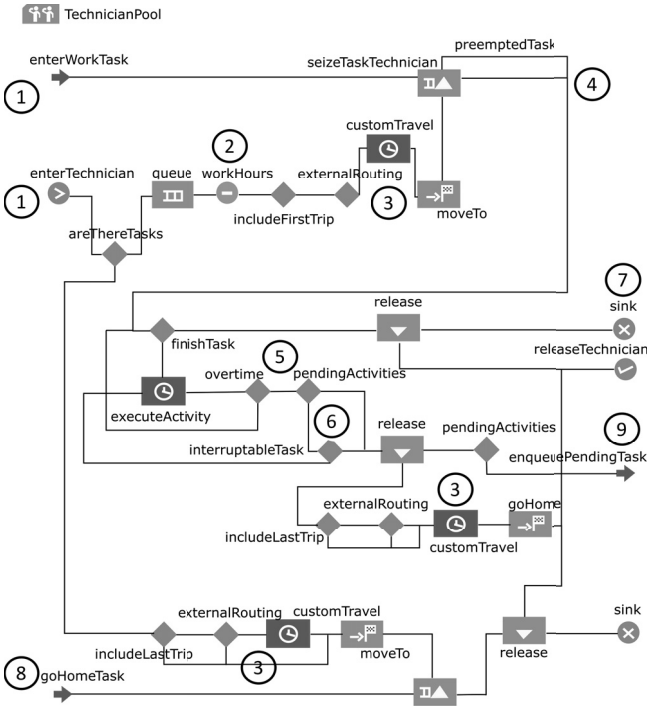


Figure 5. Process modelling implementation for tasks and technicians flow

first and last trip should be taken into account as working time.

The right part depicts the agent based part in the flowchart in which tasks life cycle from a Technician perspective is shown. Technicians start transitioning to **waitForPlan** state. When a new **DailyPlan** is received, it is assigned to the parameter **plan** and the transition (1) is triggered towards the **assignment** state. Here each task is loaded into the process modeling part, and sequentially executed according with the ordered plan received (2) until there are no more tasks, in such case the flow goes to **goHome** state, and subsequently to **waitForPlan**. In case during the **assignments** state, a callback in assigned to a technician, the task being executed is preempted and the emergency callback is attended (3). Moreover, and to model the behavior of technician that works attending callbacks and not performing scheduled tasks, (4) transition from **waitForPlan** to **callbackNoPlan**. If during the execution of a callback a scheduled plan is assigned to a technician (5) transition is followed to **callback** state. Finally, in case multiple callbacks are assigned to the same technician, these are preempted depending on their priority following the nested transitions (6).

The process modelling shows the behavioral aspects of technicians per task executed. Main aspects are:

1) Technicians of same technician unit are the resources of the pool accessing through **enterTechnician**, and each task is independently loaded by using **enterWorkTask** input from agent based **assignments**.
2) This component allows to control the working Hours for the techncians, blocking their travelling if time is not between *startTime* and *endTime* parameters.
3) These blocks are included to avoid travel times at constant speed given by AnyLogic, and to customise travel times adding traffic and weather conditions.
4) Once technician units are seized to a task, a task can only be preempted if a new one is inputted with a higher priority - *Callbacks*.
5) Activities are executed independently and once the *endTime* is reached, the time starts counting as overtime, and it is controlled inside the **overtime** conditional block.
6) For pending activities when the technician reaches *endTime*, the conditional blocks **pendingActivities** and **interruptableTask** decide if the technician can leave activities of a Task to the next day, or if all activities within the task must be finished before to stop the working day.
7) Technician and tasks are released at the same time.
8) To enable the measuring of the time that the trip home/to a hotel takes, in case this is considered as working time, and avoiding the technician stop working in the last location, a new task is created-*TransportActivity*, that is inserted into **goHomeTask**.
9) In case a task can be interrupted, the pending activities are enqueued to be post-scheduled.

EVALUATION

Two experiments are set up to study scalability and performance of models. A first scenario scales up the number of locations and units while the number of technicians and depots remains constant. In the second scenario, the locations and units is constant and and the number of technicians is varied. In both cases, the input data sets are build on real data to ensure consistency with the travel times and to avoid situations in which a travel task can not be performed due infeasible locations being assigned to some technicians. The time simulated is 24 hours, and the schedule for technicians is artificially generated to cover a period of a week. Within each of these schedules, a number of tasks is generated with diverse duration and a random

number of activities - from 1 to 10, using a 48-bit seed and using a uniform distribution.

Experiments have been executed in a Dell Laptop XPS 9370, in a CPU intel core i9-8950HK with 6 physical cores - 12 virtual, and 32GB of DDR4-2666MHz of main memory.

Scaling up the number of locations and units

For this experiment the data-set used belongs to the New York City (NYC) Open Data-set (Department Of Buildings, 2018). More specifically, these are units as per Department Of Buildings -DOB in short- of NYC to be maintained. As of 2018, DOB will require every one of the 160,000+ annual vertical elevator inspections to be submitted through DOB. The full data set of New York (***EntireNY***) encompass five areas that were simulated independently, each of these with 1 technician and 1 depot, and finally aggregated, hence, with 5 technicians and 5 depots accordingly.

	StatenIsland	*Bronx*	*Queens*	*Brooklyn*	*Manhattan*	*EntireNY*
Customer Location	661	4007	4852	7970	14759	31918
Units	1227	7575	9683	13980	43596	76061
Depots	1	1	1	1	1	5
Technicians	1	1	1	1	1	5

Figure 6 shows the results of the experiments using the data set of New York capturing the computational resources utilised to perform the experiments. These are the % of CPU stacked per each core - upper part of the figure - and amount of Memory - bottom part of the figure - required in each experiment to process the results of 24 simulated hours.

It can be seen in the % of CPU used a peak of utilisation during the first instants, ∼7s. This corresponds to the tasks executed to parse data and to create the structures required for the initialisation process of the simulation. After this, the % of CPU required to execute experiments with higher number of Units and Buildings scales up linearly. This can be seen in the following table:

	StatenIsland	*Bronx*	*Queens*	*Brooklyn*	*Manhattan*	*EntireNY*
runtime(s)	24	25	26	38	62	134
memory(Mb)	399	472	505	598	957	1430
buildings diff	–	3346	845	3118	6789	17123
units diff	–	6348	2108	4297	29616	32456

The rows **buildings diff** represents the difference between the number of buildings used as input of the simulation experiment of an area with respect to the next bigger area. This is analogous for **units diff** in the case of number of Units in the area.

The amount of memory required by the simulation engine to run is in the lower case ∼400MB. Furthermore, it also shows a linear behaviour with respect to the scalability with respect to the buildings. In addition, from the perspective of the units, the growth is even logarithmic, given that from Brooklyn to Manhattan there is a growth of 29.616 units and, from this area to the full data set of NYC a total of 32.456 units. The total amount of memory required is approximately ∼1.5GB in the experiment with the full data set.

The runtime between experiments increases in steps of ∼30s and the memory ∼500MB when the number of units is increased 30k as well, independently of the number of buildings..

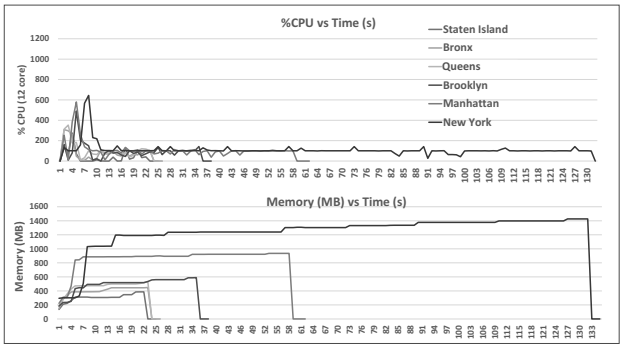


Figure 6. New York City Open Data-set evaluation scaling up locations.

Scaling up the number of technicians

The data set used in this experiment consists on the 15th of the most successful fast food chains franchises in United States of America and Canada. The reason behind the selection of these locations is the fact that $234 billion are generated by restaurant service industry (Oches, 2017). Locations are obtained from Garmin GPS Points of Interest database. The number of locations per business is listed next:

DataSource	*Units*	*DataSource*	*Units*
Arbys	3347	**Burguer King**	7269
Dairy Queen	5189	**Dominos Pizza**	3261
Dunking Donuts	8134	**KFC**	5630
Little Caesars	4019	**Mc donalds**	15474
Papa Johnś	3089	**Pizza Hut**	6672
Starbucks	11788	**Subway**	2213
Tacobell	6996	**Wendys**	6140

In addition, for a higher stress of the system, **Wallmart** with *22.212* new locations is added. This make a total of **136.663 total locations**, after cleaning noisy data from the data-sets in which we assigned 1 unit per location. This can be seen as surveillance units, refrigerators, or any other unit that requires maintenance activities.

The experiment can be sumarised as:

Technicians/Area	1	10	100	1000	10000
Technicians	63	630	6300	63000	630000
Depots	63	630	6300	63000	630000
Locations	136663	136663	136663	136663	136663
Units/Locations	1	1	1	1	1

Where technicians are scaled up in steps per experiment of an order of magnitude starting from 1 up to 10000 per state. A Depot is placed per area and a total number of 63 areas is included - one per state - in North America.

Figure 7 shows the computational resources required by the simulation models to run 24 simulated hours. The upper part depicts the execution percentage - on 12 cores - over time

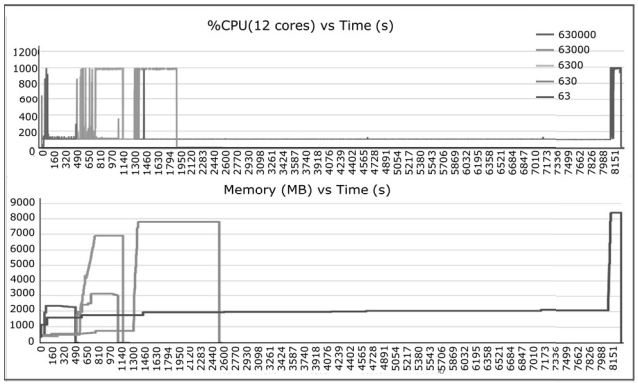


Figure 7. Restaurants data set scaling up technicians

- in seconds, while the bottom part does the analogous for the memory used - in MB - against time - also in seconds. It can be seen in the CPU execution that the data process is made in 10 over 12 cores, however, the long run processing is executed in only 1 core. This is a limitation imposed in the experiments to remove the benefit of the paralellization capabilities of the 12 cores during the runtime. AnyLogic provides only multicore executions when Montecarlo experiments are executed, in which multiple threads can be used in parallel, thus, reducing the execution time.

	63	630	6300	63000	630000
runtime(s)	490	1007	1138	1821	8155
memory(Mb)	2124	3013	6985	7699	8276
technicians diff	–	567	5670	56700	567000

It is clear that the simulations overhead is caused by the technicians models. Each component in the process modelling is translated by AnyLogic as an object, that is put in memory and that needs to interact with other objects. This is a heavy process for the system that partially overloads the machine. However, the increase of technicians is not linear, therefore, the last experiment takes approximate 2h to finalise. In this case, the reason is the increase of 567.000 technicians with respect to the previous experiment - 63000 technicians.

The peaks shown in Figure 7 at the beginning of every experiment are due to the reception and processing of the locations, the units, their creation and placement as objects in memory, and their initialisation of parameter.

It can be seen that there is a big memory step between the experiments 630 and 6300 technicians. This is most likely due to the AnyLogic black-box memory management system; to access to more comprehensive analysis is not allowed to end users. Furthermore, from 6300 to the 630k technicians the management of memory is improved as the difference is less that 2 GB for an increase of more than half a million technicians.

At the end of the execution with 630000 technicians, the computational and memory peak is due to the processing and storage of the results. In addition to the status of the units - health score -, and pending tasks to be executed, the system has been implemented to collect technicians status at the momentum of finalisation of the simulation (24h). This process collects information from technicians - tasks performed, tasks pending, final locations for all of them - and creates adequate structures in JSON format to be send to the process caller.

CONCLUSIONS

In this work we presented a flexible and scalable model for simulating field service units with sensorized components and their degradation. A design in UML 2.5 was completed and an implementation in AnyLogic and Java was performed using JSON format for communication with external code to trigger the executions.

Execution times demonstrate a strong performance. Very promising scalability results are shown in the AnyLogic implementation providing high performance in the execution. However, there might be significant overheads for components that are not used in run time. Therefore, a future work will be focusing an implementation using continuous time loop technique in a language as C++ or python for a comparison of performance and execution times with current model.

Moreover, given the promising results for the model and the implementation, we will explore an intelligent gateway to offer simulation on demand for end users.

REFERENCES

Anglani, A., Grieco, A., Pacella, M., & Tolio, T. (2002). Object-oriented modeling and simulation of flexible manufacturing systems: a rule-based procedure. *Simulation Modelling Practice and Theory*, *10*(3-4), 209–234.

Antunes, M., Armant, V., Brown N., K., Desmond, D., Escamocher, G., George, A.-M., Grimes, D., O'Keeffe, M., Lin, Y., O'Sullivan, B., Ozturk, C., Quesada, L., Siala, M., Simonis, H., & Wilson, N. (2018). Assigning and scheduling service visits in a mixed urban/rural setting. In *2018 ieee 30th international conference on tools with artificial intelligence (ictai)* (pp. 114–121). IEEE.

Borshchev, A. (2013). Anylogic 7: new release presentation, 4106–4106.

Cheong, M. L., Koo, P., & Babu, B. C. (2015). Ad-hoc automated teller machine failure forecast and field service optimization, 1427–1433.

Dantzig, G. B. & Ramser, J. H. (1959). The truck dispatching problem. *Management science*, *6*(1), 80–91.

Department Of Buildings, D. (2018). *Nyc data at work*. New York City, New York, United States: Department Of Buildings.

Holland, C., Levis, J., Nuggehalli, R., Santilli, B., & Winters, J. (2017). Ups optimizes delivery routes. *Interfaces*, *47*(1), 8–23.

Huang, E., Ramamurthy, R., & McGinnis, L. F. (2007). System and simulation modeling using sysml, 796–803.

Hueter, J. & Swart, W. (1998). An integrated labor-management system for taco bell. *Interfaces*, *28*(1), 75–91.

Jahangirian, M., Eldabi, T., Naseer, A., Stergioulas, L. K., & Young, T. (2010). Simulation in manufacturing and business: a review. *European Journal of Operational Research*, *203*(1), 1–13.

Kuo, Y.-H., Leung, J. M., & Yano, C. A. (2014). Scheduling of multi-skilled staff across multiple locations. *Production and Operations Management*, *23*(4), 626–644.

Negahban, A. & Smith, J. S. (2014). Simulation for manufacturing system design and operation: literature review and analysis. *Journal of Manufacturing Systems*, *33*(2), 241–261.

Oches, S. (2017). The QSR 50. https://www.qsrmagazine.com/reports/2017-qsr-50. [Online; accessed 13-02-2019].

Park, S. C. (2005). A methodology for creating a virtual model for a flexible manufacturing system. *Computers in industry*, *56*(7), 734–746.

Petrakis, I., Hass, C., & Bichler, M. (2012). On the impact of real-time information on field service scheduling. *Decision Support Systems*, *53*(2), 282–293.

UML, O. (2017). Unified modelling language version 2.5. unified modelling.

Weng, S. J., Gotcher, D., & Kuo, C. F. (2017). Lining up for quick service. the business impact of express lines on fast-food restaurant operations. *Journal of Foodservice Business Research*, *20*(1), 65–81.

WEB REFERENCES

http://desmoj.sourceforge.net/
http://simul.iro.umontreal.ca/
http://jaamsim.com/
http://sites.uark.edu/rossetti/java-simulation-library/
http://simjs.com/
http://simpy.readthedocs.io/
http://r-simmer.org/
http://simprocess.com/
http://extendsim.com/
http://www.arenasimulation.com/
http://www.simul8.com/
http://www.anylogic.com

AUTHOR BIOGRAPHIES

GABRIEL GONZÁLEZ-CASTAÑÉ is a researcher in the Insight Centre for Data Analytics in Cork. He holds a PhD in Computer science from University Carlos III of Madrid on the topic of energy aware cloud simulations. He has participated in several National Projects, leading a simulation workpackage in CACTOS FP7 project, and technical coordinator of the CloudLightning H2020 project. Currently works in a research project with United Technologies.

KENNETH N. BROWN Ken Brown is a Professor in University College Cork, a Principal Investigator in the Insight centre for data analytics, and a Funded Investigator in the the Confirm centre for smart manufacturing. He has over 30 years experience in applied artificial intelligence and optimisation for many different application domains.

HELMUT SIMONIS has been working on constraint programming for over 30 years. He was member of the CHIP project at ECRC in Munich, co-founder and technical director for COSYTEC in Orsay, and a principal research fellow at Imperial College London, as well as working for start-up companies Parc Technologies and CrossCore before joining Insight Centre. He has served as the President of the Association for Constraint Programming.

CEMALETTIN OZTURK is a Senior Research Scientist in United Technologies Research Center and his focus area is applications of combinatorial optimisation and simulation methods in manufacturing and service systems.

YIQING LIN is a Research Scientist in United Technologies Research Center. She has extensive experience in the area of Operations Research including continuous and combinatorial optimization, and discrete-event simulation for supply chain management, manufacturing model as well as field service operations.

MARK ANTUNES is Senior Regional Field Operations Manager in United Technologies Corporation.

AUTOMATION ROBOTICS AND DIGITAL TWINS

Intelligent Robotic Devices using Knowledge from the Cloud

P.J.S. Gonçalves, F.M.S. Santos
IDMEC, Instituto Politécnico de Castelo Branco,
Av Empresário, 6000-767 Castelo branco, Portugal.
email: paulo.goncalves@ipcb.pt

KEYWORDS
Intelligent Robots; Cloud Robotics; Knowledge Technologies; Intelligent Systems; Human-Computer Interaction

ABSTRACT

In this paper are presented several low-cost solutions to deploy Robots in selected environments and perform complex tasks, by using the Cloud. The Cloud can be used as a database or a powerful processor to extend the robots capabilities in the field. Intelligent Robots can take advantage of a distributed, web-based information system deployed in the cloud to perform high level tasks. This paper proposes a robotic control framework suited to be used by low-cost robots, performing teleoperated and/or autonomous tasks. The first part is dedicated to the development of an Android based robot control framework. This framework connects to specific low-level controllers that were developed for QuadCopters, wheeled and tracked mobile robots. The second part is dedicated to "place" the Android based robot in the cloud. There, the robot can perform Cloud based highly automated intelligent tasks in order to optimize their use and take best advantage of previous knowledge models, e.g., objects databases or 3D world models. Also, the robot can be controlled remotely using a classical teleoperation mode, using wi-fi networks. First experiments are presented when a tracked robot is performing surveillance tasks where its state can be changed to teleoperation/videoconferencing mode. The experimental results also show the robot interacting with a reasoning engine in the Cloud.

Introduction

Nowadays field robotics can be considered as a mature science, in the sense that exists in the market several types of robots for diverse applications. Such applications can go from surveillance, data collection, construction, agriculture, demining, and so on. In the meanwhile in recent years communication systems development ensured large bandwidth capabilities, while covering a large portion of the world territory. This increase of data gathering capabilities from machines enhanced a problem to the human operators, that nowadays will cooperate with robots. These large amount of data, often arriving in real-time, must be processed by humans if they want to perform adequate teleopeation commands, for example. A way to solve this problem is to leave to the robot some intelligent processing capabilities.
Artificial Intelligence have a long history in pursuing solutions for making intelligent robots with limited success, in the sense that commercial intelligent robots (capable of performing intelligent tasks) have not gained considerable comercial market. This issue can be explained by the difficulties in obtaining suitable broadband knowledge models. However, in recent years there is been a considerable effort by the research community to obtain knowledge representations in a machine readable format. Ontologies have gained its place in robotics, and are commonly used as knowledge models. Notable EU research projects, RoboEarth [1] and RoboHow [2], in this field shows the path been walked by european researchers. Being aware of this, IEEE [3] started recently to develop a standard in this filed that is expected to be finished in 2015.
However research in intelligent robotics requires top-end equipment, with high perception, storage and processing capabilities, to process the incoming perception data, and reason based on this data and the stored knowledge models. This setup is not in-line with the stated previously, i.e., to leave to the robot some intelligent processing capabilities, specially if we are thinking in low-cost robots for everyday use. Moreover, the knowledge model can rapidly increase with the data gathered by the robot.
The previously stated issues led robotics developments to use Cloud Services, either to store knowledge models that can be accessed by robots, to process the data gathered by robots, and even to reason based on the knowledge models and the data gathered.
The term Cloud Robotics was presented by James Kuffner from Google [4] and he envisioned that it could make robots lighter, cheaper, and smarter, and that Robots could "offload CPU-heavy tasks to remote servers, relying on smaller and less power-hungry onboard computers".

This paper proposes a general framework suited for low-cost Android based robots to operate with the Cloud. These robotic agents can be used:

1. to share perceived data in the Cloud,

2. for video conferencing services,

3. for remote teleoperation,

4. to fetch the Cloud based knowledge model,

5. to deliver intelligent processing to the task.

This paper is organized as follows. Section 'Mobile Robots', presents the arduino based robots developed and its low-level controller. Section 'The Android Framework', presents the developed Android framework for controlling the robot and enable connection to the cloud. In the following sections is presented, how the Android Framework can interact with the Cloud, and also how this application is aligned with a robotic ontology. The paper ends with a discussion on the presented results and draws some conclusions and future work directions.

Mobile Robots

Next sub-sections describe the hardware setup developed, i.e., the Tracked, Wheeled, and QuadCopter Robots. Also, the kinematics of the robots and the corresponding low-level Arduino (http://www.arduino.cc/) based controllers, are presented. This open source, cheap, easy to use, and largely used worldwide, rapidly was used in robotics. In [5] is presented a recent survey on the use of this type of controllers in mobile robotics, and some results were presented for telepresence robots [6], for use in indoor environments.

Tracked/Wheeled Robots

Kinematics is a fundamental mathematical tool to control robots. It allows robots to perform accurate positioning tasks, and also enable localization in the environment. For the Tracked/Wheeled Robots, consider the reference frames in figure 1, where $W = \{O_w; X_w, Y_w\}$ is a fixed (inertial) frame assumed to be the World coordinate frame, and $R = \{O_r; X_r, Y_r\}$ the robot frame.

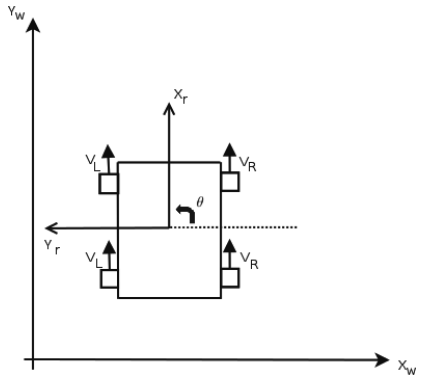
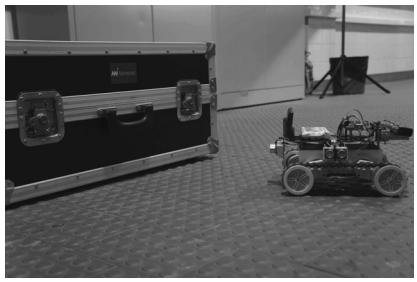


Figure 1: Inertial and Robot frames for the Tracked/Wheeled Robot.

(a) The Wheeled Robot.



(b) The Tracked Robot in a Surveillance Task.

Figure 2: Mobile robots for indoor environments.

The rotation matrix $R \in SO(3) = \{R \in \Re^{3x3} \,|\, R^T \cdot R = I,\, det(R) = I\}$ represents the transformation of the robot frame to the inertial frame. This rotation matrix R, depends on the θ parameter depicted in figure 1, and is given by:

$$R(\theta) = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 \\ -sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

The robot kinematics is described as [7]:

$$\begin{pmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\theta} \end{pmatrix} = R(\theta) \cdot \begin{pmatrix} \dot{X}_w \\ \dot{Y}_w \\ \dot{\theta} \end{pmatrix} \quad (2)$$

Equation 2 shows the relation from the world and robot frames velocities. A possible solution [7], amongst others, to obtain the robot position in world coordinates, is to differentiate its velocities. The relation from the wheel velocities V_L and V_R, respectively from the left and right wheels, is obtained from the structural dimensions from the robot [7].
The developed Wheeled Robot, depicted in figure 2(a), was build with low-cost materials. Is equipped with four DC motors, one L298 based drives (for each pair of motors), one ultrasound sensor, an Arduino board with a bluetooth shield, two LI-PO batteries (for the arduino

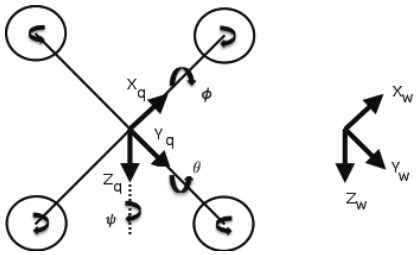


Figure 3: Inertial and Robot frames for the QuadCopter Robot.

for the motors). In the front side of the Robot is placed an Android based smartphone, for wi-fi communication. The developed Tracked Robot, depicted in figure 2(b), is based on the DAGU rover 5 chassis, from (http://www.dfrobot.com). The chassis is equipped with two DC motors, two L298 based drives, two encoders, three ultrasound sensors, an Arduino board with a bluetooth shield, two battery packs (on for the arduino and other for the motors). In the front side of the Robot is placed an Android based smartphone, for wi-fi communication.

QuadCopter Robot

For the QuadCopter Robot, consider the reference frames in figure 3, where $W = \{O_w; X_w, Y_w, Z_w\}$ is a fixed (inertial) frame assumed to be the World coordinate frame, and $Q = \{O_q; X_q, Y_q, \}$ the robot frame. The rotation matrix $R \in SO(3) = \{R \in \Re^{3x3} \mid R^T \cdot R = I,\ det(R) = I\}$ represents the orientation of the robot frame to the inertial frame. This rotation matrix R, using the Euler angles $\Phi = [\phi, \theta, \psi]$ depicted in figure 1, is given by:

$$R(\phi,\theta,\psi) = \begin{pmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi - s_\psi c_\phi & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi c_\theta & s_\psi s_\theta s_\phi + c_\psi c_\phi & s_\psi s_\theta c_\phi + c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{pmatrix} \tag{3}$$

where $c = cos(.)$ and $s = sin(.)$. The robot kinematics is described as [7]:

$$\dot{p} = R(\phi,\theta,\psi) \cdot v \tag{4}$$

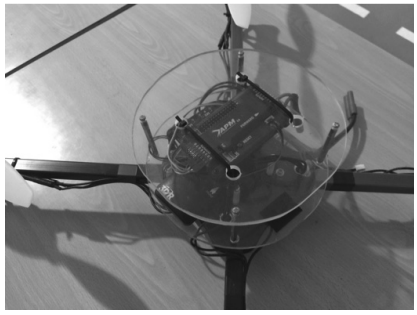
$$\dot{\Phi} = \Gamma \cdot \omega \tag{5}$$

where $v \in \Re^3$ and $\omega \in \Re^3$ are the linear and angular velocity of the QuadCopter relative to the inertial frame W, and expressed in Q, $p \in \Re^3$ is the quadcopter position expressed in W and $\Phi \in \Re^3$ is the QuadCopter attitude. Matrix $\Gamma \in \Re^{3x3}$ is given by:

$$\Gamma = \begin{pmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{pmatrix} \tag{6}$$

where $t = tan(.)$.

(a) Full view of the robot.



(b) Zooming the control boards.

Figure 4: The IPCB QuadCopter Robot.

A four motor MultiCopter, QuadCopter, was developed from scratch in our lab, and is depicted in figure 4. The aluminium structure was designed to ensure low weight, and the robot is equipped with 4 motors, the corresponding controllers, the 4 propellers, a flight stabilization control board, with a gyro, based on the Atmega168PA processor, an ArduPilot enabled with Bluetooth and Wi-Fi communications, and one Li-Po battery.

Arduino based Low-level Control

This sub-section presents the main functions developed and implemented to control the Arduino based low-level controller. In figure 5 is depicted the interaction between the robots and the teleoperation module (Android Application), that is done using an Arduino Mega board with a Bluetooth shield.

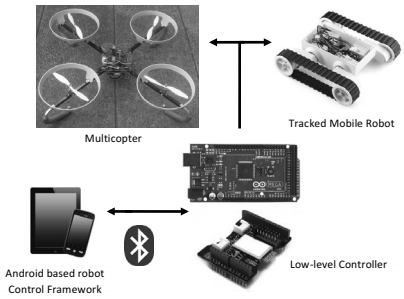


Figure 5: The Low-level Control Framework.

The main functions needed to control the Wheeled and

the Tracked Robots, i.e., the ones used to receive/send commands from/to the teleoperation module. The low-level controller receives a command for the motors velocity and direction, computes the motor speeds to be sent to the motor drives, reads the encoders, and computes the robot cartesian position and its heading to be sent to the teleoperation station. The function *CommandReceived* serves as a communication watchdog, and is called every 5 seconds to ensure that the robot have received some command. Other functions were implemented for safety, power consumption, PID controllers, trajectory planning, ultrasound sensors readings, and so on.
The low-level controller QuadCopter functions, are similar to the ones for the ground robots, but taking in mind that the command is based on the Throttle, the Pitch, Roll, and Yaw, received from the teleoperation module (Android Application). Please note that the QuadCopter have an additional flight stabilization board, with a gyro and based on the ATMega168A processor, that receives the motors Throttle, the Pitch, Roll, and Yaw.

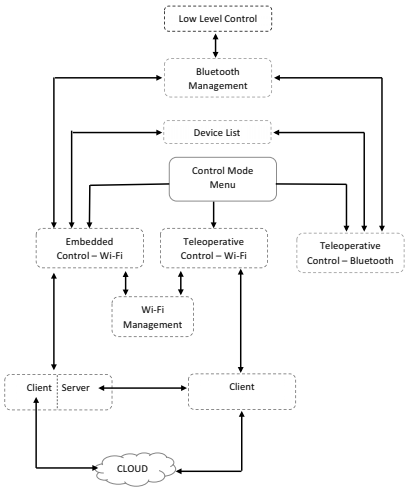


Figure 6: The Class Diagram of the developed JAVA application.

The Android framework

Several android frameworks have been developed to control various types of robots. The main drawback of all of them is that were developed for a particular platform. After a search in Google Playstore (https://play.google.com/store), were found the following applications, that enable Arduino Control via Bluetooth interface: Arduino BT Joystick, MobotBT-Car, Bluetooth Controller, RC Bluetooth RemoteControl, BluCar, Arduino Bluetooth RC Car, BlueBots, Robot Control, Bot. Only few ones have a wifi interface, but only for supervision, e.g., Mover-bot, BT BotControl. Recently, two major commercial hits are using Apple IPhone and/or Android to control the robots via wifi, AR.DRONE (http://parrot.com/). Having the first one the capability to be remotely controlled in the Cloud, and consequently able to perform teleoperation/telepresence.
A contribution of the presented work is an application that can be easily deployed to several types of robots. Indeed, the application, with few setup settings can be setup for quadcopter, tracked and wheeled robots, like the ones presented previously. This settings depends on the type of robot and are mainly related to the dimensions of the structure, types of actuators and sensors.
Figure 6 presents the main classes in the developed application. It is presented that he *Control Mode Menu* enables the application to behave in three different modes:

- *teleoperative mode, via bluetooth*, i.e., a local command;
- *teleoperative mode, via wifi*, i.e., other type of local command;
- *embedded control, via wifi*, i.e., when the robot have an embedded hardware running the Android Apllication.

It can be depicted from fig. 6 that the *Bluetooth management class* is responsible for the intereaction with the Arduino low-level controller. Also several robots are possible to be controlled, the ones that appear in the *device list*. The *Wifi management* class, is responsible for handling this type of messages. The classes *Client* and *Server* are responsible for the interconnection from the application to the Cloud, via the wifi-network.
An important class in the application is the *Embedded Contol - wifi*, that enables the third use case. This type of behaviour, as presented previously, assumes a hardware, e.g., a smartphone, running the android application. Please note one important feature of the system, that is the capability of being in *Client Mode*, or in *Server Mode*. When in *Client Mode*, the application can be connected to a router, and consequently be a Client when connected to the Cloud. When in *Server Mode*, the application will connect to a Client, that enables the Connection to the Cloud or a nearby hardware running the Android application running the *teleoperative mode, via wifi*.
In this way, robots are now connected to the Cloud and can preform teleoperation tasks with the same Android application used for the embedded control, or the teleoperative control via bluetooth.
In code listing 1 is described the beahaviour of the class *EmbeddedControlWifi*. It is associated to an activity connected to the class layout. The method *getIP()* gets the IP address of the hardware running the application. *connected()* receives the infomation of all Bluetooth devices nearby. *sendImage()* performs the management of the video stream. *onSharedPreferenceChanged()* gets

the user defined settings for the platform and connections. *UpdateMethod()* performs the messages transitions of the two handlers, e.g., *wifiMessageHandler* and *bluetoothMessageHandler*.

Listing 1: The EmbeddedControl JAVA class.

```
Package core.control.robotcontrol;
public class EmbeddedControlWifi extends Activity implements
      OnSharedPreferenceChangeListener{
        void getIP()
        void connected(BluetoothSocket socket, BluetoothDevice
             device)
        void sendImage(byte[] data)
        void onSharedPreferenceChanged( SharedPreferences prefs,
             String key)
        void UpdateMethod()
        void wifiMessageHandler(String msg)
        void bluetoothMessageHandler(String msg)
}
```

In figures 7 and 8 are depicted the Layout of the telepresence/videoconferencing Android Application, from the Tracked and QuadCopter robots. It can be seen the background photos, i.e., the video streaming from both robots. For the Wheeled and Tracked Robots, the user have access of the following features: one planar joystick, a camera focus button, a flash enabler button, a snapshot button, and a button to switch between the manual and autonomous modes. For the QuadCopter, the user have access of the following features: two planar joysticks to handle throttle, roll, pitch and yaw, and several buttons to: camera focus, snapshot, flash enabler, lift-off, landing and emergency.

Figure 7: The Wheeled and Tracked Robots Teleoperation Application.

Figure 8: The QuadCopter Teleoperation Application.

Robots in the Cloud

The increasing availability of broadband communication networks, high capacity computer memory and processing services, amongst the large amount of on-line data, e.g., relevant wikipedia and other sources, allows and suggests the use to knowledge grounding and embedded cognition for robotic agents. Cloud repositories of previously captured or real-time sensor data with semantic annotations allows the use of Artificial Intelligence algorithms with robotic agents, using knowledge models like the presented in the previous section.
Having this in mind, the Intelligent Robotic System proposed have the following general features:

1. the embedded robotic agent processing ability is a low-level interface for sensor transmission and command reception, although it can be scaled to full autonomous behaviour, depending on the user/owner of the agent,

2. the intelligence of the robotic agent can be local or in the Cloud, via a web service,

3. the Cloud-based memory comprises past robot perceptions and/or human sensory data, gathered in a Knowledge Model,

4. a robotic agent can use the Knowledge Model, for ongoing or future tasks completions,

Several systems try to enable Cloud Robotics for specific robotic tasks [1] and in this paper is proposed a generic system that enables this top-end architecture to common Android/Arduino based robots: tracked, wheeled and flying. Such a system is depicted in Fig. 9 where is observed the robotic agents in the right, the Cloud with its server having a Knowledge Model and a Reasoning engine, and on the left the presented services that can be obtained from the Cloud using this framework, Teleoperation and VideoConferencing. Please note that all connections are bi-directional, enabled by broadband communication networks, e.g., wireless internet.

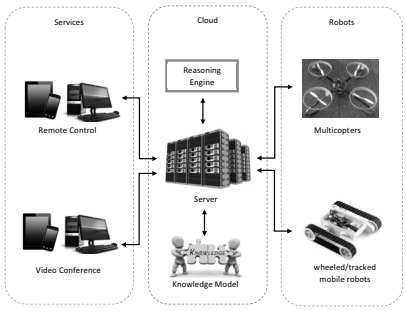


Figure 9: The Robots interfacing with the Cloud and Services.

Knowledge Representation

Recently two notable European Union research projects have undergone serious efforts to enable robots with cognition capabilities and my using th Cloud to that effort. The projects are RoboEarth [1] and RoboHow [2]. Similar efforts are ongoing in the USA [8] and Japan [9].
Knowledge models represents the real world and its interactions, namely between machine(s) and/or human(s). Within this understanding and to enable communications between humans and machines, Knowledge must be expressed in a machine readable format. Ontologies are an excellent way to represent knowledge, taking into account recent developments and its ability to represent common knowledge in a machine-readable format, [10], for better Human–Robot Interaction (HRI) interfaces.
When building Knowledge models, i.e., ontologies, one must be aware that robots are evolving and moving from structured to unstructured environments. And in this new era, there is a constant need for capable HRIs for cooperative work environments. Moreover, the general public are expecting robots to became part of their lives, e.g., the Roomba vacuum cleaner (http://www.irobot.com/), or advanced toys, such as the commercial hits ROMO (http://romotive.com/) or AR.DRONE (http://ardrone2.parrot.com/).
A successful design of future (service) robots will be based on detailed knowledge of applicable technologies and methodologies, and must tackle the following challenges:
strict requirements regarding personnel and operational safety; issues associated with autonomy; navigation in structured and unstructured environments; human coexistence in the same physical spaces; collaborative manipulation and task sharing, limiting the use of robots for carrying personal service tasks.
To accomplish these challenges, and following[11], first a core ontology [10] must be used to represent specific knowledge (definition of the robot, type of robots to use, components to use, and so on). Second, but not less important, for the *end-user*, an application domain ontology needs to be defined for the robot's work environment, e.g., knowledge of the household environment.
The first task of the *developer* of robotic systems is to identify all the hardware requirements needed for the system (what type of robot, sensors, communications, and son on). The second task, not independent from the first task, is to identify the functionalities needed for the robotic system taking in special account the task to performed and the environment where it will be performed. This first two tasks largely benefits from existing standards and knowledge defined in ontologies. Passing the first two tasks, is necessary to build the robotic system itself, i.e., in terms of physical components and software. For that the robotic platform must be built with sensors and actuators, and also be defined the software drivers of the defined components for the system, as well the operating system. These tasks were implemented in Sections 2 and 3, using low-cost platforms enabling Cloud Robotics.
The next phase is related to the HRI, i.e., how the *end-user* interacts with the robotic system. This fourth task is dedicated to build the software control system from the defined constraints in the previous three tasks. This software can be built based on existing libraries, e.g., ROS (http://www.ros.org), or specially designed for the purpose of the robotic system being developed, in our case using the Android framework. In this task of the robotic system development, ontologies play a key role on the definition of the robot control software architecture, taking into account reasoning platforms based on the ontologies developed for the application domain.

Implemented Ontology

In this sub-section is presented part of the ontology based in the work developed in[10], Core Ontology for Robotics and Automation (CORA), from a general knowledge model for robot development, including its components, controllers, and applications.
The CORA ontology was implemented in the free, open-source platform *protégé*, that provides tools to construct domain models and knowledge-based applications with ontologies. Gathering the previous presented ontologies was also performed in *protégé*, allowing to obtain seamlessly a owl file.
As seen in [10], the CORA ontology takes basic definitions from the IEEE Suggested Upper Merged Ontology (SUMO) ontology (`http://www.adampease.org/OP/`), i.e., Object, Device, and so on, as depicted in figure 10.
The parts of CORA that are of special interest while developing the control framework are related to the Robot Parts. In the RPARTS class, are defined all the possible robot components (classes), that will be instantiated when needed, depending on the type of robot that is to be controlled by the application. Since CORA is suited for all types of robots, e.g., mobile and/or manipulators, it can be observed some specific classes for manipulators, like *RPARTS:WristDevice*. For the robots used in this paper, are needed the *LocomotionDevice, RobotChassis, ElectronicDevice, Actuator, Sensor* classes.
The *ElectronicDevice* class have two sub-classes, the first related to the *CommunicationDevice* and the second related to the *Controller*. These two classes are instantiated, for example by the bluetooth device and the Arduino Board with its motor drivers, respectively. Moreover, these devices are used in the robots presented in the paper.
The *LocomotioDevice* class have different sub-classes applied at each type of robot. For example, the Quad-Copter needs to use the *PropellerDevice* class, the wheeled robot needs to use the *FixelWHeel* class, and

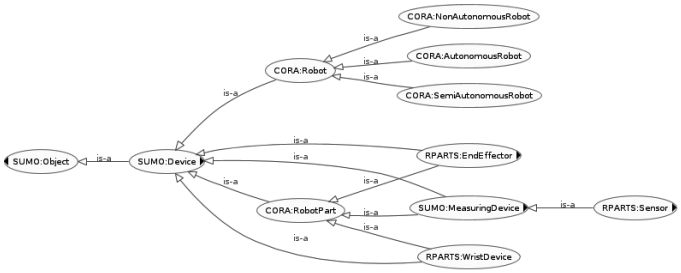


Figure 10: Snapshot of the CORA Robotic Ontology [10], the upper-level classes.

finally the tracked robot needs to use the *TrackDevice* class.
All of the above classes and sub-classes will be instantiated by different types of components, provided by different manufacturers.
CORA is very important for the overall proposed framework, since the settings of the Android application are defined in this ontology, leading to a XML file describing the robot to be controlled: its type, dimensions, controller, wireless interfaces, and application task. In figure 10 are depicted the components and its relations. It can be seen the knowledge related to the task, and to the robot components, i.e, all mechanical, electronic devices used for the robots used in this paper (Wheeled, Tracked and QuadCopter).

Results and Discussion

Teleoperation/VideoConferencing

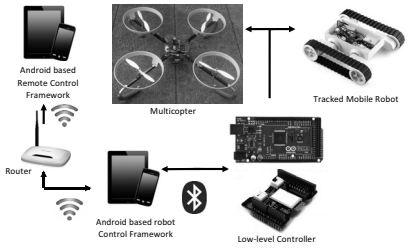


Figure 11: Teleoperation Robot Control.

In Fig. 2(b) is depicted the tracked robot in a surveillance task, while is moving around the lab avoiding obstacles. The trajectory generated for the surveillance task is random, inspired in the ROOMBA robot. The obstacle avoidance is performed using three ultrasound sensors located in the robot front, left and right sides. When an obstacle is found, as seen in figure 2(b), the surrounding sub-task was defined as: to keep a constant distance, 10[cm], from a box (measured with the ultrasound sensor that first observes it).
For the experiment presented, the Android application was used in the configuration depicted in figure 11, where the embedded control sends the supervision image to the teleoperation command, via a router, receiving by the same way the user commands.

Figure 12: Result image, with SIFT features, from the identified box.

Interaction with a Reasoning Engine

The first result obtained while fetching the knowledge model was when the Tracked Robot found a transportation box, seen in the figure 2(b), that he was trying to find while moving around the lab. The step-by-step procedure followed in this case was:

- in video conferencing mode the user asked the robot to find the box while in surveillance mode,
- the robot switched to autonomous mode and moves around the lab,
- the robot at each timestamp sends to the Reasoning Engine, Objects of Daily Use Finder – *ODUFinder* [12], an image captured with the embarked Android platform,
- if the box is found the knowledge model is updated with the information of the object localisation in the lab,
- the robot continues in surveillance mode.

The reasoning engine used is running under the ROS framework in the server, ODUFinder receives, the image sent and also the position and orientation of the robot, referred in the code listing **??**, after been passed to a

ROS : *Odometry* message. The later process in running in the Android application, at the smartphone embedded in the Tracked Robot. The *ODUFinder* framework applied computes for 2D images of the object, a set of SIFT descriptors that are used to train the the objects models, using the Inverse Frequency method [12]. This method was tested for 52 images and obtained a 96% success rate, and figure 12, presents the image of the identified object, the transportation box, with the clustered SIFT descriptors (each cluster in different colour). This interaction with the cloud is to obtain the descriptors to perform uncalibrated visual servoing, as presented in [13].

Conclusions and Future Work

In this paper were tackled several issues on the development of a control framework for low-cost intelligent robots, that can interact with the Cloud to enhance its capabilities. It was shown that top-end control algorithms for robots, e.g., that requires cognition, can be used by low-cost robotic platforms. These frameworks need to be setup in an on-line server and takes advantage of off-the-shelf hardware running Arduino/Android, for local/low-level control. Beside the benefits already presented in the these conclusions, the use ontologies will ease the robot-human interaction. In fact the knowledge related to the task, and to the robot components, i.e, all mechanical, electronic devices used for the robots used in this chapter (Wheeled, Tracked and QuadCopter), can be stored in a knowledge database, in a machine readable format, and easily accessed via the Cloud.

Future works will be devoted to include in the knowledge model, standards currently under development for robotics, namely for service robots.

Acknowledgement

This work was supported by FCT, through IDMEC, under LAETA, project UID/EMS/50022/2019. This work was partly supported by project 0043- EUROAGE-4-E (POCTEP Programa Interreg V-A Spain-Portugal).

REFERENCES

[1] Waibel, M., Beetz, M., Civera, J., D'Andrea, R., Elfring, J., Galvez-Lopez, D., Haussermann, K., Janssen, R., Montiel, J., Perzylo, A., Schiessle, B., Tenorth, M., Zweigle, O., van de Molengraft, R.: Roboearth. Robotics Automation Magazine, IEEE **18**(2) (2011) 69–82

[2] Tenorth, M., Beetz, M.: KnowRob – A Knowledge Processing Infrastructure for Cognition-enabled Robots. International Journal of Robotics Research (IJRR) **32**(5) (April 2013) 566 – 590

[3] Schlenoff, C., Prestes, E., Madhavan, R., Gonçalves, P., Li, H., Balakirsky, S., Kramer, T., Miguelanez, E.: An ieee standard ontology for robotics and automation. In: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. (2012) 1337–1342

[4] Guizzo, E.: Cloud Robotics: Connected to the Cloud, Robots Get Smarter. http://spectrum.ieee.org/automaton/robotics/robotics-software/cloud-robotics (2011) [Online; accessed 03-March-2014].

[5] Araujo, A., Portugal, D., Couceiro, M.S., Rocha, R.P.: Integrating arduino-based educational mobile robots in ros. Journal of Intelligent & Robotic Systems (2014) 1–18

[6] Torres, P., Gonçalves, P., Lopes, P.: Robiho–a robot companion for elderly people's homes. Applied Mechanics and Materials **282** (2013) 158–161

[7] Siegwart, R., Nourbakhsh, I.R., Scaramuzza, D.: Introduction to autonomous mobile robots. MIT press (2011)

[8] Balakirsky, S., Kootbally, Z., Kramer, T.R., Pietromartire, A., Schlenoff, C., Gupta, S.: Knowledge driven robotics for kitting applications. Robotics and Autonomous Systems **61**(11) (2013) 1205–1214

[9] Shiomi, M., Kamei, K., Kondo, T., Miyashita, T., Hagita, N.: Robotic service coordination for elderly people and caregivers with ubiquitous network robot platform. In: ARSO. (2013) 57–62

[10] Prestes, E., Carbonera, J.L., Fiorini, S.R., Jorge, V.A., Abel, M., Madhavan, R., Locoro, A., Gonçalves, P., Barreto, M.E., Habib, M., Chibani, A., Gérard, S., Amirat, Y., Schlenoff, C.: Towards a core ontology for robotics and automation. Robotics and Autonomous Systems **61**(11) (2013) 1193–1204

[11] Haidegger, T., Barreto, M., Gonçalves, P., Habib, M.K., Ragavan, V., Li, H., Vaccarella, A., Perrone, R., Prestes, E.: Applied ontologies and standards for service robots. Robotics and Autonomous Systems **61**(11) (2013) 1215–1223

[12] Dejan Pangercic, V.H., Beetz, M.: Fast and robust object detection in household environments using vocabulary trees with sift descriptors. In: In IROS, Workshop on Active Semantic Perception and Object Search in the Real World. (2011)

[13] Gonçalves, P.J.S., Paris, A., Christo, C., Sousa, J.M.C., Pinto, J.R.C.: Uncalibrated visual servoing in 3d workspace. In Campilho, A., Kamel, M., eds.: Image Analysis and Recognition, Berlin, Heidelberg, Springer Berlin Heidelberg (2006) 225–236

INFLUENCE OF TASK ALLOCATION PATTERNS ON SAFETY AND PRODUCTIVITY IN HUMAN-ROBOT-COLLABORATION

Titanilla Komenda, Fabian Ranz and Wilfried Sihn
Fraunhofer Austria Research GmbH
Production and Logistics Management
Theresianumgasse 7, 1040 Vienna, Austria
E-mail: titanilla.komenda@fraunhofer.at

KEYWORDS
Human-robot collaboration, task allocation, safety.

ABSTRACT

Collaborative applications, where humans and robots work in a pre-defined workspace at the same time, are characterized by flexible task allocation patterns. Here, task allocation is dependent on the required degree of automation as well as on the demands by the individual processes and provided abilities by the collaborative resources. The dynamics of this allocation process influences boundary conditions for the risk assessment, which is mandatory for the industrial implementation. This paper addresses the need to understand the correlation between task allocation and exposures on the human operator. By analysing different task allocation patterns on an industrial application, the effects on operator exposure and productivity are determined.

INTRODUCTION

Human-robot collaboration enables a high degree of flexibility and adaptability when it comes to varying batch sizes and product variants and thus varying degrees of automation. By applying dynamic task allocation patterns on the available resources in a collaborative system, the degree of automation can be changed dependent on the operational objectives. However, different task allocation patterns not only influence the degree of automation of the overall system but also the number and duration of exposures to the operator. Dependent on the collaboration mode, this can influence productivity in return, as the collaboration mode also defines the machine's motion behavior, i.e. velocity, when it comes to exposures. By defining a different task allocation pattern, not only the number of tasks for each resource changes but also the sequence of tasks for each resource and thus the overall path of individual resources. By changing the path, the number and duration of exposures on the operator changes affecting cycle time at the same time, which is not always obvious in most cases.

PROBLEM FORMULATION

Task allocation patterns are mostly conducted by defining and quantifying requirements of individual tasks (Linsinger et al., 2018), on the one hand, and available skills by individual resources (Ranz et al., 2017), on the other hand. The objective of the task allocation is to optimally match skills and requirements without under- or overstraining resulting not only in high-quality but also ergonomic processes (Faber et al., 2017; Castro et al., 2019). The objective functions in those cases, try to either minimize the number of resources and thus stations (Koltai and Tatay, 2011) or the cycle time (Glogowski et al. 2017; Bänzinger et al., 2018), by just taking into account individual processing times and skill-requirement-matches. Following on from this, Gombolay et. al. (2015, 2018) also take temporal and spatial constraints into consideration, relating to start and finish times of subtasks as well as physical constraints restricting resource proximity. However, the objective function misses the influence on specified collaboration modes (Hoffmann and Breazeak, 2009) and thus the influence on the overall cycle time by applying different allocation patterns (Pellegrinelli et al. 2017). The importance of considering the collaboration mode in task allocation results from the changing motion behavior (Komenda and Sihn, 2018) of individual resources depending on the proximity of other resources or collaborating partners. Thus, the overall cycle time not only depends on the sequence and allocation of tasks but in particular on the spatial configuration of resources at the time of allocating tasks (Wang et al., 2018).

The technical specification ISO/TS 15066 defines four collaboration modes, which conclude in following exposures and thus motion behavior of resources (Fig. 1):

i. *Safety-rated monitored stop:* No contact between operator and machine as machine stops its movement when operator enters the collaborative workspace. The machine's velocity drops from v_{max} to *0* when the operator reaches a distance defined by ISO 13855, $d_{rel} < d_{col}$.
ii. *Hand guiding:* Controlled contact between operator and machine. The machine's velocity is set by the operator's execution velocity $v_{machine} = v_{human}$
iii. *Speed and position monitoring:* No contact, but regulated distance or velocity to the operator. This collaboration mode can be additionally separated into *velocity monitoring by constant distance values* and *distance monitoring by constant velocity values*. The former sets the relative distance between the machine and the operator constant by changing the machine's velocity $v_{machine}(d_{rel})$, while the latter changes the relative distance between the machine and the operator dependent on the current machine velocity $d_{rel}(v_{machine})$.
iv. *Power and force limiting by design and control:* Allowed contact. The machine's velocity is defined by the operator's velocity vector and distance to the machine $v_{machine}(d_{rel, } v_{human})$

Based on this, a task allocation pattern is proposed which not only considers temporal and spatial constraints in terms of performing tasks in a given time frame and proximity of another resource but also the collaboration mode specifying

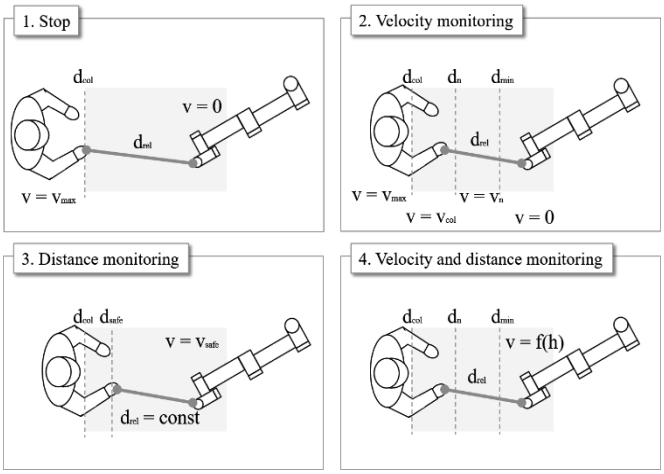


Figure 1: Strategies for Velocity and Distance Monitoring

the resource motion behavior and thus influencing the execution time of an allocated task. This approach can be considered a dynamic problem (Allaoui and Artiba, 2004) as the execution time of tasks allocated to a specific resource may change with every time step dependent on the current location of other resources.

TASK ALLOCATION APPROACH

This section presents the task allocation approach for multiple resources considering not only process requirements and resource skills but also temporal and spatial constraints as well as collaboration modes:
The task allocation approach has following inputs:

- $\boldsymbol{\tau}$…a $1 \times m$-matrix specifiying the set of tasks to be performed. A task is given by $\tau_i \in \boldsymbol{\tau},\ i \in \{1, \dots, m\}$
- $\boldsymbol{P}$…a $m \times m$-matrix specifying the precedence, i.e. temporal, constraints $p_{ik} \in \boldsymbol{P},\ i,k \in \{1, \dots, m\}$ of each task $\tau_i \in \boldsymbol{\tau}$, which may not be executed until all of its predecessor tasks are finished. $p_{ik} \in \{0,1\}$ is a binary decision variable specifying whether τ_i comes after or before τ_k.
- $\boldsymbol{R}$…a $1 \times n$-matrix specifiying the set of all resources, i.e. operators and machines, performing tasks. A resource is given by $r_j \in \boldsymbol{R},\ j \in \{1, \dots, n\}$
- $\boldsymbol{C}$…a $m \times n$-matrix specifiying the set of resource capabilities, i.e. the tasks each resource may perform. $c_{\tau_i r_j} \in \{0,1\}$ is a binary decision variable for assigning a task τ_i to a resource r_j.
- $\boldsymbol{M}$…a $1 \times n$-matrix specifiying the resource characteristics, i.e. the motion behavior resulting in a defined processing velocity of each resource $r_j \in \boldsymbol{R}$ dependent on the assigned collaboration mode as well as individual motion conditions. Thus, the completion time $t_{\tau_i}^{r_j}$ for each task $\tau_i \in \boldsymbol{\tau}$ is dependent on the allocated resource $r_j \in \boldsymbol{R}$ and its individual motion characteristics.
- $\boldsymbol{t_C}$…objective function to minimize the overall cycle time.

A solution to the problem consists of assigning all of the tasks to the available resources such as each task is assigned to at least one resource that is capable of performing the task, i.e. $\forall \tau_i \exists! r_j \in \boldsymbol{R}$, and at the same time satisfying spatial and temporal constraints while minimizing the overall cycle time. This problem can be formulated as a mixed integer linear programming (MILP)

$$\min t_C = \sum_{j=1}^{n} \sum_{i=1}^{m} t_{\tau_i}^{r_j} \tag{1}$$

subject to

$$\sum_{j=1}^{n} c_{\tau_i r_j} = 1, \forall \tau_i \in \boldsymbol{\tau}, r_j \in \boldsymbol{R}, i = 1, \dots, m \tag{2}$$

$$t_{\tau_i}^{r_j} = f(m_j), \forall \tau_i \in \boldsymbol{\tau}, r_j \in \boldsymbol{R} \tag{3}$$

In this formulation, equation 2 ensures, that each task is assigned to one resource. Equation 3 defines the resource specific processing time for an individual task based on its individual motion characteristics and the active collaboration mode.

EXPERIMENTAL DESIGN

The experimental setup was derived from a real industrial use case packaging printed circuit board (PCB) including six tasks (see Fig. 2), i.e. (1) scanning, (2) packaging, (3) tray handling, (4) cover handling, (5) container handling and (6) labelling. The workflow was specified as follows: PCBs were placed in batches of 12 in the packaging work station after cutting. Each loose PCB was scanned in order to check the successful execution of preprocesses. According to the test result, a PCB was packaged either in a OK-tray or NOK-tray. Trays had a batch size of 54. After filling an entire tray, it was placed in a container. A full container contained four full trays and one empty on top as dust protection. Once the empty tray was placed in the container, the lid was placed on it, the container was discharged and provided with an adhesive label.
For the packaging application, there were two resources, an operator and a robot, available. Thus, the application could have been executed either by the operator alone, by the robot alone, or by both parties in collaboration. The application with six tasks and two resources leads to

$$a_d = r^{\tau} = 2^6 = 64 \tag{4}$$

possible task distributions, which then have to be ordered in a sequence.

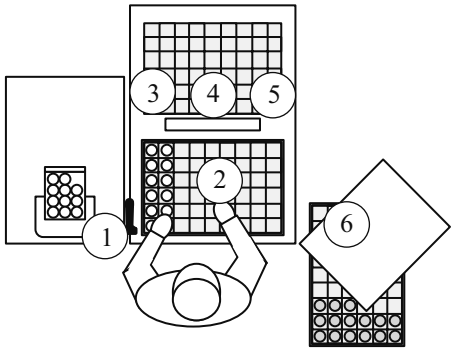


Figure 2: Packaging Workplace

If, for example, two tasks are distributed to the operator ($a_{\tau,o} = 2$) and four tasks to the robot ($a_{\tau,r} = 4$),

$$a_{seq} = a_{\tau,o}! \cdot a_{\tau,r}! = 2! \cdot 4! = 48 \quad (5)$$

different sequences can be formulated. For 64 possible task distributions, there are theoretically 5040 possible sequences in total.

For the experimental design, five task allocation patterns were defined: (i) operator does everything, (ii) operator does tray, cover and container handling as well as labelling, whereas scanning and packaging is done by both in collaboration, (iii) operator does tray, cover and container handling as well as labelling, whereas robot does scanning and packaging, (iv) operator only does labelling, (v) robot does everything.

Table 1 shows the mean processing times (according to MTM-1 for the operator and MTM-MRK for the robot) for each task conducted by the operator and the robot respectively (Schröter et al., 2016). As the operator is almost three times faster when it comes to handling single PCBs, the robot needs to handle three at once to result in the same cycle time. Thus, scanning and packaging tasks are conducted 216 times for the operator and 72 times for the robot when filling a whole container. As it can be seen, the operator is also faster in handling trays, covers and containers, but this can be balanced again by manipulating three PCBs at once.

Table 1: Processing Times for both Resources

No.	Task	Operator	Robot
1	Scanning	2,84 s	2,80 s
2	Packaging	1,44 s	7,20 s
3	Tray handling	1,80 s	35,00 s
4	Cover handling	1,80 s	10,00 s
5	Container handling	18,00 s	20,00 s
6	Labelling	6,84 s	3,00 s
Process time for one container		16,00 min	15,47 min

However, the given processing times for each task of the robot are results of a specific robot velocity defined by the collaboration mode of the robot in accordance to ISO/TS 15066. Thus, the motion behavior, i.e. velocity of the robot, can be considered as a dynamic quantity which is dependent on the proximity of other resources, such as operators, at the time of allocating a task. And the spatial arrangement of resources at that specific time is dependent on the previous task allocation pattern and the motion behavior of the resources at that time. In this sense, task allocation cannot be handled static anymore.

As the collaboration mode changes the robot's velocity dependent on the spatial distribution of resources in time, the optimizer was coupled to a simulator in order to receive the current resource locations as new inputs. Based on this, the new processing time for allocating a specific task to the robot was calculated and used as an input for the task allocator, where the updated processing times were used for further minimization of the overall objective function (see Fig. 3). The simulation model was implemented as a hybrid model (Barton and Lee, 2002) based on the simulation environment of Deatcu and Pawletta (2009) using the system specification formalism of discrete event and differential equation systems

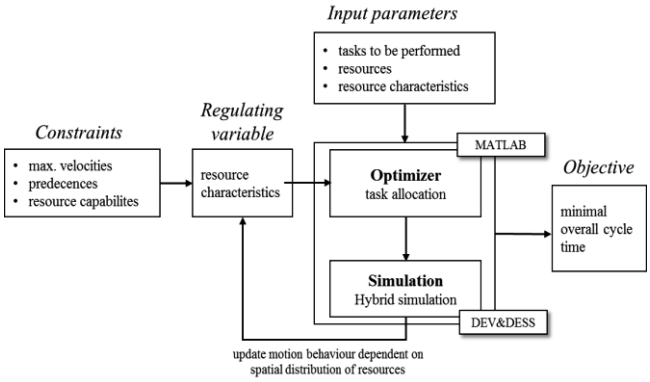


Figure 3: Principle of the Optimization Workflow

introduced by Zeigler (2000). Here, hybrid systems are defined by an 11-tuple including a set of discrete and continuous inputs and outputs, a set of possible states, external and internal state transition functions, discrete and continuous output functions, a rate of change function as well as a state event condition function.

In this experiment, three different collaboration modes were defined. The first mode stopped the robot as soon as the operator entered the collaboration space. The second mode reduced the robot's velocity as a function of the distance to the operator and the third collaboration mode specified the robot's velocity as a constant value according to ISO/TS 15066 (see pseudocode example in Fig. 4).

MOTION_BEHAVIOUR(τ, r, x, y, m)
1: **if** there is any resource in the collaboration space
2: calculate distance to that resource
3: check active collaboration mode
4: set resource velocity dependent on relative distance
5: calculate new processing time for the allocated task
6: **return** resource dependent processing time
7: **endif**

Figure 4: Pseudocode for Defining the Resource Specific Motion Behaviour based on the Active Collaboration Mode

RESULTS

The results of the five different task allocation patterns are shown in Fig. 5. The different patterns are given in the rows, whereas the coloumns show the spatial distribution of resources at different times. The overall cycle time of each task allocation pattern as well as the number of operator exposures are given in Table 2. The overall cycle time is given in minutes for loading one whole container, i.e. packaging 216 PCBs.

Table 2: Cycle Time Analysis of Different Task Allocations

No.	Scenario	Exposures	Cycle Time
i	Operator	0	16,00 min
ii.a	Collaboration (v250)	32	13,50 min
ii.b	Collaboration (vmax)	15	24,35 min
iii	Robot Packaging	6	21,57 min
iv	Operator Labelling	1	15,80 min
v	Robot	0	15,47 min

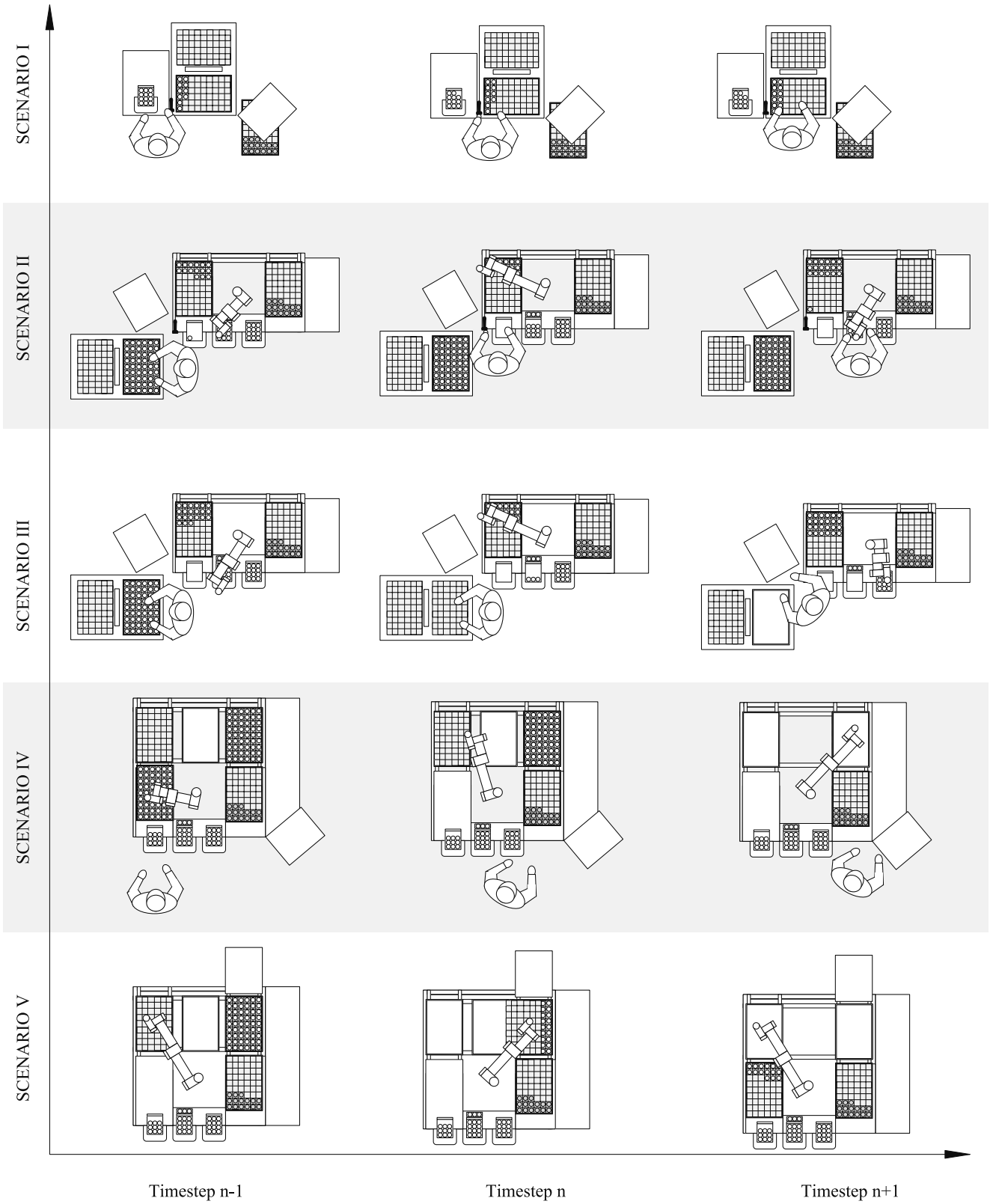


Figure 5: Extract from Different Task Allocation Patterns at Specific Time Steps

DISCUSSION

The results show, that the overall cycle time varies with a specific task allocation pattern. This is not only because of the precedence constraints of the overall process and the resulting waiting times if predecessor tasks were not finished at the time of allocating the new task, but rather due to spatial arrangements of resources at the time of allocation. Spatial proximity of resources may influence the motion behavior of a resource due to its pre-defined collaboration mode, resulting in different processing times at the time of allocation. Furthermore, task allocation influences the number of exposures to the operator as the sequence of tasks conducted by a specific resource can overlap with tasks of a different

resource within the same workspace. By influencing the number of exposures, safety is influenced as well. Thus, task allocation becomes a complex dynamic problem which needs to be solved by an optimizer in combination with simulation.

CONCLUSION

This paper showed the effects of different task allocation patterns on an industrial application in terms of operator exposure and productivity. Furthermore, it presented a method, how the dynamic problem of task allocation considering spatial influences on the individual motion behavior of resources and thus dynamically changing processing times can be solved. The implementation of a hybrid simulation model in combination with an optimizer showed good results in solving a problem with two resources and six tasks. In the future, a sensitivity analysis has to be conducted as well as an performance analysis using a higher number of resources, tasks and different optimizers.

ACKNOWLEDGEMENT

This work was supported by internal fundings of Fraunhofer Austria Research GmbH. The experimental setup was conducted in the TU Wien Industry 4.0 Pilot factory.

REFERENCES

Allaoui, H. and Artiba, A. 2004. "Integrating simulation and optimization to schedule a hybrid flow shop with maintenance constraints". In *Computers & Industrial Engineering*, No. 47, 431-450.

Bänziger T.; Kunz A. and Wegener K. 2018. "Optimizing human-robot task allocation using a simulation tool based on standardized work descriptions". In *Journal of Intelligent Manufacturing*.

Barton, P.I. and Lee, C.K. 2002. "Modeling, simulation, sensitivity analysis and optimization of hybrid systems". In *ACM Transactions on Modeling and Computer Simulation*, No. 2, 256-289.

Castro, R. P.; Högberg, D.; Ramsen, H.; Bjursten, J. and Hanson, L. 2019. "Virtual Simulation of Human-Robot Collaboration Workstations". Springer Nature Switzerland, 250-261.

Deatcu, C. and Pawletta, T. 2009. "Towards Dynamic Structure Hybrid DEVS for Scientific and Technical Computing Environments". In *SNE Simulation Europe*, No. 19, 75-78.

Faber, M.; Mertens, A. and Schlick C.M. 2017. "Cognition-enhanced assembly sequence planning for ergonomic and productive human-robot collaboration in self-optimizing assembly cells". German Academic Society for Production Engineering, No. 11(2), 145-154.

Glogowski, P.; Lemmerz, K.; Schulte, L.; Barthelmey, A.; Hypki, A.; Kuhlenkötter, B. and Deuse, J. 2017. "Task-based Simulation Tool for Human-Robot Collaboration within Assembly Systems". In *Tagungsband des 2. Kongresses Montage*, Springer-Verlag, 1-9.

Gombolay, M.C.; Huang, C. and Shah, J.A. 2015. "Coordination of Human-Robot Teaming with Human Task Preferences". Association for the Advancement of Artificial Intelligence.

Gombolay, M.C.; Wilcox, R.J. and Shah, J.A. 2018. "Fast Scheduling of robot Teams Performing Tasks with Temporospatial Constraints", 1-20.

Hoffmann, G. and Breazeak, C. 2009. "Effects of anticipatory perceptual simulation on practiced human-robot tasks". In *Autonomous Robots*, No. 28, 403-423.

Koltai, T. and Tatay, V. 2011. "Formulation of simple workforce skill constraints in assembly line balancing models". In *Social and Management Sciences*, No. 19, 43-50.

Komenda, T. and Sihn, W. 2018. "Cycle Time Optimisation in Self-Organising Production Lines with Human Machine Collaboration". In *Proceedings of the Vienna Young Scientists Symposium*, 42-43.

Linsinger, M.; Sudhoff, M.; Lemmerz, K.; Glogowksi, P. and Kuhlenkötter, B. 2018. "Task-based Potential Analysis for Human-Robot Collaboration within Assembly Systems". In *Tagungsband des 3. Kongresses Montage*, Springer Verlag, 1-12.

Pellegrinelli, S.; Orlandind, A.; Pedrocchi, B.; Umbrico, A. and Tolio T. 2017. "Motion planning and scheduling for human and industrial-robot collaboration". In *CIRP Annals - Manufacturing Technology*, No. 66, 1-4.

Ranz, F.; Hummel, V. and Sihn, W. 2017. "Capability-based task allocation in human-robot collaboration". In *Procedia Manufacturing*, No. 9, 182-189.

Schröter, D.; Kuhlang, P.; Finsterbusch, T.; Kuhrke, B. and Verl, A. 2016. "Introducing Process Building Blocks for Designing Human Robot Interaction Work Systems and Calculating Accurate Cycle Times". In *Procedia CIRP*, No. 44, 216-221.

Wang; W.; Li, R; Diekel, Z.M. and Jia, Y. 2018. "Robot action planning by online optimization in human-robot collaborative tasks". In *International Journal of Intelligent robotics and Applications*, No. 2, 161-179.

Zeigler, B.P.; Praehofer, H. ang Kim, T.G. 2000. "*Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems".* Academic Press.

AUTHOR BIOGRAPHIES

TITANILLA KOMENDA was born in Vienna, Austria and went to the University of Applied Sciences Technikum Wien, where she studied Mechatronics/Robotics and obtained her Master's degree in 2011. She worked as a lecturer and research assistant for the above mentioned university for a couple of years before moving to ProAutomation in 2013 and to Centauro in 2014, where she has been working in the field of simulating automation systems. In 2018 she started to work for Fraunhofer Austria Research GmbH in the field of human-robot-collaboration.

FABIAN RANZ was born in Luenen, Germany and went to the University of Reutlingen, where he studied Operations Management and obtained his Master's degree in 2014. He worked as a research associate in the research group value adding and logistics systems for the above mentioned university for a couple of years before moving to Fraunhofer Austria Research GmbH, where his work focuses on skill-based task allocation in flow shop through human-robot-collaboration.

WILFRIED SIHN was born in Pforzheim, Germany and graduated at the University Stuttgart. 1982 he started to work at the Fraunhofer Institute for Manufacturing Engineering and Automation (IPA), where he became deputy director in 2002. Since 2004, he is professor at the Institute of Management Sciences at the Technische Universität Wien and since 2009 executive board. Since November 2008, Wifried Sihn holds the position of Managing Director at Fraunhofer Austria Research GmbH where he is in charge of production and logistics management.

Enhanced Interaction with Industrial Robots Through Extended Reality Relying on Simulation-Based Digital Twins

Moritz Alfrink M. Sc.
Department of Man-Machine Interaction,
RWTH University,
Aachen, NRW, Germany
email: alfrink@mmi.rwth-aachen.de

Univ.-Prof. Dr.-Ing. habil. Jürgen Roßmann
Department of Man-Machine Interaction,
RWTH University,
Aachen, NRW, Germany
email: rossmann@mmi.rwth-aachen.de

KEYWORDS
Augmented Reality, Simulation, Industrial Robots, Industry 4.0, Digital Twins, eRobotics, HoloLens, Interaction, Intuitive.

ABSTRACT

The main objective of this paper is to introduce a novel approach for intuitive and natural interaction with heterogeneous robot types. To enable the visualisation of the robots as well as contextually relevant information an extended render client architecture for the Microsoft HoloLens is developed. This architecture takes advantage of the Microsoft HoloLens' capabilities to display three-dimensional content in augmented reality in combination with its gesture recognition. To provide the necessary data to the HoloLens, a server plugin is created as part of a feature rich simulation software. Using the eRobotics principle of Digital Twins, the server is able to represent and to augment arbitrary robot types with information generated in the simulation. By coupling the capabilities of the HoloLens with the powerful VEROSIM simulation software, a previously unseen form of natural interaction and information representation for heterogeneous robot types in augmented reality is created.

INTRODUCTION

The increased usage of intelligent robots for man-machine interaction as well as growing amounts of data, generated by the robots in online operation, results in high requirements for intuitive and natural interaction methods with robots, which current methods do not satisfy. Therefore, the objective of this work is an improvement and extension of existing interaction methods by using the latest technology in the field of augmented reality.
In Industry 4.0 machines are more and more able to capture their entire work process. The gathered data is transferred to a virtual counterpart of the robot - the Digital Twin -. Beyond representing the robot's information, the Digital Twin can also validate and augment them based on a simulation (Grösser 2018). As (Hempe 2016) states: The Digital Twin is part of the greater system of eRobotics aiming for virtualising the entire development process staring at the basic design, over virtual test beds, trainings and testing to a native visualisation of data based on a semantic world model. Comparing the current and previous robot generation a large increase of generated data took place, which still requires accessibility for humans. Typically, service technicians have to obtain their information from robot vendor specific control panels or from decentralized computers, which is neither intuitive nor efficient and introduces a gap between the physical and the virtual world as shown in figure 1, left. Therefore, the primary focus of this work is to simplify and enhance the interaction with physical robots. To reach this goal the reality will be systematically augmented with intuitive information about the robot and facility. This will close the gap between physical and virtual robot as outlined in figure 1, right.

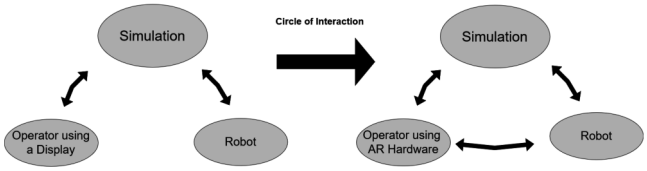


Figure 1: Closing the Gap using AR Hardware

Before reviewing the related work, this work should be positioned in (Milgram et al. 1995) reality-virtuality continuum, which describes the smooth transition from reality on the left to virtuality on the right. In order to augment the real environment with content and information, this work should be place as far as possible to the left side of the continuum. The presented solution is therefore using augment reality (AR), which is the closest technology to reality, in opposition to virtual reality (VR), the nearest to virtuality.
The current state of art suggests a few solutions trying to solve this problem partially. One approach from (Pagano 2017) uses smartphones to control robot swarms. The commands to the robots can be given on the smartphone's screen, which displays the augmented scene as seen through the smartphone's camera. The fo-

cus lies on a very user-friendly interface usable without knowledge of the underlying technology. To establish the ego-pose tracking required for augmented reality, the smartphone's camera is used to track AR marker in the scene. This forces the camera to detect the markers all the time, in order not to lose the tracking, which reduces the flexible usage. Furthermore, the interface by itself is less natural than a head mounted AR device, since the content is project onto two dimensional smartphone display, as shown by (Popelka and Brychtova 2013).
Two other projects from (Butler Robotics 2017) and (ABB Robotika 2016) using the HoloLens, try to close the gap between simulation and physical world shown in figure 1. The contribution from (ABB Robotika 2016) uses the HoloLens to display several of their robot models in AR and allows to move those around in space using the spatial gestures of the HoloLens. By visualising the robots (Butler Robotics 2017) closes the gap only in the direction pointing from the virtual to the real world, but lacks of complex feedback in the other direction, which is required to close the gap entirely. Furthermore, the virtual robot is not coupled with a simulation or a real robot, minimising the applicability in industry. Here, (Butler Robotics 2017) is a step ahead, in addition to rendering the robot in AR, the software also allows to record a trajectory using gestures. Afterwards the physical robot follows the trajectory after previewing the movement in AR. Even though the approach from (Butler Robotics 2017) is a step in the right direction, it still lacks from simulation-based information extension and validation. Moreover, the alignment between physical and virtual world has to be done manually in all reviewed projects, requiring an experienced user. Another issue none of the proposed projects handles, is a unified AR user interface, enabling the controlling of heterogeneous robot types.
Revising the requirements and state of art the following goals can be derived: Creation of a render client for the HoloLens, able to visualise arbitrary robots augmented with information provided from a service plug-in of a feature rich simulation. Implementation of a unified user interface providing control over any required functions and information. Moreover creating an automated alignment procedure to find the transformation between real and virtual world. Furthermore the implementation should be highly parallelised to maximise the rendering performance and thereby increase the AR experience.
The enabling core component is the mixed reality display HoloLens (Spillinger 2017), which allows visualising three dimensional content in augmented reality as well as hand tracking and gesture recognition. To make use of this capability a render client architecture, which is connected to a simulation, is conceptualised and implemented. The provided solution should display applications orientated, individualised information on arbitrary robots in combination with gesture-controlled interaction with the robot directly or a three dimensional user interface.

DESIGN

To meet the identified requirements the following server client architecture is proposed (see figure 2): The robot obtains its commands from the central simulation over a robot control unit and simultaneously passes its data via this unit back into the simulation. The simulation is also in permanent connection with the AR hardware, which indirectly closes the gap between robot and operator as shown in figure 1. This architecture has the advantage of separating different functionalities to their corresponding hardware. Furthermore, the expansion to other hardware requires only the reimplementation of the client side.

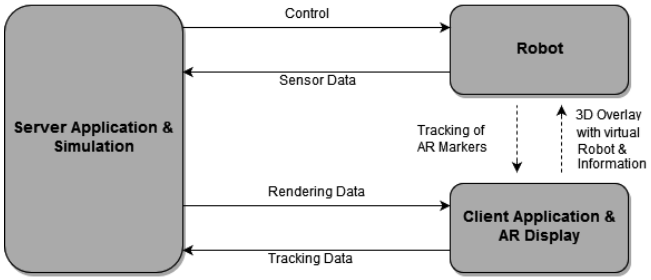


Figure 2: Basic Design

SERVER APPLICATION

On the server side, the primary objective is simulating the Digital Twins and holding the connections to the robots and the AR devices in synchronisation (see figure 3). The central components realising this concept are the synchronisation manager, the network manager, the interaction manager and the simulation. The synchronisation manager has the primary purpose to determine relevant changes for the client in simulation. These changes are then propagated to the client with the help of the network manager, who is in charge of submitting the synchronisation packages and to keep up the connection. Another important component is the interaction manager that incorporates the user interactions into the simulation. This allow e.g. to control the virtual or physical robot. The heart of the server application is the simulation, which can represent semantic world models. These models can contain arbitrary things as robots, facilities and environments. In the simulation dynamics, kinematics and generalised rigid body physics can be modelled, rendering it a very powerful database for the client.

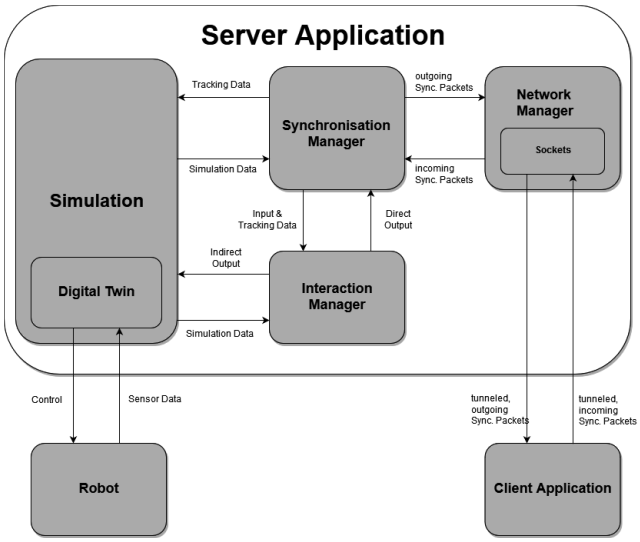


Figure 3: Server Design

CLIENT APPLICATION

The counterpart to the server is the client application managing two main tasks: First rendering the scene including geometry and all relevant information and second the tracking of the ego pose, the hands and AR markers. Figure 4 shows the resulting architecture. Similarity to the server the client also features a network manager and a synchronisation manager. The small difference in that the client's network manager only holds one connection, while the server can hold multiple. These two managers hide the physical separation of the client and server.

Moreover, the client features a tracking engine divided into its two core functionalities: Spatial- and Marker-tracking. The Spatial tracking is implemented in the HoloLens hardware (Microsoft 2018) and features the ego pose tracking, generation of an environment mesh and the tracking of the user's hands. The tracking of is ego pose is relative to the initial start point of tracking in a previously unknown environment (Environment CS), which afterwards remains static for further runs in this environment. Even though, the HoloLens can recognise environments by storing the generated spatial mesh, it cannot recognise an environment based on a partial, spatial mesh, which is a typical use case, when the user wants to render additional content to an environment, without having a representation of entire spatial mesh. This renders the HoloLens unable to determine the transformation between simulation and real model based on spatial tracking. Based on this constraint, the tracking data is extended by the marker-tracking module, which tracks ArUco markers in the HoloLens' coordinate system using the HoloLens' camera (OpenCV Community 2015). Figure 5 visualizes the perception of the two tracking engines. Accounting the

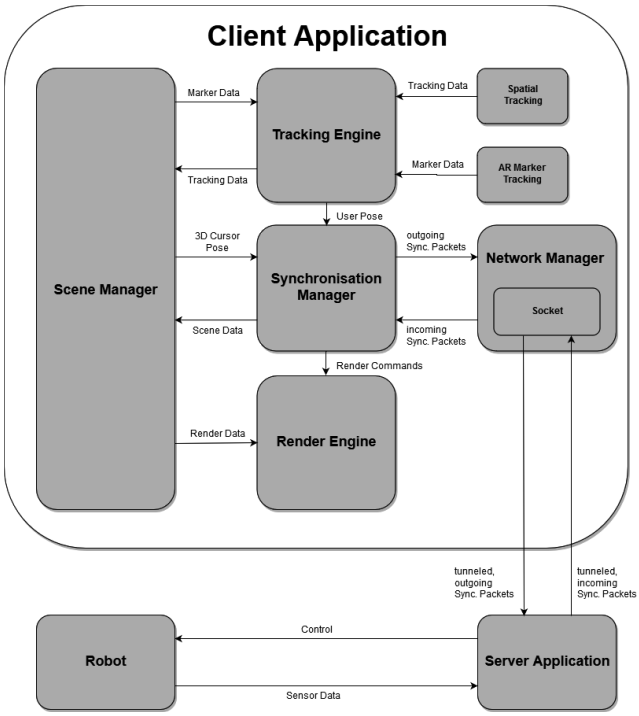


Figure 4: Client Design

marker tracking data, one can simply compute the relative transformation to the virtual marker in the simulation (see section implementation), which corresponds to transformation between the virtual and physical model.

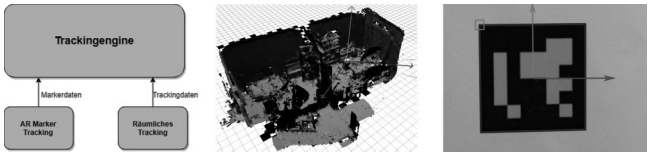


Figure 5: Perception of the Tracking Engine

Another essential component of the client is the scene manager, which stores the scene graph containing all relevant information about the scene. Beyond that, it keeps kinematic chains consistent. Finally the stereoscopic rendering of the scene graph is realised with the render engine.

These components are part of the detailed plan of the architecture, which is the key to a sustainable application. In the upcoming section will the implementation details of the tracking engine be unveiled.

IMPLEMENTATION

Within the scope of this paper, the implementation section will focus on the tracking engines. As shown in the design section, the tracking engine is divided into marker and spatial tracking. The spatial tracking estimates the ego pose of the HoloLens in an arbitrary initial coordinate system ${}^{Env}\underline{\mathbf{T}}_{HoloLens}$ and the marker

tracking estimates the pose of markers in the HoloLens' coordinate system ${}^{HoloLens}\underline{\mathbf{T}}_{Marker}$ as shown in figure 6.

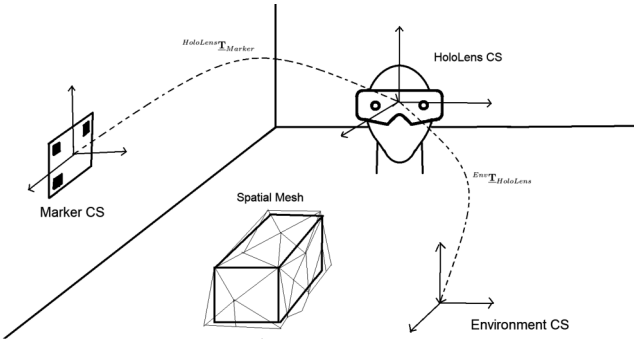


Figure 6: Tracking Engine

The pose of the marker is estimated in the camera's coordinate system as position $\mathbf{t}$ and Rodrigues rotation vector $\mathbf{r}$. In order to obtain a rotation matrix, the following formula (1) is applied.

$$\begin{aligned} &\theta = ||\underline{\mathbf{r}}|| \quad with \quad \underline{\mathbf{r}} \in R^3 \\ &\underline{\mathbf{r}} = \underline{\mathbf{r}}/\theta \\ &\underline{\mathbf{R}} = \cos\theta \cdot \underline{\mathbf{I}} + (1-\cos\theta)\cdot \underline{\mathbf{r}} \cdot \underline{\mathbf{r}}^T \\ &\quad + \sin\cdot\theta \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} \quad with \quad \underline{\mathbf{R}} \in R^{3\times 3} \end{aligned} \tag{1}$$

In the next step, the final transformation matrix (2) of the marker in the environment coordinate system can be computed.

$$\begin{aligned} {}^{Env}\underline{\mathbf{T}}_{Marker} &= {}^{Env}\,\underline{\mathbf{T}}_{HoloLens} \cdot {}^{HoloLens}\,\underline{\mathbf{T}}_{Marker} \\ &= {}^{Env}\,\underline{\mathbf{T}}_{HoloLens} \begin{bmatrix} \underline{\mathbf{R}}(\underline{\mathbf{r}}) & \underline{\mathbf{t}} \\ \underline{\mathbf{0}} & 1 \end{bmatrix} \end{aligned} \tag{2}$$

Given ${}^{Env}\underline{\mathbf{T}}_{Marker}$ from the real marker and the marker's pose in the simulation ${}^{Env}\underline{\mathbf{T}}_{MarkerSim}$, the transformation between both can be computed using equation (3).

$$\begin{aligned} &{}^{Env}\underline{\mathbf{T}}_{MarkerSim} \overset{!}{=} {}^{Sim}\,\underline{\mathbf{T}}_{Env} \cdot {}^{Env}\,\underline{\mathbf{T}}_{Marker} \\ \Longrightarrow \quad &{}^{Sim}\underline{\mathbf{T}}_{Env} = {}^{Env}\,\underline{\mathbf{T}}_{MarkerSim} \cdot {}^{Env}\,\underline{\mathbf{T}}^{-1}_{Marker} \end{aligned} \tag{3}$$

The inverse of this transformation ${}^{Sim}\underline{\mathbf{T}}^{-1}_{Env}$ can be used to transform all geometries from simulation to the HoloLens. In addition to calculating the transformation from one single pair of markers, a process very prone to errors during marker pose estimation, one can also calculate the transformation accounting for multiple markers, given the markers centres $\mathbf{x}_k$ $k \in sys_1, sys_2$ by first calculating their centres of gravity $\mathbf{r}_k$ using equation (4).

$$\bar{\underline{\mathbf{r}}}_k = \frac{1}{N} \cdot \sum_N^{i=1} \underline{\mathbf{x}}^i_k \quad k \in sys_1, sys_2 \tag{4}$$

By subtracting the centres of gravity from the positions, the point groups are only different in their rotation $\underline{\mathbf{R}}_{min}$ resulting in equation (5).

$$\begin{aligned} \underline{\mathbf{x}}^i_{sys_1} - \bar{\underline{\mathbf{r}}}_1 &= \underline{\mathbf{R}}_{min} \cdot (\underline{\mathbf{x}}^i_{sys_2} - \bar{\underline{\mathbf{r}}}_2) \\ \bar{\underline{\mathbf{x}}}^i_{sys_1} &= \underline{\mathbf{R}}_{min} \cdot \bar{\underline{\mathbf{x}}}^i_{sys_2} \end{aligned} \tag{5}$$

To determine $\underline{\mathbf{R}}_{min}$ the Kabsch Algorithm is used (Kabsch 1978). The algorithm is aimed at minimising the root-mean-square deviation defined in equation (6).

$$RMSD = \sqrt{\frac{E}{N}} = \sqrt{\frac{\sum_N^{i=1} ||\bar{\underline{\mathbf{x}}}^i_{sys_1} - \bar{\underline{\mathbf{x}}}^i_{sys_2} \cdot \underline{\mathbf{R}}_{min}||^2}{N}} \tag{6}$$

Equation (6) requires to minimize E (equation 7).

$$\begin{aligned} E &= \sum_N^{i=1} ||\bar{\underline{\mathbf{x}}}^i_{sys_1} - \bar{\underline{\mathbf{x}}}^i_{sys_2} \cdot \underline{\mathbf{R}}_{min}||^2 \\ &= \sum_N^{i=1} ||\bar{\underline{\mathbf{x}}}^i_{sys_1}||^2 + ||\bar{\underline{\mathbf{x}}}^i_{sys_2}||^2 \\ &\quad - 2 \cdot \sum_N^{i=1} \bar{\underline{\mathbf{x}}}^i_{sys_1} \cdot (\bar{\underline{\mathbf{x}}}^i_{sys_2} \cdot \underline{\mathbf{R}}_{min}) \\ &= E_0 - 2 \cdot L(\underline{\mathbf{R}}_{min}) \end{aligned} \tag{7}$$

In order to minimize E we have to maximize E_0 (equation 8).

$$\begin{aligned} L(\underline{\mathbf{R}}_{min}) &= \sum_N^{i=1} \bar{\underline{\mathbf{x}}}^i_{sys_1} \cdot \bar{\underline{\mathbf{x}}}^i_{sys_2} \cdot \underline{\mathbf{R}}_{min} \\ &= Tr(\underline{\mathbf{X}}^T \cdot \underline{\mathbf{Y}} \cdot \underline{\mathbf{R}}_{min}) \\ &= Tr(\underline{\mathbf{R}} \cdot \underline{\mathbf{\Omega}}) \end{aligned} \tag{8}$$

Finally we can decompose $\underline{\mathbf{\Omega}}$ in its singular values (equation 9), with $\underline{\mathbf{V}}$ and $\underline{\mathbf{W}}^T$ to be orthogonal matrices and σ_i the singular values.

$$\begin{aligned} L(\underline{\mathbf{R}}_{min}) &= Tr(\underline{\mathbf{R}} \cdot \underline{\mathbf{\Omega}}) \\ &= Tr(\underline{\mathbf{R}}_{min} \cdot \underline{\mathbf{V}} \cdot \underline{\mathbf{S}} \cdot \underline{\mathbf{W}}^T) \\ &= Tr(\underline{\mathbf{S}} \cdot \underline{\mathbf{T}}) \\ &= \sigma_1 T_{11} + \sigma_2 T_{22} + \sigma_3 T_{33} \end{aligned} \tag{9}$$

Since $\underline{\mathbf{T}}$ is a product of orthogonal matrices, it follows $T_{ii} \leq 1$, therefore $\underline{\mathbf{T}} = \underline{\mathbf{I}}$ maximises L, which results in equation (10).

$$\underline{\mathbf{I}} = \underline{\mathbf{W}}^T \cdot \underline{\mathbf{R}}_{min} \cdot \underline{\mathbf{V}} \tag{10}$$
$$\Rightarrow \underline{\mathbf{R}}_{min} = \underline{\mathbf{V}} \cdot \underline{\mathbf{W}}^T$$

To ensure a proper rotation matrix with $\det(\underline{\mathbf{R}}_{min}) = 1$, a correction matrix is inserted in equation (11).

$$\underline{\mathbf{R}}_{min} = \underline{\mathbf{V}} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & det(\underline{\mathbf{V}} \cdot \underline{\mathbf{W}}^T) \end{bmatrix} \cdot \underline{\mathbf{W}}^T \tag{11}$$

The final transformation is presented in equation (12).

$${}^{sys1}\underline{\mathbf{T}}_{sys2} = \begin{bmatrix} \underline{\mathbf{1}} & \bar{\underline{\mathbf{r}}}_2 \\ \underline{\mathbf{0}} & 1 \end{bmatrix} \cdot \begin{bmatrix} \underline{\mathbf{R}}_{min} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & 1 \end{bmatrix} \cdot \begin{bmatrix} \underline{\mathbf{1}} & -\bar{\underline{\mathbf{r}}}_1 \\ \underline{\mathbf{0}} & 1 \end{bmatrix} \tag{12}$$

Summing up, two ways to determine the transformation between the real and virtual coordinate systems are shown. The implementation can switch dynamically between both, depending on the amount of tracked markers. The marker tracking quality is determined in the evaluation section. In the next section, the features of the resulting application are shown.

APPLICATION

The main application consists of two parts: The server and the client. The server is implemented as a plug-in of the 3D cross-platform simulation software VEROSIM in C++. In VEROSIM, factories, processes and general robotics can be simulated. This is made possible by modelling the underlying physics, mechanics as well as a wide variety of typically utilised sensors. Furthermore, VEROSIM can natively control a variety of robots and mirror their data onto Digital Twins. In order to extend this software a HoloLens service proxy is created as a plug-in for VEROSIM. The plug-in hooks into the database of VEROSIM to provide all relevant information to connected HoloLenses. Furthermore, the plug-in outputs all information received from the HoloLens into VEROSIM dataflow modelling interface, which allows to connect the input and output data of any simulation instance with another.

TRAJECTORY ASSISTANT

A task, which is usually difficult to perform on a display due to the lack of depth perception due to their two-dimensional nature, is generating trajectories for a robot. To simplify this task, a trajectory assistant is created as part of the plug-in. The assistant allows to record a trajectory, simulating the movement on the Digital Twin and finally transferring it to the real robot. The big advantage is that the operator can directly create the trajectory in the physical scene while paying attention to the layout of the robot's working cell and other obstacles. Figure 7 shows the process of creating a tracking using the assistant.

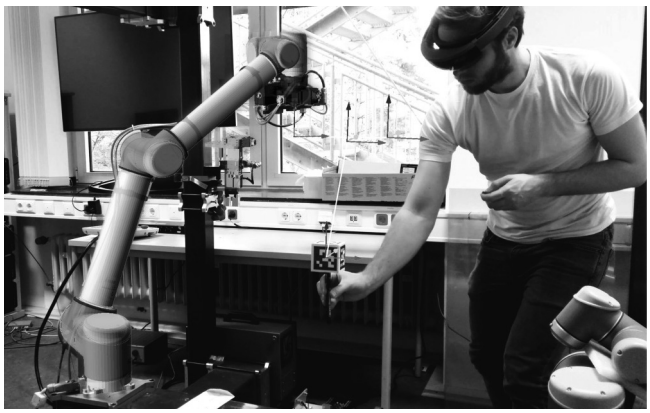


Figure 7: Trajectory Assistant

3D USER INTERFACE

In order to create a unified 3d-interface for every robot, the plug-in also provides building blocks for user interfaces. The blocks are an extension to geometry nodes in scene converting those into selectable nodes on the HoloLens. As figure 8 shows, the menu from simulation (shown on the left) is transferred to the real scene in augmented reality. The inputs of the operator to the interface in augmented reality are transferred into the graphical data interface in the simulation and can be used to control arbitrary actions.

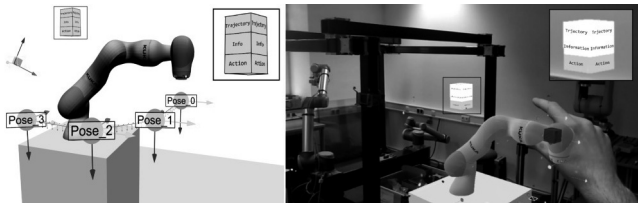


Figure 8: 3D User Interface

3D Cursor

Whereas the previous section described the server application's features, the focus lies in this section on the client application, starting with the 3d-cursor. The cursor looks like a little puck as seen in figure 9. The main purpose of the cursor is to give a visual feedback to the user about the surface he is currently gazing at. The information provided are the orientation of the surface and the distance to it. In addition to that, the cursor also features two indicators on the left and right, representing the user's hand being in the tracking area and being able to make user input with the 3d user interface.

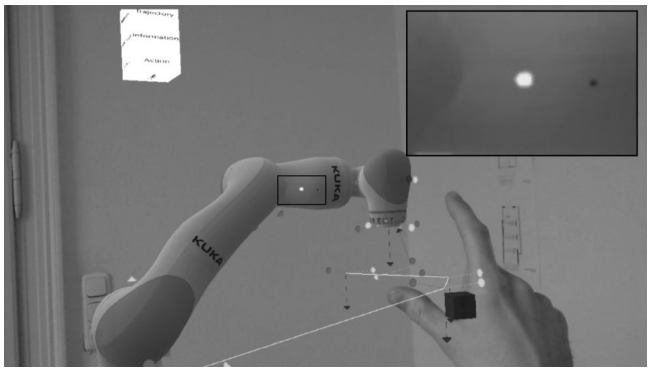


Figure 9: 3D Cursor

3D Marker Controller

Figure 9 shows a blue box within the right hand, which visualises its tracked position. This also illustrates the problem with the hand tracking on the HoloLens, which is sufficient for basic gestures, but not for exact 3-d pose tracking, that could, e.g., be used for trajectories. To overcome this limitation of the HoloLens, the marker tracking as described in client's design is used to track groups of markers, placed on one physical object. On the left, figure 10 shows a marker cube mounted on a stick and on the right the AR-cube overlaying it. The big advantage is the precise mapping between physical and virtual cube, which leads to good immersion and is necessary for for precise positioning. The menu items by themselves can be customised in the simulation software. Moreover, the input can be directed to any required functionality in the simulation, which makes the controller a universal interaction tool.

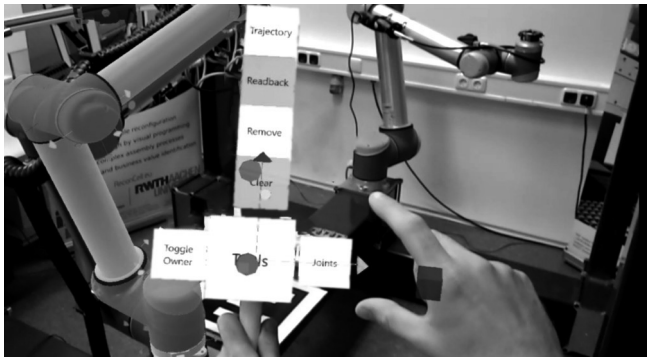


Figure 10: 3D Marker Controller

EVALUATION

In the scope of this work, three properties are very important to evaluate: The latency between server and client, the rendering performance of the HoloLens and the marker tracking accuracy. Therefore, this section will focus on measuring the given indicators.

Server-Client Latency

The server-client latency is the time a change in the simulation on the server takes effect on the HoloLens representing the client side. In order to measure this duration a ping and a pong network event for the synchronisation manager is created, which can hold a timestamp as payload. Hence, a measurement is done the following way: The sync-manager on an arbitrary side creates a ping event with the current time stamp, which is tunnelled though the network-manager on the both sides to the sync-manager on the receiving side. There a pong event is created with time stamp from the ping event and sent back. After travelling the same way back, the sync-manager on the initiating side receives the pong event and compares the time stamp in it with the current time. The latency is approximately half of this time difference. The measurement unveiled an average latency of 33ms, with a peak of 39ms on the target hardware using Wi-Fi. This mean latency is exactly twice the frame duration of the HoloLens, which implies that actions on the server take effect two to three frames after happening on the server. To identify the cause of this delay a second measurement has made on two systems connected via LAN using the HoloLens emulator. The delay measured in this configuration is 16ms, which is half of the previously measured delay. One can conclude that the Wi-Fi connection to the HoloLens is one main cause of delay.

RENDER PERFORMANCE

One very important criterion for AR-hardware is the rendering performance. The term rendering performance relates to the amount of triangles, which can be drawn, while still keeping the target hardware's display refresh rate. The problem with measuring this value is that scenes can have arbitrary content and are viewpoint-depending. This results in varying amount of overdraw and shading complexity. Furthermore, the aspect of effectiveness of draw calls and GPU resources provided by the software needs to considered. In order to evaluate both and leaving out the dependency on the software side rendering algorithms, two test scenes were rendered with active depth write but without depth testing and disabled v-sync. The test scenes can be seen in figure 11. The analysis showed a linear increase of the frame duration with respect to the triangle count of about 1ms per 6250 triangles. This gives an upper bound of about 90.000 triangles to maintain the 60 frames per second required for the HoloLens display. Furthermore, the independence of geometry instances transformation is shown by comparing the render duration of spheres with hundreds of triangles versus the boxes with twelve triangles. While the sphere instance count leaded to a fast increase in rendering time the box instance count did almost not increase the render time at all.

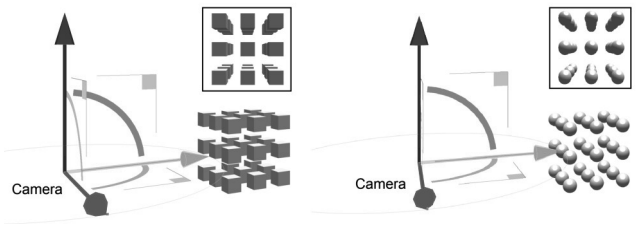


Figure 11: Test Scene

MARKER TRACKING ACCURACY

The alignment of holograms in augmented reality constitutes the last big influence on the immersion experience. The marker tracking has the strongest impact on this alignment as it determines the transformation between the simulation model and augmented reality model. Before any measurement took place, the HoloLens intrinsic camera parameters were determined. The quality of the calibration can be evaluated with the reprojection error, describing the geometric error between the measured and projected points in the image. The calibration resulted in a reprojection error of around two pixels. In order to measure the marker tracking quality, the HoloLens is mounted statically and a marker is placed in the camera's centre of view. With this configuration, three errors are measured: $\Delta x_{\parallel}$, the error in distance tracking from camera to marker, $\Delta x_{\perp}$, the error in distance tracking depending of the orthogonal offset from camera's centre of view $\Delta\alpha$ and $\Delta\beta$, the error of estimating the angle the marker is rotated around an axis orthogonal the camera's axis of view and $\Delta\gamma$, the error in estimating the marker's orientation around the axis parallel to the camera's axis of view. Figure 12 illustrates the setup, the physical setup consisted of a robot mounted to a linear axis.

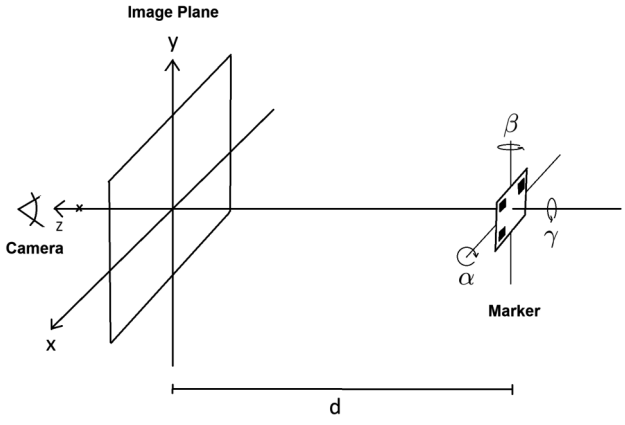


Figure 12: Marker Tracking Evaluation

The evaluation of the marker tracking accuracy of the calibrated camera shows that $\Delta x_{\parallel}$ grows quadratically and is at around $0,01$m in 1m of distance. $\Delta x_{\perp}$ lies within $0,005$m in a distance of 1m. $\Delta\alpha$ and $\Delta\beta$ are under 1° in the range from 0° to 40°. $\Delta\gamma$ has an upper bound of 4°. Measured on the HoloLens spatial tracking accuracy, the marker tracking is sufficiently precise for the required purpose.

CONCLUSION

The aim of this work is to improve the interaction with heterogeneous robot types, by visualising relevant information in a natural way using augmented reality on the HoloLens. The results stand out due to a flexible and reliable server-client architecture based on the feature rich simulation software VEROSIM. The created framework allows modelling of various robot types in arbitrary scenes and visualising them in augmented reality. Moreover, a complex interaction and control system is proposed, uniting the heterogeneous robot interfaces. The implemented marker tracking system enables the usage of even more complex interaction tools like the marker cube and simplifies the calibration of scene alignment. Finally, this system bridges the gap between simulation and physical world. For future applications, it would be interesting to implement the render client on different AR-hardware. Another aspect would be enhanced interaction between humans using this technology, e.g. could one technician mark a broken part for another technician to fix it later. In summary, the proposed system offers many new ways for man-machine interaction, it is designed to easily make use of the rapid technological development in the field and has therefore the potential to provide even larger benefits for its many possible users.

REFERENCES

ABB Robotika, 2016. *Augmented reality with ABB Robotics*. URL `https://www.youtube.com/watch?v=Q0vpxe8mLZY`.

Butler Robotics, 2017. *Hololens + ABB IRB120*. URL `https://www.youtube.com/watch?v=eqFShD6WQPs`.

Grösser P.D.S., 2018. *Digitaler Zwilling*. URL `http://wirtschaftslexikon.gabler.de/Archiv/-2045879713/digitaler-zwilling-v1.html`.

Hempe N., 2016. *Bridging the gap between rendering and simulation frameworks : concepts, approaches and applications for modern multi-domain VR simulation systems / Nico Hempe*. Ph.D. thesis, RWTH Aachen. Series: Research, Wiesbaden : Springer Vieweg 2016, ISBN 978-3-658-14400-5 Hauptberichter/Gutachter: Roßmann, Jürgen ; Kuhlen, Torsten Tag der mündlichen Prüfung/Habilitation: 2015-09-25.

Kabsch W., 1978. *A discussion of the solution for the best rotation to relate two sets of vectors. Acta Crystallographica Section A: Crystal Physics, Diffraction,*

Theoretical and General Crystallography, 34, no. 5, 827–828.

Microsoft C., 2018. *HoloLens hardware details*. URL https://developer.microsoft.com/en-us/windows/mixed-reality/hololens_hardware_details.

Milgram P.; Takemura H.; Utsumi A.; and Kishino F., 1995. *Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum*. In *Proceedings of the SPIE Conference on Telemanipulator and Telepresence Technologies*. Boston, Massachusetts, USA, *Proceedings of SPIE*, vol. 2351, 282–292.

OpenCV Community, 2015. *ArUco Marker Detection*. URL https://docs.opencv.org/3.1.0/d9/d6a/group__aruco.html#gafdd609e5c251dc7b8197323657a874c3.

Pagano A., 2017. *Controlling Robot Swarms With Augmented Reality*. URL https://spectrum.ieee.org/video/robotics/robotics-software/controlling-robot-swarms-with-augmented-reality.

Popelka S. and Brychtova A., 2013. *Eye-tracking Study on Different Perception of 2D and 3D Terrain Visualisation*. *The Cartographic Journal*, 50, no. 3, 240–246. doi:10.1179/1743277413Y.0000000058. URL https://doi.org/10.1179/1743277413Y.0000000058.

Spillinger D.I., 2017. *Microsoft HoloLens and Mixed Reality*. Microsoft. URL http://www.semiconkorea.org/en/sites/semiconkorea.org/files/data15/docs/4_Ilan%20Spillinger_Microsoft_distribution.pdf.

AUTHOR BIOGRAPHY

MORITZ ALFRINK received his double master degree in electrical engineering from the Rheinisch-Westfälische Technische Hochschule Aachen, Germany and the KTH Royal Institute of Technology, Sweden, in 2018. His current field of research at the Institute of Man-Machine-Interaction in Aachen, Germany is simulation technology and computer graphics. Besides his research he develops a computer graphics engine in C++ and creates little robots.

A CASE STUDY OF A DIGITAL TWIN FOR DESIGNING INTERMODAL RAILWAYS OPERATIONS FOR A MARITIME TERMINAL

Emanuele Morra

MEVB Consulting GmbH
28 Hubelistrasse, 4600
Olten, SWITZERLAND
E-mail: emanuele.morra@mevb-consulting.ch
e.morra@hotmail.it

Lorenzo Damiani
Roberto Revetria
Anastasiia Rozhok
Department of Mechanical Engineering
Genoa University
E-mail: Lorenzo.Damiani@unige.it
revetria@dime.unige.it
rozhok_anastasiya@mail.ru

1. KEYWORDS
digital twin, intermodal transportation system, monorail, timetables generator software, minerals transportation, road-rail-ship transport, simulation modelling.

2. ABSTRACT

The design of an intermodal transportation system, composed by *road-rail-ship* paths, usually passes through the capacity evaluation of the different stages, affected by means of transport features. Synchronicity of different stages and features of the utilities are a key factor in such infrastructure design. The simulation of a project, traditionally based on a static calculation of timetables for transportation connection guarantee, can be done actually with a Digital Twin approach for a whole dynamic simulation of the process. The paper deals with a real case study in which a full intermodal system for minerals transportation, located in a desert, has been fully simulated, with interesting results analysis the for project validation. The intermodal system in object includes a mine-to-terminal truck transportation, a terminal-to-seaport monorail train transportation and a highly automated system for bulk material conveying from seaport to ships.

3. INTRODUCTION

The project presented in this paper has been originated by the opportunity of exploitation of natural resources placed in the desert of the Middle East in an extreme region of the world. The connection of internal area with seaports on the coast, passing through the desert for more than 700 km is a challenging operation that involve geological, engineering and economics considerations.
The economical sustainability of the project is based on several key points:

1. Analysis of the capacity of the mines of bulk materials for the management of the extraction of natural resources on long term,
2. Use of an intermodal transport system with a high degree of automation for reducing the presence of people due to environment conditions,
3. Evaluation of the transportation capacity of the whole system,
4. Optimization of structures and utilities due to high maintenance costs.

The design phase for the construction of an intermodal transportation system is the critical task that realizes points 3 and 4 of the list above. It is based on four critical steps: I) Feasibility study; II)Structural, infrastructural and logistics executive design; III) Synchronization of the bulk material movement trough different transportation systems; IV) Validation of the project by dynamic simulation modelling.
The aim of this paper is the description of the solution adopted for the realization of point IV of the list above and results achieved. The modelling technique adopted for dynamic simulation (Abu-Taieh and El Sheikh, 2010, Romano et al., 2015, Law and Kelton, 1991) is the modern *Digital Twin* creation, a simulation model realized through a commercial simulation software that recreates the plant on several simulation layers as a highly detailed copy of the real one. The use of modern PC allows to perform fast simulations of a more than one year scenario in less than one hour, for real capacity evaluation and data extraction for analysis.
The use of software package AnyLogics has the following characteristics:
1. completeness of coverage of tasks in the field of the required research;
2. sufficiency of detail in modeling and simulating various scenarios;
3. the sufficiency of accuracy and reliability for the measurement and conformity assessment.
The use of this modern tool is naturally compared with software tools traditionally used for timetable generation for rail and ships, even if their scope is originally different and relative technical features are based on different assumptions. However this comparison is needed for the synchronization of the transportation means, which is a key point for the validation of the system capacity.
Designing intermodal railways operations for a maritime terminal is a complex multicomponent task for which the application of the modeling and simulation methods allows analyzing of arbitrary behavior of the entire system (Sharma, Sunil Kumar, 2015).
Conclusions and results are focused on the importance of dynamic simulation process which allows to show effects and behaviors of the whole system. The use of analytical methods in solving problems posed in this work is

impossible, since, unlike modeling and simulation, these methods are too simplified for real systems (Christy, D. and H. Watson, 1983). The real behavior of the system cannot result from simulation and modelling based on static software with a low compliance to the space modelling of the plant.

4. REAL CASE

The real case treated in the paper is a complex intermodal system with bulk materials flow that follows the following scheme:

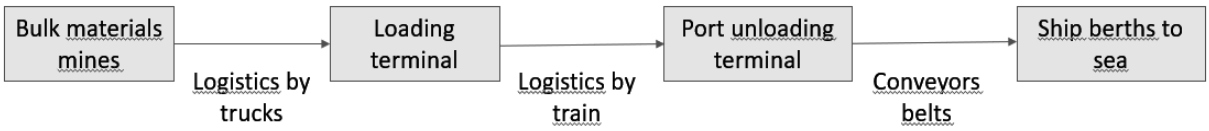


Figure 1: Logistic chain.

As visible in figure 1, logistics involves: transportation by trucks from mines to rail terminal stations; transportation by train on railway from loading terminal stations to port terminal; transportation by conveyors belts from port terminal unloading station to seaport berths for cargo-vessels loading; transportation by vessels from berths to delivery destination.
Train transportation is realized through a monorail system in which rail traffic between forward and backward trains to and from the port is regulated by the placement of intermediate passing loops at regulars distance to allow crossing of paths.

5. SIMULATION MODEL

The modelling technique used for Digital Twin realization is based on several model layers that tend to reproduce real operation. The simulation model used employs Agent Based technique and system dynamics, made possible by the AnyLogics software. The former is used to simulate most of the objects introduced inside the model, while the latter is used to simulate bulk material flow. External data input have been prepared in order to simulate real timetables generated by a static software and imported into the model (Escalona, José L., and Javier F., 2018).
Model characteristics are: I) Variable train composition; II) Switch management and routing; III) Train crossing management.
For this purpose the following aspects will be simulated: i) Network traffic from loading terminals to Sea Port and vice versa. Loaded trains coming from quarries to port shall have a higher priority; ii) Train fleet, to simulate as precisely as possible the deliverable load; iii) Full network topology, including all infrastructure – passing loops, bulk and mixed terminals (Bruzzone et al., 1999, Cassettari et al., 2011).

5.1 Logic Layer of the Model

The main complexity in the logic layer modelling concerns the trains traffic management in the single Passing Loops of the rail network to avoid trains collisions and exploiting as much as possible the low cost monorail with backward and forward trains.
Logic at the crossing considers the following features: i) Priority train is supposed not to wait and travel straight through at a crossing with another train; ii) In this case, both the trains coming from the port are priority trains, according to the XML schedule where we can see that they do not wait; iii) If it arrives earlier and has to wait, a delay ("stop-go for priority train") is considered. This is recorded in statistics.
The Figure 2 describes a passing loop scheme that takes into account the most important parameter - the logic of train behavior at the intersection.

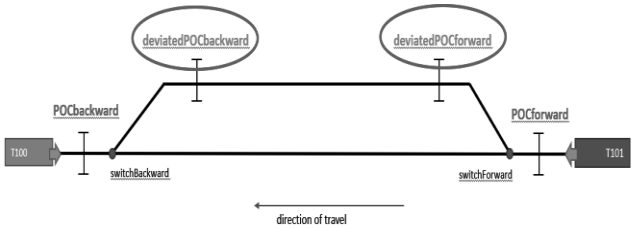


Figure 2: Passing Loop scheme.

The logic provides that train with lower priority waits until major priority train has passed the switch forward and then can be released according to scheduling.
When a priority train has crossed successfully the PL, a release message is sent to the waiting train in the PL-POC (point of control), but scheduling controls the real (Wang et al., 2017).
As visible in figure 2, passing loop itself has been modelled as agent in which agent train is sent by logic interfaces.
Regarding the 2D/3D layer of the model, the Network has been automatically generated from Java Code snippet. In the following figures 2D creation of the net is represented in scale 1:1 between real measures and model dimensions. Routing of trains is read from a .xls file "trains.xls" and Passing Loop data and timing crossing information have been added.

6. PROJECT VALIDATION

6.1 Static Software for Timetable Generation

The realization of the model provided to practical effects an intermodal system simulator for the validation of the timetables according to network topology data.
Procedure for validation is based on the following report:.

6.2 Scenario Description:

- Time Window: YEAR 12
- DownTime on Net: EXCLUDED
- Randomness on Loading Operation: Triangular (mean = 16844 t, min = - 3%, max = + 15 %)
- Inspection Time: 131'
- Maximum n. of contemporary loading operations at the terminal: 1
- Maximum n. of contemporary inspection operations at the terminal: 2

6.3 Analyzed Results:

- Simulation stops: at 25 of march 2018 (after 2 months of randomness exposure)
- Full net BulkTonnage transported by the net (terminals to Port):

Is to be noted that some trains at inspection are too late due to terminal congestion and cannot be injected on the net in time, 3 trains cancelled from planning, next timetable cannot be respected, simulation stops.

7. RESULTS

7.1 Results Validation

The process of validation has passed through the comparison of an input scheduling timetable represented by its Train Graph for crossing coincidence on weekly plan and the dynamic simulation results.
Validation of the model is complete if the model is able to end the simulation run without unfeasibility in the traffic management due to train collisions, passing loops that remain busy and block further crossing and trains that need to exceed net constraints like maximum speed or minimum servicing time in the maintenance stations.
Validation results have shown the following in the input timetables: i) Collisions due to too strict coincidences; ii) More than one train that has to occupy the same passing loop due to unfeasibility of timetables; iii) Overcoming of maximum speed of trains to respect timetables. Iterative adjustments on the timetables due to Digital Twin diagnostics has brought to complete feasibility and positive results in validation (Demartini et al, 2017, Revetria et al., 2016, Myers and Montgomery, 1995).

7.2 Logic Modelling Results

Logic modelling results is strictly connected to the capability of the implementation to stop simulation when unfeasibility arises, alerting with information that allows the user to understand input data incoherence. Error diagnostic allow to understand the reason of blocking situation.

8. CONCLUSIONS

At present *Digital Twin* represents one of the most powerful and complete modelling technique due to advance in PC performances and quality of simulation software. The opportunity of simulating plants with a space and time high reference, with a 2D/3D conception in model building also for showing results in addition to logic modelling, has revealed capabilities never exploited before which do not belong to static software modelling tools.
In conclusion, the use of a Digital Twin simulation model allowed the validation of the project after an iterative assessment of the real capacity of the system mainly due to adjustments and correction of the synchronization of train bidirectional routes on the monorail system with passing loops placed at a distance of 30 km.
During the iterative validation process, the Digital Twin model has revealed local unfeasibility of train graphs generated by static software that might compromise the real capacity of the rail system. These local issues were generated by the limits of static software that have been listed below:

- Impossibility to take into account real space size of trains. Bulk trains can be 2 km long while the parallel track of a passing loop is only 2.5 km long;
- Impossibility to take into account dynamic behavior of the train movement with acceleration and deceleration of the locomotive;
- Impossibility to simulate statistical effects of delay on long term simulation that can affect the traffic in specific events and test the robustness of timetables.

Other issues revealed by the Twin regarded Loading Terminal operations. In particular, a former under estimation of servicing on the trains.
Through the utilization of dynamic Digital Twin simulation issues have been detached and solved.
From the scientific point of view, results have shown the great capability of Digital Twin to reproduce reality better than other static software even on their specific fields. In particular the re-creation of a space and time model layer in scale 1:1 to reality and then the development of a higher level logic control customized on agents/objects management reveals itself as a winning choice: Digital Twin tends to be a multipurpose simulation model which shows better performance than specific software tools.
This is due to the complete amount of information that Digital Twin can receive and process instead of specific limited assumptions on which static software are based.
Great acknowledgement consists in the peculiar feature of Digital Twin for dynamics results showing through 2D and 3D interfaces which accelerate the model inspection and make it more effective for communication among users too.

9. REFERENCES

Abu-Taieh, E.M.O., El Sheikh, A.A.R., Discrete event simulation environments, (2010) Information Science Reference.

Bruzzone A. G., Giribone, P., Revetria, R., Operative requirements and advances for the new generation simulators in multimodal container terminals, (1999) Winter Simulation Conference Proceedings, 2, pp. 1243-1252.

Cassettari L., Mosca R., Revetria R., Rolando F., Sizing of a 3,000,000t bulk cargo port through discrete and stochastic simulation integrated with response surface methodology techniques (2011) Recent Advances in Signal Processing, Computational Geometry and Systems Theory - ISCGAV'11, ISTASC'11, pp. 211-216.

Christy, D. and H. Watson. 1983. The Application of Simulation. IEEE Transactions on Systems, Man and Cybernetics, Vol. 13 No 1, pp. 18-39.

Demartini M., Damiani L., Revetria R., Tonelli F., An innovative model for crashing DSS implementation process in industry

4.0: A case study, (2017) Lecture Notes in Engineering and Computer Science, 2, pp. 970-976.
Escalona, José L., and Javier F. Aceituno. "Multibody simulation of railway vehicles with contact lookup tables." International Journal of Mechanical Sciences (2018).
Law A.M., Kelton, W.D., (1991) Simulation Modeling and Analysis.
Myers R.H., Montgomery D.C. (1995) Response Surface Methodology.
Revetria R., Guizzi G., Giribone P., Management redesign of a multi-product line through discrete event simulation and response surface methodology, (2016) Proceedings of the IASTED International Conference on Modelling, Identification and Control, 830, pp. 1-7.
Romano E., Damiani L., Revetria R., Guizzi G., Different Approaches for Studying Interruptible Industrial Processes: Application of Two Different Simulation Techniques, (2015) Handbook of Research on Computational Simulation and Modeling in Engineering, DOI: 10.4018/978-1-4666-8823-0.ch002.
Sharma, Sunil Kumar, et al. "Challenges in rail vehicle-track modeling and simulation." International Journal of Vehicle Structures & Systems 7.1 (2015): 1.
Wang H., Zhang X., Damiani L., Giribone P., Revetria R., Ronchetti G., Video Analysis for Improving Transportation Safety: Obstacles and Collision Detection Applied to Railways and Roads, IAENG Conference IMECS 2017, Hong Kong, 15 – 17 March 2017.

10. AUTHOR BIOGRAPHIES.

LORENZO DAMIANI is a Researcher in the Mechanical Engineering department of Genoa University. He holds a Ph.D. in Engineering from Genoa University, His research interests include industrial plants, modeling and simulation. His email address is Lorenzo.Damiani@unige.it.

EMANUELE MORRA is a modelling and simulation consultant at Mevb Consulting GmbH as well as a contract Professor in both Mechanical Engineering Department of Genoa University and Production Management Department at the Politecnico of Turin. He holds a five years Degree in Electrical Engineering at the Politecnico of Turin. His research interests include modeling, simulation and industrial plants. His email address is emanuel.morra@mevb-consulting.ch.

ROBERTO REVETRIA is a Full Professor in the Mechanical Engineering Department of Genoa University, received his Ph.D. from Genoa University, His research interests include modeling, simulation and industrial plants. His email address is Revetria@dime.unige.it.

ANASTASIIA ROZHOK is a graduating Master Student in the Mechanical Engineering Department of Genoa University. She holds a four years Degree at the Faculty of Power Engineering in the field of Industrial Safety and Ecology at the Bauman Moscow State Technical University. Her research interests include modeling, simulation and industrial plants. Her email address is rozhok_anastasiya@mail.ru.

ENGINEERING SIMULATION

INFLUENCE OF PARAMETRIC UNCERTAINTY OF SELECTED CLASSIFICATION MODELS OF SEWAGE QUALITY INDICATORS ON THE SLUDGE VOLUME INDEX FORECAST

Bartosz Szeląg[1], Jan Studziński[2], Izabela Rojek[3]
[1] Kielce University of Technology, Faculty of Environmental, Geomatic and Energy Engineering
Al. Tysiąclecia Państwa Polskiego 7, 25 – 314 Kielce, Poland
E-mail: bszelag@tu.kielce.pl
[2] Polish Academy of Sciences, Systems Research Institute (IBS PAN)
Newelska 6, 01-447 Warsaw, Poland
E-mail: studzins@ibspan.waw.pl
[3] Kazimierz Wielki University, Bydgoszcz
M.K. Ogińskiego 16, 85-092 Bydgoszcz, Poland
E-mail: izarojek@ukw.edu.pl

KEYWORDS
Parametric and classification models, sludge volume index, biological sewage treatment plants.

ABSTRACT
One of the key parameters determining the operation of a biological reactor in a sewage treatment plant is the sludge volume index which determines the sludge sedimentation capacity. At the same time the sludge sedimentation is significantly influenced by the quality indicators of raw sewage, however, due to the time and costs of their determination, there may be problems with obtaining sufficiently long and undisturbed measurement sequences. This is a significant limitation of the ability to identify the sedimentation capacity of sludge and its improvement by changing the operational parameters of the biological reactor. However, the wastewater quality indicators included in mathematical models simulating sludge sedimentation can also be determined by mathematical modelling. The question arises as to how errors in the results of the forecast of individual quality indicators determined by means of modelling will influence the results of the forecast of sludge sedimentation capacity obtained also by means of a mathematical model. The paper presents the methodology of modelling the sedimentation capacity of activated sludge by means of classification models, in which the values of sewage quality indicators were also determined by means of mathematical models. A detailed analysis of the impact of the forecast error of individual sewage quality indicators determined by classification models on the accuracy of the simulation of the activated sludge volume index has been carried out.

INTRODUCTION
Classification and regression models in the form of black box models can be used to predict the processes occurring in sewage treatment plants (STP). In regression models the result is a specified numerical value, while in the case of classification models, belonging to the appropriate class is determined. In many cases, such an approach enables the identification of technological parameters of the treatment plant using a much smaller number of model input variables than in regression models. The predictive capacity of regression models is assessed on the basis of commonly used fitting measures such as correlation coefficient, absolute and relative error, etc. For classification models, the accuracy of the assignment shall be determined, i.e. the number of events that have been properly identified. In regression models, the standard deviations of the determined parameter estimators are also used as a measure of their fit to the measurements, but in the case of classification models these measures are not used, which may lead to some discrepancies in the assessment of the models obtained by different methods (Graniero 1996, Dybowski and Roberts 2006).
In wastewater treatment plants an important parameter determining the operation of a biological reactor is the sludge volume index (SVI). Regression and classification models used for reactor modelling do not define the analytical relationship between input and output variables of the model, as in the case of physical models, which hinders reliable analysis of calculation results and requires additional actions to support the assessment of the usefulness of the model determined.
In addition to problems related to the assessment of the correctness of the models determined, significant limitations may appear at the stage of their application in operating conditions. This is due to the fact that the sedimentation capacity of activated sludge is significantly influenced by the raw sewage quality indicators, and due to the time and costs of continuous measurement of these indicators, there are often problems with obtaining suitably long and continuous measurement sequences.
Therefore, a concept was developed to determine the values of wastewater quality indicators contained in the parametric mathematical model also by means of mathematical modelling. However, the question arises how errors in the forecasting of individual quality indicators will influence the results of calculations of the mathematical model for forecasting the sedimentation capacity of activated sludge. It is doubtful whether the sludge sedimentation forecasts obtained by means of a model in which the values of the quality indicators are not measured, but calculated, will be reliable and can be used by a treatment plant technologist to control the sewage treatment process.
In order to clarify these doubts, the paper considers the possibility of forecasting the sedimentation capacity of activated sludge using classification models, in which the included indicators of wastewater quality were calculated with regression models.
Due to the fact that the determined models are characterized by limited predictive capabilities, in order to assess the usefulness of the proposed methodology for determining models with utility properties, the impact of forecasting errors in particular indicators of sewage quality in regression models and errors in

estimating parameters in classification models on the accuracy of modelling the activated sludge volume index was analysed in detail.

OBJECT OF STUDY

The data for the calculations came from the monitoring carried out in the municipal sewage treatment plant with a capacity of 72,000 m^3/d, which receives sewage from the sanitary sewage system of the city of Kielce and two communes. First, the sewage is treated mechanically and then it reaches the primary clarifiers, from where it is directed to the biological part of the treatment plant. Organic compounds, nitrogen and phosphorus are removed from the sewage and after biological treatment the wastewater flows into secondary clarifiers, where it is separated from the activated sludge and reaches the Bobrza river.

The monitoring carried out since 2012 at the sewage treatment plant includes measurements of the quantity (Q) and quality indicators of the wastewater (biochemical (BOD_5) and chemical (COD) demand for oxygen, suspended solids (TSS), total nitrogen (TN), ammonium nitrogen ($N\text{-}NH_4$), nitrate nitrogen ($N\text{-}NO_3$), total phosphorus (TP)). In addition, chlorides (Cl) and biological reactor operating parameters (pH, temperature (T_{sl}), activated sludge concentration (MLSS), recirculated sludge concentration (REC), recirculation rate (RAS), chemical coagulant dosing (PIX) and oxygen concentration in nitrifying chambers (DO)) are measured.

METHODOLOGY

In the following a method to assess the impact of the uncertainty of the forecast of selected indicators of sewage quality and parametric uncertainty of classification models on the results of the simulation of activated sludge sedimentation (characterized by SVI value) with the use of selected classification models is shown. Individual calculation stages of the developed method in form of a calculation scheme are presented in Fig. 1.

There is to see from the calculation scheme that the first step of the method proposed is to develop classification models for the simulation of sludge sedimentation, i.e. to determine the variables (x_i) explaining the modelled value of the sludge volume index. In order to compare the predictive abilities of the classification models, the use of the logistic regression model (RL) and the Gompertz model (GM) is considered. These models are commonly used in medicine, microbiology and social sciences. Our goal was to compare the predictive capabilities of these models in the forecast of activated sludge sedimentation in sewage treatment plants and to indicate a model, which, if applied in everyday operational practice, would lead to making appropriate technological decisions and improving the efficiency of sewage treatment plant operations.

The classification models considered in the paper describe the following relations:

- logistic regression model:

$$p(X)=(\exp(X))/(1+\exp(X)) \quad (1)$$

- Gompertz model:

$$p(X)=\exp(-\exp(-X)) \quad (2)$$

where X is the linear vector being the combination of independent variables (x_i), i.e. of the quantity and quality of wastewater and operating bioreactor parameters, described as:

$$X = \alpha_0 + \sum_{i=1}^{m} \alpha_i \cdot x_i \quad (3)$$

Only explanatory variables (x_i), which are statistically significant at the assumed confidence level (α=0.95), were taken into account in the developed classification models.

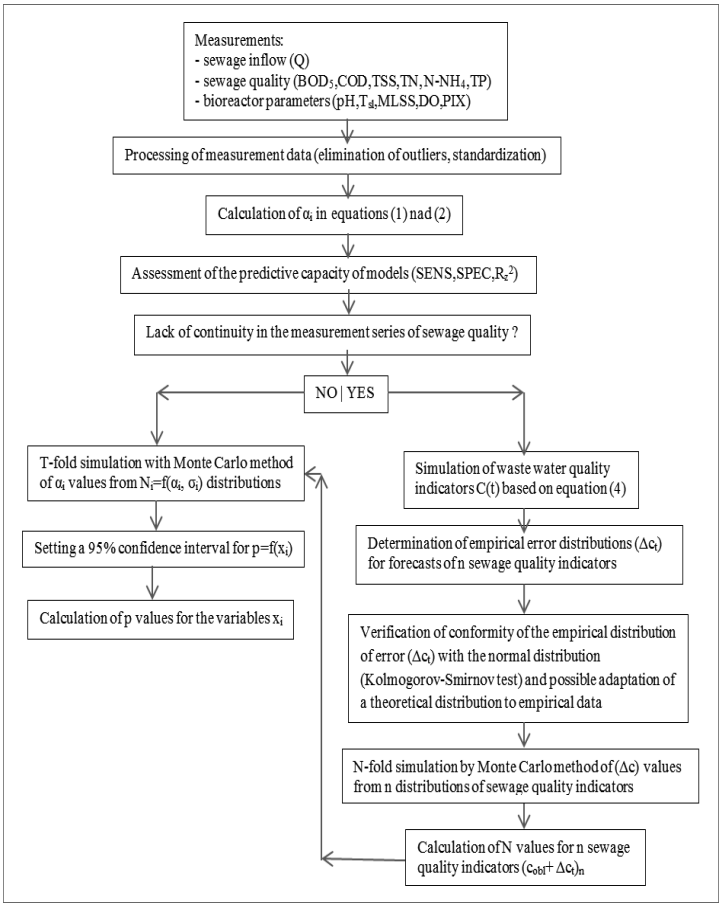


Figure 1. Calculation scheme of the influence of parametric and forecast uncertainty of sewage quality indicators on the simulation of SVI with classification models.

In further analyses, when assessing the influence of the considered sewage quality indicators and operational parameters of activated sludge chambers on the forecast of the sludge volume index, its reference value of SVI_{lim}=150 ml/g was referred to (Bayo et al. 2006, Martinez et al. 2006). Correlation coefficients R^2 McFadden, R^2 Cox-Snell and R^2 Nagelkerke as well as values of sensitivity (SENS), specificity (SPEC) and counting error (R_Z^2) (Harrell 2001) were used to assess the predictive capacity of classification models.

In the next stage of investigation, if the explanatory variables in the model (1) or (2) are the indicators of wastewater quality and there are problems with obtaining their continuous measurements, the black box modelling methods can be used to supplement the missing values. For this modelling the methods of Support Vector Machines (SVM) and k-Nearest Neighbours (k-NN) have been used, which for this purpose were already applied by Minsoo et al. 2016, Kusiak et al. 2010 and Szeląg et al. 2016, 2017.

The calculations carried out foresee a case in which on the basis of the values of the flow and temperature of the sewage at moments $(t\text{-}m)_n$ and $(t\text{-}k)_n$, the values of n sewage quality indicators are predicted, which can be recorded in the following way (Szeląg et al. 2018):

$$C(t) = f(Q(t-1), Q(t-2), Q(t-m), \ldots, T_{in}(t-1), T_{in}(t-2), T_{in}(t-k))_n \quad (4)$$

To identify the variables explaining the sewage quality indicators, the method of Fortified Trees (FT) was used, which was discussed in detail by Verma and Kusiak 2013.

The k-NN method is considered to be one of the simplest non-parametric methods, in which the value of the explained variable is determined according to the relation:

$$\hat{y} = \frac{1}{K} \cdot \sum_{m=1}^{l} y_i \cdot J(x_i, x_j) \quad (5)$$

where: x_i is one of the K closest neighbours x_j when the distance $d(x_i,x_j)$ belongs to the shortest distance between observations, l - number of observations, $J(x_i,x_j)$ - function assuming respectively values: 1 when x_i is one of the K closest neighbours x_j, 0 when the first condition is not fulfilled.

Mahalonobis distance, which is the distance between two points in the n-dimensional space differentiating the contribution of individual components, was used in the calculations. In this case, the number of closest neighbours K was searched for using the method of successive approximations until the minimum values of absolute (MAE) and relative errors (MAPE) were obtained, which is a common practice.

Much greater complexity of the model than in k-NN method, and thus greater accuracy of forecasts confirmed in numerous works (Szeląg et al. 2017), is characterized by SVM method, whose model structure resembles a neural network. In this method, the so-called kernel function is used to transform N-dimensional nonlinear space into K-linear space with a larger dimension, and the results of the simulation are determined using the formula:

$$y = \sum_{i=1}^{N_{sv}} (\alpha_i - \alpha_i') \cdot K(x, x_i) + w_0 \quad (6)$$

in which: N_{sv} - number of supporting vectors corresponding to the number of non-zero Lagrange coefficients, dependent on C and ε, α_i - Lagrange coefficients, $K(x,x_i)$ - kernel function.

The values of α_i coefficients and the number of supporting vectors (N_{sv}) are determined by minimising the expression:

$$\sum_{i=1}^{z} \frac{C}{l} \cdot |y_i - f(x_i)|_\varepsilon + \frac{1}{2} \cdot \|f_k\|^2 \quad (7)$$

where: $|y_i\text{-}f(x_i)|_\varepsilon = \max\{0, |y_i\text{-}f(x_i)\text{-}\varepsilon|\}$, ε - permissible error value, l - number of observations in the teaching data set, C - constant value dependent on ε, $\|f_k\|$ - norm f in Hilbert space.

Formulas (6) and (7) show that the predictive abilities in the SVM method are influenced by three parameters: capacity (C), kernel function and threshold of insensitivity ε=0,01, which are determined on the basis of works of Burges (1998) and Ossowski (2013). For the prognosis of quality indicators in equation (3) a sigmoidal function has been used. Optimal parameters of C and kelner function (γ=0.2÷1.0) were searched for until minimal values of MAE and MAPE were obtained. To develop the models to forecast selected indicators of sewage quality using k-NN and SVM methods the software STATISTICA10 was used.

Based on the obtained results of calculations (developed classification models for the simulation of sludge sedimentation and forecast of selected sewage quality indicators), an analysis was carried out in order to assess the impact of the uncertainty of sewage quality forecast and parametric uncertainty of classification models on the results of the calculation of the probability of exceeding the limit value of the sludge volume index SVM.

Fig. 1 shows that the calculation of the impact of the forecast error of individual sewage quality indicators on the determination of the probability p is the result of the execution of the following algorithm steps:

a) determination of n distributions of errors of the quality indicators forecast $\Delta c_t = f(\mu,\sigma)_n$ and adjustment of empirical distributions to theoretical ones;

(b) calculation of the Spearman correlation coefficient (R) value for pairs of analysed indicators; if the obtained R-value indicates a significant correlation between the analysed variables, appropriate data sampling methods, e.g. the Iman-Conover method, should be used;

(c) T-fold simulation with the Monte Carlo method of errors Δc_n of indicator forecasts on the basis of n theoretical distributions;

(d) calculation of l values for n quality indicators in T samples as $(c_{obl} + \Delta c)_n$;

(e) T-fold calculation of l values of p from formulae (1) and (2) for other explanatory variables in classification models;

(f) determination of confidence intervals for individual p-values corresponding to the used values of x_i.

If the input data concerning the sewage quality indicators are series of continuous measurements, the influence of the model parametric uncertainty on the results of the calculation of the probability p of exceeding the sludge volume index is analysed according to the calculation scheme (Fig. 1). The basis for identification of n theoretical distributions are the obtained mean values of coefficients α_i and standard deviations (σ_i), which are the parameters of normal distributions $N(\alpha_i, \sigma_i)$.

The scheme in Fig. 1 also shows that it is possible to assess simultaneously the impact of both parametric uncertainty and the forecast of sewage quality indicators on the results of the calculation of p from formulae (1) and (2). In this case the Monte Carlo method is used first to simulate T times forecast errors for l values of n quality indicators, according to the methodology described above, and then N simulations of α_i, σ_i values are performed and on this basis, based on N*T calculations, the values of confidence intervals for individual p values are determined.

RESULTS

On the basis of the collected measurements of the quantity and quality of sewage and parameters of the biological reactor (Table 1), it was found that the data collected change in a wide range, which leads to significant changes in the value of the activated sludge volume index (Szeląg and Studziński 2018, Szeląg and Gawdzik 2017).

In the time period under consideration, there were problems with activated sludge sedimentation at the sewage treatment plant, which is confirmed by both the average and maximum sludge volume index. In practice, this leads to problems in the sewage treatment plant operation and in order to eliminate them, there is a need to develop a mathematical model for forecasting the sedimentation capacity of activated sludge.

Based on the measurements acquired, with the help of STATISTICA program, classification models have been developed in which the vector constituting a linear combination of model input variables is described by the following formula:

$$X = \alpha_1 \cdot \frac{BOD_5}{TN} + \alpha_2 \cdot \frac{BOD_5}{TP} + \alpha_3 \cdot L_{N-NH_4} + \alpha_4 \cdot MLSS + \alpha_5 \cdot T_{sl} + \alpha_6 \cdot m_{PIX} + \alpha_7 \cdot DO + \alpha_e \cdot REC + \alpha_0$$

Calculated values of α_i coefficients in the logit and Gompertz models as well as measures of adjustment of the calculation results to the measurements are presented in Table 2. Analysing the data in Table 2 it was found that the developed classification models are characterized by satisfactory predictive abilities, however, a better adjustment of the calculation results to the measurements was obtained with the Gompertz model than with the logit model, which is indicated by the values of SENS (Sensitivity), SPEC (Specification) and R_z^2.

Table 1. Range of variability of parameters describing the quantity and quality of sewage and operational parameters of the activated sludge chambers.

Variable	Minimum	Mean	Maximum	1. quantil	5. quantil
Q, m^3/d	32564	40698	86592	37205	47095
T_{in}, ^0C	10.6	16.3	20.9	14.3	18.5
T_{sl}, ^0C	10.0	15.9	23.0	12.88	18.00
pH	7.2	7.7	7.8	7.50	7.70
MLSS, kg/m^3	1.98	4.26	6.59	3.85	5.86
REC, kg/m^3	6.54	8.61	9.84	8.22	9.11
RAS, %	44.6	91.4	167.5	72.54	115.25
F/M, kg BOD/kg MLSS·d	0.030	0.070	0.150	0.051	0.086
PIX, m^3/d	0	0.8	1.93	0.48	0.85
DO, mg/dm^3	0.55	2.56	5.78	1.90	2.26
SVI, cm^3/g	95	166	320	149.63	199.6
BOD_5, mg/dm^3	127	309	557	253.5	394.25
COD, mg/dm^3	384	791	1250	660.5	885.5
TSS, mg/dm^3	126	329	572	274.0	350.5
N- NH_4^+, mg/dm^3	24.4	49.4	65.9	43.8	55.0
TN, mg/dm^3	39.9	77.7	124.1	68.63	83.25

Table 2. Values of β_i coefficients in the logit and Gompertz models as well as measures of adjustment of calculation results to measurement data.

Variable	Logit model			Gompertz model		
	α_i	standard deviation	p	α_i	standard deviation	p
BOD_5/TN	0.0530	0.002	0.001	0.0530	0.002	0.001
BOD_5/TP	0.0140	0.001	0.039	0.015	0.001	0.042
$L_{N\text{-}NH4}$	0.0011	0.0001	0.035	0.0010	0.0001	0.037
T_d	-0.517	0.030	0.032	-0.517	0.024	0.016
MLSS	-2.543	0.150	0.019	-2.184	0.080	0.019
REC	1.339	0.290	0.031	1.198	0.150	0.022
DO	-1.644	0.280	0.022	-1.656	0.160	0.015
PIX	-0.723	0.210	0.026	-0.849	0.090	0.016
free word	9.965	0.058	0.023	9.563	0.060	0.001
	$R^2_{McFadden}$=0.781	SENS	0.8795	$R^2_{McFadden}$=0.958	SENS	0.9971
	$R^2_{Cox\text{-}Snell}$=0.599	SPEC	0.9886	$R^2_{Cox\text{-}Snell}$=0.834	SPEC	0.9886
	$R^2_{Negelkerke}$=0.868	R^2_z	0.9261	$R^2_{Negelkerke}$=0.919	R^2_z	0.9891
	AIC=37.787	SBC	58.882	AIC=30.822	SBC	51.916

Analysing the data in Table 2 it was also found that the developed models have identical ability to identify the considered events when SVI<150 ml/g, which confirms the value of SPEC=0.9886. On the other hand, the Gompertz model has better ability to identify cases for SVI>150 ml/g, which is indicated by SENS values, which are respectively 0.9971 and 0.8795 for Gompertz and logit model. The obtained functional relationships $p=f(x_i)$ are confirmed by analyses carried out by Flores-Alsina et al. (2013), Bayo et al. (2007), Luo and Zhao (2013) and Bezak-Mazur et al. (2016), based on the measurements collected from municipal sewage treatment plants.

In the next stage of the study, on the basis of the obtained classification dependencies (Table 2), models were developed for forecasting the flow rate and sewage quality indicators concerning BOD_5, TN, TP and N-NH_4. The variables explaining individual indicators were determined on the basis of calculated values of the importance coefficients (IMP) of the tested variables by means of the method of fortified trees FT (Fig. 2).

The results presented in Fig. 2 are confirmed by the results of calculations done by the authors within the framework of earlier investigations (Szeląg and Studziński 2017a,b) as well as by Lubos et al. 2017, which means that the quality of the analysed sewage quality indicators is primarily affected by the flow rate and temperature of sewage inflowing to the STP. This indicates that the quality of sewage in the analysed sewerage system is determined by the dilution of wastewater and biochemical reactions occurring in sewage flowing through the sewers. Based on the results from Fig. 2, models for forecasting the flow rate and sewage quality indicators were determined using the SVM and k-NN methods; the parameters of matching the calculation results to the measurements for the teaching and testing data sets are presented in Fig. 3.

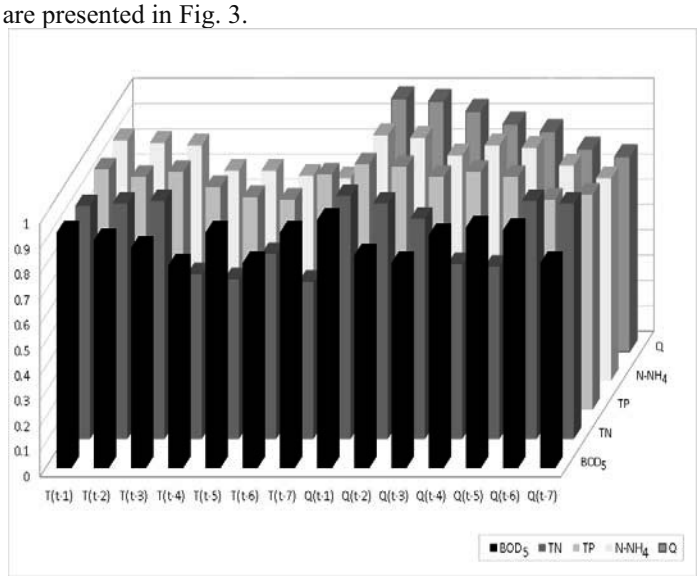


Figure 2. Values of significance coefficients (IMP) for particular variables representing the indicators of wastewater quality (BOD_5, TN, TP, N-NH_4) and wastewater flow rate.

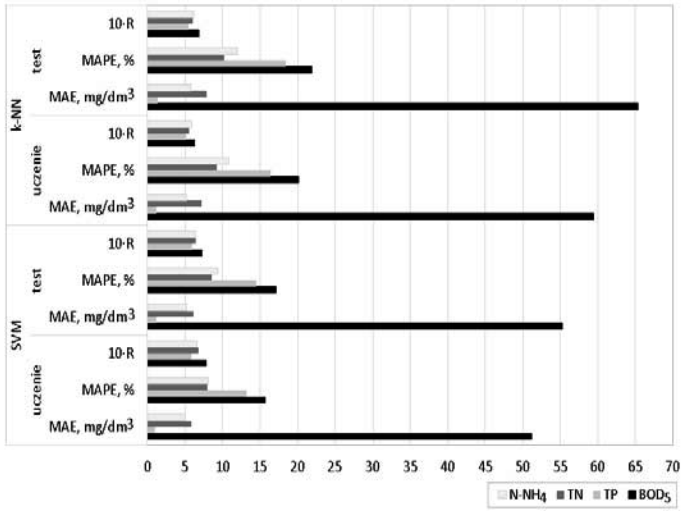


Figure 3. Determined values of adjustment parameters (MAE, MAPE, 10*R) indicating the conformity of the measurements of the analysed wastewater quality indicators with the results of calculations obtained using the SVM and k-NN methods (uczenie=teaching).

Analysing the obtained results it was found that smaller errors in matching the calculation results to the measurements were obtained by SVM method than by k-NN method. The obtained results are confirmed by analyses carried out for sewage treatment plants by Dogan et al. (2008) and Szeląg et al. (2016, 2017).

In the case of the model to simulate the intensity of sewage inflow to treatment plants using the SVM and k-NN methods, the values of forecast errors were MAE=1.923 m^3/d,

MAPE=4.57%, R=0.92 and MAE=3.446 m^3/d, MAPE=6.54%, R=0.86, respectively.

On the basis of the obtained results, the errors of forecasting the values of the analysed sewage quality indicators and flow rate were calculated, empirical distributions of errors were made and the conformity of the obtained distributions of errors with the normal distribution was checked by means of the Kolmogorov-Smirnov test. The analyses showed that there are no grounds to reject the hypothesis that the analysed distributions are normal, which is confirmed by the calculated values of p (Table 3). On this basis the parameters of the obtained distributions were determined and on their basis it was found that the mean values are equal to 0, while standard deviations for individual variables are presented in Table 3.

Table 3. Summary of test probabilities p calculated with the Kolmogorov-Smirnov test and errors in the forecasts of sewage quality indicators calculated by SVM and k-NN methods.

Quality indicator	SVM		k-NN	
	Standard deviation	p	Standard deviation	p
TP	1.44	0.18	1.62	0.23
BOD_5	64.80	0.12	70.15	0.16
TN	7.15	0.11	7.50	0.18
$N\text{-}NH_4$	6.01	0.20	6.15	0.24
Q	4291	0.06	5396	0.07

At the same time, the calculations showed that the values of the correlation coefficient R between pairs of particular variables Δc_i are smaller than R=0.26, which indicates their weak correlation and therefore the relations between these variables were omitted in further analyses.

Based on the results of the calculations, curves p=f(MLSS, DO, REC), taking into account the inaccuracy of the forecast of sewage quality indicators, are presented in Fig. 4. In Fig. 5 examples of logite curves are shown, that have been calculated taking into account the uncertainty of estimated parameters (α_i) in equations (1) and (2).

On the basis of the obtained results (Figures 4, 5) it was found that the greatest influence of the considered parameters on the accuracy of the probability of exceeding SVI_{lim} has the uncertainty of the estimated parameters in equation (2), which is confirmed by the calculated 95% confidence interval. On the other hand, the accuracy of the forecast of selected sewage quality indicators (BOD_5, TN, TP, $N\text{-}NH_4$) and flow rates, as illustrated in Fig. 5, has a smaller influence on the uncertainty of the probability forecast of the sludge volume index (SVI=150 ml/g).

Among the considered methods of sewage quality modelling, a lower uncertainty of the forecast of the probability of exceeding SVI_{lim} was obtained by SVM method than by k-NN method (Fig. 4), which is also indicated by the determined parameters of matching the calculation results to the measurements (Szeląg et al. 2017). Bearing in mind the above calculation results, Fig. 5 presents examples of logite and Gompertz curves taking into account the parametric uncertainty of models and the error of quality indicators forecast obtained exclusively by k-NN method, prepared on the basis of the calculation algorithm presented in Fig. 1.

From the calculations presented in Fig. 5 it can be stated that simultaneous consideration of the forecast errors of sewage quality indicators and model parameters indicates an increase in the uncertainty of prediction of the probability of exceeding the sludge volume index as compared to the case when the above mentioned uncertainties were considered individually. For practical applications related to the modelling and control of reactor parameters, this is an important distinction that can have a decisive influence on the proper functioning of the biological reactor.

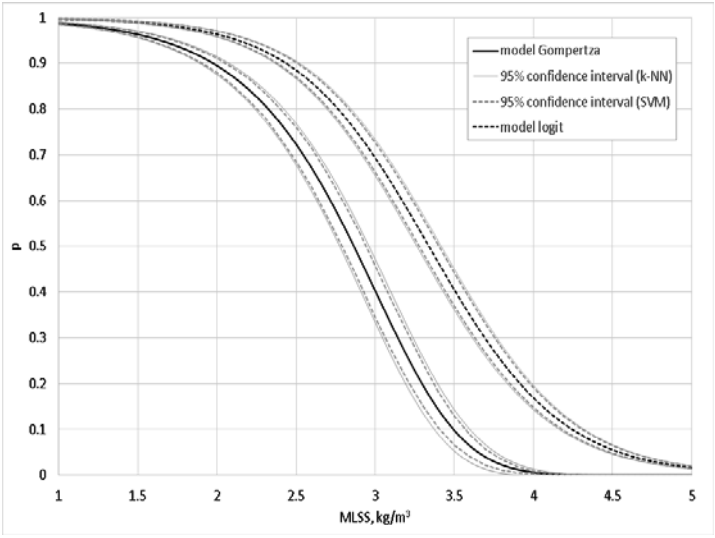


Figure 4. Examples of logite and Gompertz curves taking into account the influence of uncertainty of sewage quality indicators obtained by using k-NN and SVM methods.

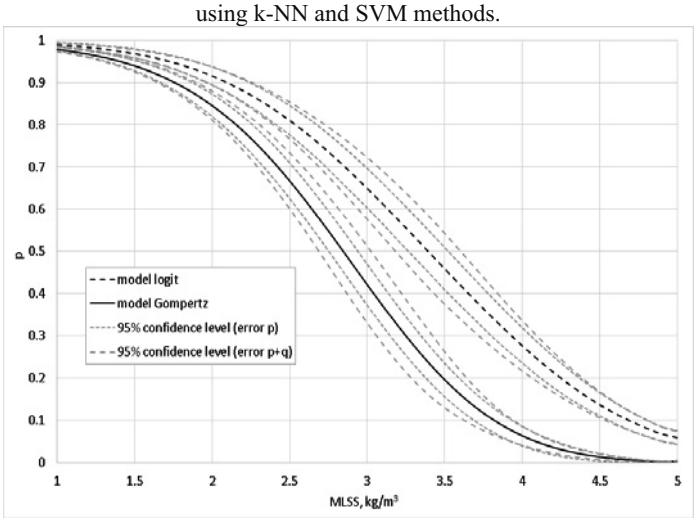


Figure 5. Examples of logite and Gompertz curves taking into account the influence of uncertainty of estimation of α_i parameters in the model forecasting the sewage quality indicators using k-NN method.

CONCLUSIONS

Mathematical models are a useful tool to forecast and control the sedimentation of activated sludge in sewage treatment plants. However, it is often the case that the explanatory variables necessary to be included in the model are difficult to measure and there are problems with obtaining sufficiently long and continuous measurement sequences for them. Then these variables can be determined by mathematical modelling, but this can lead to high uncertainty in the calculation results. The paper presents the possibility of modelling the sedimentation of activated sludge in sewage treatment plants by means of logistic regression and Gompertz methods, in which the values of explanatory variables included in the models, i.e. sewage quality indicators, were predicted using SVM and k-NN methods. Such a modelling approach gives the possibility to improve the efficiency of a biological reactor in the absence of continuous measurements of wastewater quality.

REFERENCES

Abyaneh, H.Z. (2014). Evaluation of multivariate linear regression and artificial neural networks in prediction of water quality parameters. Journal of Environmental Health Science and Engineering. 12(40), 1–8, DOI: 10.1186/2052-336X-12-40.

Al-batah, M.S., Alkhasawneh, M.S., Tay, L.T., Ngah, U.K., Lateh, H.H., Isa, N.A.M. (2015). Landslide Occurrence Prediction Using Trainable Cascade Forward Network and Multilayer Perceptron. Mathematical Problems in Engineering, 2015, 1–9, http://dx.doi.org/10.1155/2015/512158.

Bartkiewicz, L., Szeląg, B., Studziński, J., (2016). Ocena wpływu zmiennych wejściowych oraz struktury modelu sztucznej sieci neuronowej na prognozowanie dopływu ścieków komunalnych do oczyszczalni (*Assessment of the influence of input variables and the structure of the artificial neural network model on the forecasting of municipal sewage inflow to the treatment plant*). Ochrona Środowiska. 38(2).

Bayo, J., Angosto, J. M., and Serrano-Aniorte, J. (2006). Evaluation of physicochemical parameters influencing bulking episodes in a municipal wastewater treatment plant. Water Pollution VIII: Modelling, Monitoring and Management. 95, 531–542.

Bezak-Mazur, E., Stoińska, R., Szeląg, B. (2016). Ocena wpływu parametrów operacyjnych i występowania bakterii nitkowatych na objętościowy indeks osadu czynnego – studium przypadku (*Evaluation of the influence of operational parameters and the occurrence of filamentous bacteria on the volume index of activated sludge - case study*). Ochrona Środowiska. 18, 487–498.

Burges C. (1998): A Tutorial on Support Vector Machines for Pattern Recognition. In: U. Fayyad, Knowledge Discovery and Data Mining, Kluwer, 1–43.

Capizzi, G., Sciuto, G.L., Monforte, P., Napoli, Ch. (2015). Cascade Feed Forward Neural Network-based Model for Air Pollutants Evaluation of Single Monitoring Stations in Urban Areas. Int. Journal of Electronics and Telecommunications, 61(4), 327–332.

Dogan, E., Ates, A., Yilmaz, E.C., Eren, B. (2008). Application of artificial neural networks to estimate wastewater treatment plant inlet biochemical oxygen demand. Environmental progress. 27(4), 439–446.

Flores-Alsina, X., Comas, J., Rodriguez-Roda, I., Gernaey, K.V., Rosen, Ch. (2009). Including the effects of filamentous bulking sludge during the simulation of wastewater treatment plants using a risk assessment model. Water Research. 43(18), 4527–4538.

Harrell, F., (2001). Regression Modeling Strategies with Application to Linear Models, Logistic Regression, and Survival Analysis. New York: Springer.

Luo I., Zhao Y. (2012). Sludge Bulking Prediction Using Principle Component Regression and Artificial Neural Network. Mathematical Problems in Engineering. 1.

Minsoo K., Yejin K., Hyosoo K., Wenhua P., Changwon K. (2016). Evaluation of the k–nearest neighbour method for forecasting the influent characteristics of wastewater treatment plant. Frontiers of Environmental Science & Engineering. 10(2), 299–310.

Ossowski S. (2013): Neural Networks for information processing. Publishing House of the Warsaw University of Technology, Warszawa 2013.

Szeląg, B., Gawdzik, J., (2016). Application of selected methods of artificial intelligence to activated sludge settleability predictions. Pol. J. Environ. Stud. 25(4), 1709-1714, DOI: 10.15244/pjoes/62262.

Szeląg, B., Gawdzik, J., (2017). Assessment of the effect of wastewater quantity and quality, and sludge parameters on predictive abilities of non-linear models for activated sludge settleability predictions. Pol. J. Environ. Stud. 26(1), 315–322, DOI: 10.15244/pjoes/64810.

Szeląg, B., Studziński, J., (2017 a). Modelling and forecasting the sludge bulking in biological reactors of wastewater treatment plants by means of data mining methods, ISPEM, 1–10.

Szeląg B., L. Bartkiewicz, J. Studziński, K. Barbusiński (2017). Evaluation of the impact of explanatory variables on the accuracy of prediction of daily inflow to the sewage treatment plant by selected models nonlinear. Archives of Environmental Protection.

AUTOMATICALLY GENERATED REAL-TIME LOGICAL SIMULATION OF THE ELECTRICAL SYSTEM

mgr inż. Jerzy Kocerka*†, dr inż. Michał Krześlak*

*Tritem sp. z o.o.
Ligocka 103
40-568 Katowice
Poland

†Institute of Automatic Control
Silesian University of Technology
Akademicka 2A
44-100 Gliwice
Poland

Email: j.kocerka@tritem.eu, m.krzeslak@tritem.eu

KEYWORDS
Simulation, Simulation in Engineering, Hardware-in-the-loop, Real-time simulation

ABSTRACT
Embedded software testing plays an important role in quality assurance, which is especially important for complex, embedded, safety-related systems. Such systems are often subject to additional regulations regarding functional safety such as ISO 26262 norm for road vehicles or EN 50128 for the railway industry. Testing on the real object is very expensive. In order to reduce the costs and meet project deadlines hardware-in-the-loop techniques are commonly used to verify the behaviour of the control system, but building a simulation environment can be a very labour-intensive task. This paper presents a method and a tool for automatic generation of the logical model of the electrical system which speeds up and simplifies the simulation development, together with the possible, future fields of research.

INTRODUCTION

With the embedded software playing a key role in most of the modern systems it is extremely important to achieve a proper level of functional safety of such systems. One of the important parts of the quality assurance and risk management is software testing. Due to the increasing complexity of systems, the number of functions (and requirements) that need to be verified in the testing process can reach up to several thousand. To meet the expected delivery schedules it is important to make development and testing process as efficient as possible, but even then testing is one of the most time-consuming parts of the development process which takes 40-70% of the overall effort (Kosindrdec, and Daengdej 2010). An important part of this effort is test environment development, defined as building the physical test infrastructure (test rig), but more importantly preparing a proper software (simulation) to be able to perform the test.

In our study, we have used the design science research (DSR) as a methodology. With its focus on artefact development, utility, innovation, and iteration, DSR helps to develop innovative and useful artefacts, while the behavioural science helps to understand causal relationships, usually expressed in the form “does A causes B?” (Thuan et al. 2018)(Winter 2008). The method usually starts with the identification of a need (in this case, a demand for an easier, faster and more reliable method of simulating an electrical system), followed by artefact design, construction and evaluation (Vaishnavi and Kuechler 2008)

Following the DSR methodology, the paper is structured as follows. Section 0 presents an analysis of the current way of simulating the electrical system for the purpose of the embedded software testing and identification of the need for such an approach. In the following section (0) the theoretical concept of the automatically generated model together with its properties is presented. Section 0 illustrates a tool developed to verify the proposed concept and the evaluation of the tool by the industry. Finally, section 0 contains brief conclusions with the ideas for future research projects.

SIMULATION OF THE ELECTRICAL SYSTEM

Embedded software testing on the real object, such as train, is not only very expensive (may cost more than $10 000 per day (Reinholdt and Kocerka 2018), but also usually not even possible at the early stage of the project, due to the fact, that the controlled object does not exist yet. In such applications, in order to hold down the costs and meet deadlines most of the software testing is done using Hardware-in-the-loop technique.

The basic aim of the Hardware-in-the-loop system is to provide proper inputs and bidirectional communication with the System Under Test (SUT) in order to make operational in the laboratory environment. It allows performing validation and verification of the embedded software in the SUT.

In many projects, especially in the railway domain, the control system (i.e. our SUT) consists of:

- Central Processing Unit – CPU (often also called Communication and Control Unit – CCU);
- Peripheral IO stations;

- Control Units for various subsystems (e.g. Brake Control Unit, Drive Control Unit);
- Communication busses between control units and IO stations;

Quite often CCU is not just one device, but a group of two or more devices either to provide redundancy or to separate functions which require different safety integrity levels.

Depending on the current level of testing different elements of the system are simulated, but in almost every level and HIL configuration, the simulation of the behaviour of the electrical system is needed in order to provide input both for peripheral IO stations and as an input to other models (pneumatic, hydraulic, thermal).

In most cases, it is not necessary to model the exact physical behaviour of the electrical system including currents and voltages, but just the logic behind the system and determine whether the digital IO is active or not.

In a conventional approach, the electrical system model is created manually by analysis of the wiring diagrams and developed in some modelling or programming language.

Such an approach, although quite simple has several major disadvantages. First – in order to prepare the model the complete wiring diagram needs to be analysed to find the control-feedback loops, which requires a deep understanding of the electrical system. Second – the electrical system of the complex object (e.g. train) may contain more than 20 000 connections and 1 000 devices, which makes the analysis and model development process very time-consuming. Lastly – our experience has shown that people tend to simplify the model comparing to the original wiring diagrams by skipping various relations, without the proper impact analysis on the simulation behaviour and testing process.

The complexity of this task is even higher at the beginning of the project, especially when the team uses an iterative approach for software and electrical system engineering, often providing a new revision of the software and wiring diagrams.

In summary, participating in many projects related to testing the embedded software of the control systems, the need of the real-time, logical model of the electrical system was noticed, which can be used in the Hardware-in-the-Loop test environment. To ensure the accuracy of the model and to be able to create it in a short time, the model creation process should be automatized.

AUTOMATICALLY GENERATED MODEL

In order to generate the model of the electrical system automatically, it is necessary to define what are the inputs for such model, or rather, how the real electrical system's behaviour and operation can be defined.

On the logical level, the behaviour of the electrical system is, in general, defined by:

- Devices that are used in the system;
- Interconnections (wires) between the devices;

In one of the simplest possible cases, one can define an electrical system with three devices – a battery, a switch and a lamp, connected with 3 wires (Figure 1).

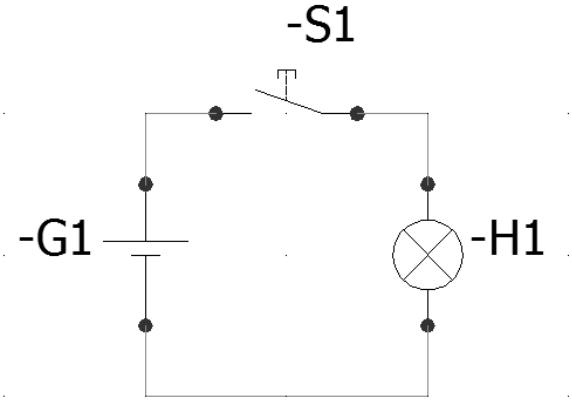


Figure 1: Example of the small electrical system

The logic of such a system can be defined on the devices level and the wires level. Table 1 shows the logic on both levels in such a simple setup.

Table 1: Devices logic

Device	Device state logic	Device connections logic
Battery	Always on	Supplies voltage on wires
Switch	State set based on the user command	Connects wires if switched on
Lamp	On if there is voltage on wires	-

In order to determine the state of the entire electrical system one has to determine the state of each wire and device. Due to the fact that the state of the device usually depends on the wires and vice versa – state of the wire depends on the state of the device the calculation has to be done in a sequential manner.

The device state is usually just a Boolean value defining whether the device is on (powered) or off (unpowered). Such definition is enough to define the state for most relays, lamps or power supplies, but it's not sufficient for more complex devices like multi-position switches, time-dependent relays or devices with internal logic (with µ-processors). In such a case, the device state needs to be represented as one or more numeric values.

Several common physical quantities can be used to define the wiring state. This includes wire potential (or voltage between wires) and the current flowing through the wire. Analyzing the requirements we defined, that it's enough to simulate the electrical system on the logical level. With such an approach, the state of each wire can be simplified to its potential and assume, that if the potential is not 0 the current will be high enough to supply devices. It can be also defined, that all the wires connected together are equipotential. To give the model more flexibility the state of the wire has been defined as a floating point numeric, where the value can be in 3 different classes:

- > 0 – wire is connected to some voltage source and can supply devices;
- 0 – wire is connected to the ground;
- -Inf – wire is not connected to any voltage source, its potential is unknown and it can not be used to supply the devices;

To be able to generate the model of the entire electrical system automatically it has to be defined which devices are used in it and how they are connected. This information can be extracted from the electrical CAD tools. Most of the tools, including industry standard tools like Siemens COMOS or Aucotec ELCAD, can export a list of devices types (defining what kind of the device it is) together with their unique identifiers on the diagram and all the connections between the terminals of the devices also with their unique identifiers (defining how the devices are connected between each other). Using this information, the only missing part is the behaviour of each device type, which has to be modelled manually based on the device documentation.

Although this process requires some effort, we have found out that most companies use standard devices in their projects. This means, that once a certain database of devices models has been defined it's enough to develop models of the missing components when starting a new project. Based on our experience we estimate that in a completely new project 10-30% of devices models need to be implemented, whether up to 90% is ready. Also modelling the simple components (like relay or switch) is very simple and can be done even by the inexperienced engineer with simple programming skills which limits the cost of creation devices database.

As defined in the requirements above, it should be possible to run the model in a real-time environment. A real-time simulator needs to solve the model equations for one time-step within the same time as in the real-world clock. Therefore, it produces outputs at discrete time intervals, where the system states are computed at certain discrete times using a fixed time-step (Faruque et al. 2015). At the beginning of the simulation (t_0), all wires should be set to an unpowered state (W_0) and all devices should be set to the default state (off) (D_0). In the first step, the model can calculate the state of the wires based on the initial state of devices ($W_1 = f(D_0)$). In a typical setup, this will power the wires based on batteries or other power supplies providing first power for the devices. Next, based on the current states of wires, the model can calculate the new state of devices for the first simulation step ($D_1 = f(W_1)$), which will determine the outputs for the UUT. In subsequent iterations the calculation process is repeated, each time calculating the state of the wires based on the state of the devices from the previous iteration ($W_n = f(D_{n-1})$) and the state of the devices based on the state of the wires from the current iteration ($D_n = f(W_n)$).

TOOL FOR AUTOMATIC MODEL GENERATION AND EXECUTION

To validate the concept of the automatically generated model of the electrical system we have developed the tool called ELMo – Electro-mechanical Logic Modelling.

ELMo was created in LabVIEW development environment (Johnson 1997) and consist of two major parts:

- ELMo parser
- ELMo engine

ELMo parser is a tool that creates proper data structures based on the export from the electrical CAD tools. It's a Microsoft Windows application running on a standard PC parsing the input files (usually in a csv format) and producing a JSON (ECMA 2017) files with the data necessary to execute the ELMo engine. ELMo parser does also a basic data validation (for example returning a log entry when there are two wires connected to the same pin) giving the instant feedback to the electrical CAD engineers if there are errors in the wiring diagram.

As a part of the full hardware-in-the-loop setup, ELMo engine can be run under National Instruments VeriStand on the Pharlap ETS or NI Linux Real-Time OS. It allows ELMo to be connected as part of a larger system enabling developers and testers to benefit from a complete suite of NI hardware modules and I/O communication protocols such as ProfiNET, Profibus, MVB, CAN or OPC-UA and interconnect the Real-Time simulation with physical devices under test.

ELMo engine uses the data structures created in ELMo parser to execute devices models and calculate the state of the electrical system as described in section 0. As the tool was part of the testing environment, ELMo engine has also the possibility to introduce errors by:

- disconnecting any wire from the device;
- forcing device or wire state;
- short-circuit two or more wires;

In order to support automatic testing, a special automation interface has been implemented to be able to perform all the actions automatically. The interface is based on the standard TCP protocol, so the test automation can be easily performed using Python or C-family languages.

As specified in the previous section, ELMo engine executes devices models and based on the state of devices determine the state of the wires. Each model is a separate LabVIEW VI and the models are called sequentially as in sequential modular (SM) simulation (Marquardt 1996). As a next step, ELMo solves the equations needed to determine the state of each wire as in equations oriented (EO) simulation. As such, the whole engine can be described as a hybrid simulation, combining both calculations methods in one tool.

As a part of our study, ELMo has been compared to other simulation tools available on the market. The analysis showed that most of them simulate the electrical system with high accuracy, but are not able to perform the simulation in the real-time (almost all tools based on Spice). On the other hand, there are several simulators capable of real-time simulation of the entire grid (e.g. RTDS Simulator from RTDS Technologies (Kuffel et al. 1995) or HYPERSIM from OPAL-RT), which are very expensive and require dedicated hardware to be run on. The only real competitor for ELMo is Siemens SIMIT Simulation Platform. Table 2 present a few basic parameters of both systems.

Table 2 Electrical system simulation tools

Tool	Company	Main properties
ELMo	Tritem	Real-time simulation of the electrical system Works on Windows and selected real-time OS Compatible with NI VeriStand, can be easily integrated with almost any HIL system
SIMIT	Siemens AG	Semi-real-time simulation of the electrical system (too slow for big setups) Works on Windows Compatible with Siemens ecosystem

The first version of ELMo was created in 2015 and shortly after it was used by Siemens AG Mobility Division to simulate the electrical system of the high-speed train (Reinholdt and Kocerka 2018). The feedback from the engineers working in this project was very positive, especially regarding the ease of creating a model, overall calculation speed and the usability of the tool.

CONCLUSION

Increasing complexity and safety demands of the embedded software systems contribute to the intensification of the testing phase of such system development. Thus testing becomes one of the most time-consuming phases mainly due to design and development of test environment, test rig and simulation. In such applications, the most common solution is to use Hardware-In-The-Loop approach, where the simulation of the electrical system behaviour is the main and most time-consuming part of the simulation.

To examine the quality and development effort of electrical system simulation one needs to take into account the necessity of the real-time simulation and the possibility to generate the model in a short time.

Successful usage of our solution called ELMo as an automatically generated logical model of the electrical system in a commercial project shows that those properties and requirements were properly identified and the overall method of the logical simulation can be valuable extension to existing or future embedded software testing systems.

Based on the success of ELMo the next stage of our research is to evaluate the possibility to use a similar approach not only in simulating an electrical system, but also to simulate pneumatic or hydraulic systems.

REFERENCES

ECMA, “The JSON Data Interchange Syntax, ECMA-404 Standard”, https://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf, 2017.

Faruque M. O., et al, "Real-time simulation technologies for power systems design, testing, and analysis.", IEEE Power and Energy Technology Systems Journal 2.2 (2015): 63-73.

Johnson G. W., LabVIEW graphical programming, Tata McGraw-Hill Education, 1997.

Kosindrdec N., Daengdej J., “A Test Case Generation Process and Technique”, Journal of Software Engineering 4(4), April 2010, pp. 265-287, 2010.

Kuffel R., et al, RTDS-a fully digital power system simulator operating in real time. In Proceedings 1995 International Conference on Energy Management and Power Delivery EMPD'95 (Vol. 2, pp. 498-503). IEEE, 1995.

Marquardt W., Trends in computer-aided process modeling. Computers & Chemical Engineering, 20(6-7), 591-609, 1996.

Reinholdt M., Kocerka J., “Reducing Risk and Cost With Virtual High-Speed and Commuter Train Test”, See http://sine.ni.com/cs/app/doc/p/id/cs-17677, 2018 (Accessed: March 1st, 2019).

Thuan N. H., Drechsler A., Antunes P, „Construction of Design Science Research Questions”, Communications of the Association for Information Systems, October 2018.

Vaishnavi K., Kuechler W., “Design Science Research Methods and Patterns: Innovating Information and Communication Technology”, Auerbach Publications, 2008.

Winter R., “Design science research in Europe”, European Journal of Information Systems 17:470–475, 2008.

WEB REFERENCES

COMOS – Plant Engineering Software, See https://new.siemens.com/global/en/products/automation/industry-software/plant-engineering-software-comos.html (Accessed: March 1st, 2019)

ELCAD and AUCOPLAN, See https://www.aucotec.com/en/products/elcad-aucoplan/ (Accessed: March 1st, 2019)

HYPERSIM Real-time Simulator, See https://www.opal-rt.com/systems-hypersim/ (Accessed, March 3rd, 2019)

What is NI VeriStand, See http://www.ni.com/veristand/whatis/ (Accessed, March 1st, 2019).

TRAFFIC SIMULATION AND URBAN MOBILITY

ACS SIMULATION AND MOBILE PHONE TRACING TO REDRAW URBAN BUS ROUTES

Giuseppe Iazeolla
University of Roma TorVergata and
University Guglielmo Marconi
Rome, Italy
iazeolla@uniroma2.it
http://www.sel.uniroma2.it/iazeolla

Fabio Pompei
University Gugliemo Marconi
Rome Italy
fabio.pompei@gmail.com

KEYWORDS
Bus transport , Bus routes, Mobile telephony, Origin-Destination Matrix, Ant Colony System modelling, Simulation modelling, Urban mobility.

ABSTRACT

A method is introduced to optimize city bus transport and match transport demand and supply. The method combines mobile phone tracing with ant colony modelling and simulation. The study is carried out through a 1-year, 24 hours/day application. The results show an increase of transported passengers by more than 150% with an improvement of transport demand (passengers/inhabitant) and supply (seats/inhabitant). The case study is the urban area of Rome, Italy. The method however applies to any town wishing to seek smart mobility.

INTRODUCTION

Public transport is designed to provide mobility either to people who cannot use their own means or as an alternative to private transport. Matching service demand and transport supply requires evaluation of many space-time aspects, related to bus transport capacity in appropriate and significant time slots.
This paper introduces a method to assess and improve the public transport capacity in connection to the potential transit demand in the municipality of a big city as Rome, Italy. The method can be extended to any town wishing to seek smart mobility.
The method uses information derived from 1) moving users data, obtained through records from mobile phones, 2) city area data, 3) open data of the bus transport in Rome, 4) census data.
A survey illustration of points 1) through 4) can be found in preliminary papers (Pompei 2017a, Pompei 2017b) that, however, do not illustrate the method.
The method is based on the use of mobile phones data, on the use of OD (origin destination) matrices, on the use of a routing-algorithm based ant colony systems (ACS) modeling, and of an algorithm to simulate the bus moves in significant time slots.
The paper is organized as follows: after the introductory Section I, Section II gives the state of the art on the use of mobile phones tracing and ACS models. Section III presents the analyzed city transport area and data. Section IV introduces the modeling approach.
Section V and Section VI illustrate the ACS algorithm and the bus algorithm, respectively. Section VII presents the method results and Section VIII the concluding remarks.

RELATED WORKS

The use of mobile phone tracing and ant colony modeling can be found in various papers dealing with local public transport analysis and improvement.
In (Yu 2005) a contribution can be found on the use of the ant colony modeling to optimize the bus traffic in the city of Dalian, China. The use of mobile phones tracing can be found in works that deal with Vehicle Routing Problems applied to delivery processes in the city of Lugano, Switzerland (Brandão 2014). The use of mobile phones data to optimize routing can be also found in (Melis et al.2016, Longo et al. 2013, Todoky et al. 2016) that see the use of mobile telephony data as a way to yield transport intelligence and enhance smart mobility in towns. In (Çolak et al. 2016) a study can be found that uses mobile phones traces in traffic peak hours of cities like Boston and San Francisco (USA), Lisbon and Porto (Portugal) and Rio de Janeiro (Brasil), to analyze congested travel problems. In (Talbot 2013) an IBM study is reported, that uses cell-phone data to redraw the Abidjan (Ivory Coast) bus routes, and in (Williams et al. 2015) a similar study for the Nairobi (Kenya) bus system.
All such works generally use either mobile phone tracing or ant colony modelling separately to study local transport problems. In this paper we combine phone tracing with ant colony modelling and simulation to redraw bus routes and match transport demand and supply.

THE ANALYZED CITY TRANSPORT

The metropolitan city of Roma is one of the 10 metropolitan cities with over 4.3 million inhabitants, served by train, underground and surface transport (Pompei, 2017b).
This paper study is limited to surface transport within an area of around 2.5 million people. The considered traffic

consists of service provided by the local transport company with over 2700 buses.
Information was derived from different data sources and related to:

- City area data (identification of areas to be analyzed),
- Population data (census data),
- Demand data (people moves among identified zones),
- Transport supply data (bus transits, stops, routes etc.),

an enormous amount of heterogeneous, anonymized, encrypted and aggregated data.
Data sources used were:

- The phone network operator database.
- The ISTAT (National Institute for Statistics) census database.
- The Rome Mobility Agency public transport database.

numbered (left-right, top-down) 1 through 3025, the upper-left zone being zone 1 and the lower-right zone 3025.
The grid can be seen as a matrix of 55 by 55 elements, with as many entries as sub-areas. Each *(i, j)-th* entry of the matrix is also called **cell**. For *i* being the row and *j* the column of the matrix, the matrix entry *(i, j)* corresponds to the *(i by 55+j)-th* zone of the grid. For the sake of brevity, such a zone will also be denoted *(i, j)-th* **zone**.
The matrix will be called origin-destination matrix (**OD matrix**) for further on cleared reasons.

The network operator data

Movement tracking was made possible through an automatic and pseudo-anonymous (to defend users privacy) processing of passive location data, collected by a system which locates all operator users, including roaming customers, by the time they interact with the network. More specifically, user location is collected basing on events such as switching on and off the terminal, location update (periodically and not), return to the coverage, switch of technology from 2G to 3G or 4G, incoming / outgoing call or text.

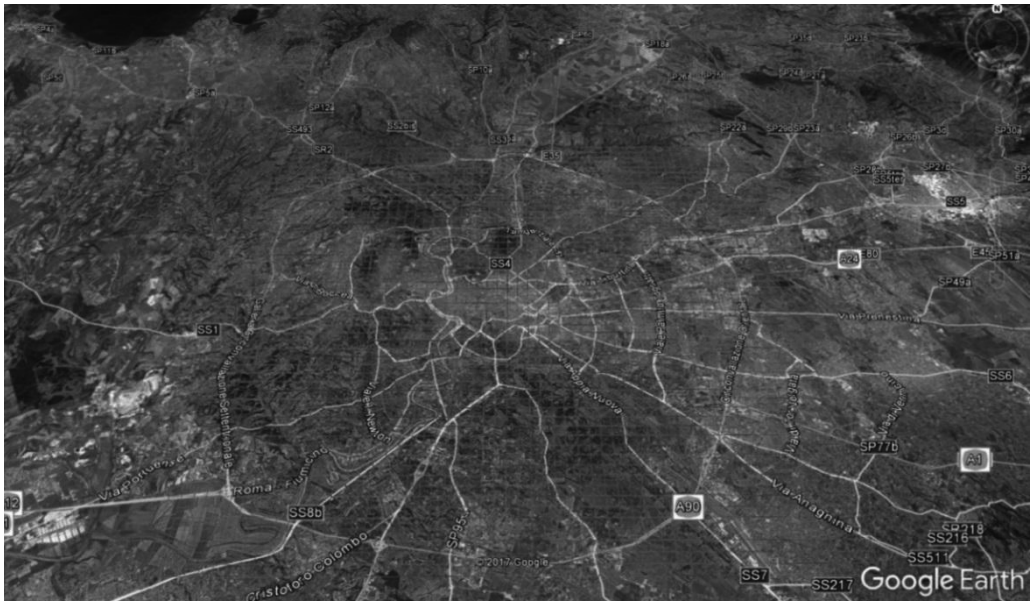


Figure 1: Grid of the Analyzed City Transport Area

The considered urban area

The considered city transport area of Rome is bounded by a rectangle with a diagonal of lower left vertex at coordinates [41°79'46"N, 12°36'79"E], and upper right vertex at coordinates [41°99'42"N, 12°63'75"E]. This area roughly includes that part of the town delimited by the Ring Highway of Rome. The area is in large part characterized by daily congested traffic that compromises the urban mobility.
For the scope of this work, the area is divided into a *grid* of 55 by 55 elements for a total of 55 by 55 = 3025 elements, see Fig.1, each element relating to a *sub-area* (also called **zone**) of 400 by 400 meters. Such 400 by 400 meters subdivision is chosen to conform to the granularity of the network operator data. The zones are

An automatic system was used to derive the OD matrices from the operator data. Number 24 matrices per day were derived, one matrix per hour. Collected data refer to a whole week and relate to a population of about 2.5 million individual users.

THE SYSTEM MODEL

The model operates by deriving progressively enriched OD matrices in an **8** steps procedure, as follows:
Step 1: Basing on the network operator data, an OD(h) matrix is derived for each hour h of the day (Fig.2). The matrix entry (i,j) gives the number $n_{ij}(h)$ of **traces** (pedestrians or cars) registered by the operator **in** the (i,j)-zone at hour h**.**

Number 24 such matrices are obtained per day, denoted OD(1), OD(2), …, OD(24).
For each amount $n_{ij}(h)$ of registered traces, the smaller amounts $n_{ij}(h)^* < n_{ij}(h)$, $n_{ij}(h)^{**} < n_{ij}(h)$, etc. are also reported, to denote the subset of $n_{ij}(h)$ traces that, according to operator records, moved from zone (i,j) **to a nearby** destination zone (i*, j*), or (i**, j**) etc. (see Step3). Assume for example that $n_{12}(h)=55$ traces were registered in zone (1.2) at hour h. Quantity $n_{12}(h)^* = 35$ may denote the subset of traces that moved from (1,2) to e.g. (2,3) and quantity $n_{12}(h)^{**} = 20$ the remaining ones that moved from (1,2) to e.g. (2,2).

	1	2	3	4	…	55
1		$n_{12}(h)$ $n_{12}(h)^*$ $n_{12}(h)^{**}$				
2						
3			$n_{33}(h)$ $n_{33}(h)^*$ $n_{33}(h)^{**}$			
4						
…						
55						

Figure 2: The OD(h) Matrix

Step 2: basing on the Mobility Agency dataset, an OD(L) matrix is derived for each bus line L. The matrix cells give the bus route from the departure zone A to the arrival zone B, with intermediate stops at zones F1, F2, F3 etc. (Fig.3). By graphically connecting such cells, an oriented graph can also be thought to graphically represent bus L route.

	1	2	3	4	…	55
1		A				
2			F1			
3				F2		
4			F3			
5				B		
…						
55						

Figure 3: The OD(L) Matrix

Step 3: by overlapping the OD(h) and the OD(L) matrices, a new matrix OD(L,h) is obtained (Fig.4) that gives the transport demand for the bus line L at hour h, with A denoting the bus departure zone, with F1, F2, F3 denoting the intermediate bus stop zones, with B the bus arrival zone, and quantities $n_{ij}(h)^*$ denoting amount of registered traces that moved between nearby zones according to the bus route (A, F1, F2, F3, B).
In the Fig.4 example, $n_{1,2}(h)^* = 35$ denotes the number of traces that moved from A to F1, and that can be assumed to represent transport demand of 35 passengers (i.e. potential passengers to use bus L to go from A to F1). Similarly $n_{2,3}(h)^* = 24$ denotes the number of traces that moved from F1 to F2, or the transport demand of 24 passengers to go from F1 to F2. And so on, the demand of $n_{3,4}(h)^*=52$ passengers to go from F2 to F3 and of $n_{4,3}(h)^*=33$ passengers to go from F3 to B.

	1	2	3	4	…	55
1		**A** **$n_{12}(h)^*=35$**				
2			**F1** **$n_{23}(h)^*=24$**			
3				**F2** **$n_{34}(h)^*=52$**		
4			**F3** **$n_{43}(h)^*=33$**			
5				**B**		
…						
55						

Figure 4: The Transport Demand OD(L,h) Matrix

Step 4: The model *first objective* is to match passenger's demand and bus capacity. According to the OD(L,h) matrix, the L line bus should be able to inboard, at hour h, number 35 passengers at the departure zone A, number 24 at the bus stop F1, number 52 at F2 and number 33 at F3. For a total of 35+24+52+33 = 144 of demanding passengers. The OD(L,h) matrix gives to the transport company useful information to predict transport demand (144) and predict the needed bus max capacity. In the simplest assumption that no passengers stay longer that one bus-stop (i.e. that all 35 outboard at F1, all 24 at F2, all 52 at F3 and all 33 at B) the required max capacity is 52 seats. In reality one has to consider the "get-in and get-out" process, and this is done by the Capacity Redraw procedure of the Bus Algorithm" (see Sect VI).
Step 5: The model *second objective* is to identify the optimal demand and supply route for the L line bus between A and B. The ACS Algorithm (illustrated in

SectV) finds the series of zones (i,j) with the max number of mobile phone traces.
Fig.5 (where, for the sake of simplicity, we use numbers in parenthesis to denote quantities $n_{ij}(h)$*) shows an example of optimal route *P'* (called *ACS route*) i.e. *A'*=(1,2), *F1'*=(1,3), *F2'*=(2,4), *F3'*=(4,4), *B'*=(5,4), in italics, found by the algorithm that redraws the original route P i.e. A= (1,2), F1=(2,3), F2=(3,4), F3=(4,3), B=(5,4) used by transport company. As seen further on, ACS identifies a max-flow, min-cost route, by redrawing the bus route in a way to identify route *P'* that includes the **max** number of traces with **min** path length increase.
The ACS algorithm of Sect V illustrates how the **max** *flow* path is identified. The **min** *path length increase* is instead easily obtained by identifying for *P'* the mostly adjacent zones to P. The mostly adjacent zone to (i,j) is defined as the zone that differs by only one unit in *i* or *j*. In the Fig.5 example, the mostly adjacent zone to F1= (2,3) of P can be *F2'*= (2,4) of *P'*, rather than, say, zone (2,5) that would differ by two units in j.

	1	**2**	**3**	**4**	**…**	**55**
1		**A/*A'*** **(35/*35*)**	***F1'*** ***(34)***			
2			**F1** **(24)**	***F2'*** ***(60)***		
3				**F2** **(52)**		
4			**F3** **(33)**	***F3'*** ***(40)***		
5				**B/*B'***		
…						
55						

Figure 5: The OD(L,h) Matrix with the Original Route P and the ACS Route (in italics) *P'* of Bus.

Step 6: consequence of the foregoing Step 5 is that, being the redrawn route *P'* mostly adjacent to the original route P, the transport company can serve *P'* with small variants to the original route and nevertheless improve the number of served passengers. The L line bus can indeed serve number 35 + 34 + 60 + 40 = 169 passengers if the redrawn *P'* route is used in place of the original route P, along which only 144 passengers were instead served.
By use of the Fig.5 OD(L,h) matrix, the transport company may evaluate the economical return of carrying 169-144=25 more passengers at hour h at the cost of an eventually larger capacity bus (60 rather than 52 seats), and an eventually increased route length. The reasoning will obviously be extended to the 24 hours of the day.
Step 7: In conclusion, the model can be used by the transport company to identify, for each L line bus and hour h of the day:

7.1 For the original route P, the number (144) of served passengers, and the necessary max capacity (52 seats) of the bus.

7.2 For the ACS redrawn route *P'*, the number (169) of served passengers, the necessary max capacity (60 seats) of the bus and the eventually extended route length.

Step 8: By extending the Step 7 evaluations to all bus lines of the considered urban area, and all hours h of the day, the transport company may predict the difference between the economic yield of routes P and *P'* and between the costs to face (e.g. bus of larger capacity, increased distance in km between the visited points, etc.) in the 24 hours of the day.

THE ACS ALGORITHM

The early ACS (Ant Colony System) Algorithm is a known algorithm (Dorigo et al. 1999, Dorigo et al. 2003) born as *shortest path* algorithm. The algorithm is based on the swarm intelligence of ant colonies and is used to heuristically solve transport network problems and routing problems in a computational-time shorter than the time of exact or approximate mathematical algorithms (Montemanni et al. 2002).
The algorithm is inspired by the movements of ants that heuristically identify the shortest path from nest to food. Initially, an explorer-ant seeks the source of food through an arbitrarily chosen route. Once found, the explorer returns to the anthill and leaves behind a trail of pheromone. Subsequent ants will travel the trail left by the explorer and, on the way back, will leave their own pheromone trail, which will be followed by subsequent companions and so on. In this process, the trail that will be mostly strengthened becomes the shortest, being traveled by the majority of ants per unit of time. In the end, the whole anthill will use this trail to reach the food.
The ACS Algorithm of this work is a modification of the early ACS Algorithm. It is *max-flow min-cost* algorithm, rather than a shortest path one, but maintains the computational-time effectiveness of the original ACS. Its *objective function* is finding the route with **max** number of phone-traces and **min** path-length.
The algorithm assumes that there exists only one ant, which simulates the moves of an explorer of the transport company. The pheromone is predefined and is given by the mobile phones tracks recorded by the network operator in the various areas. The explorer moves from the starting zone A to the arrival zone B of route P and identifies the adjacent zones of P with the highest number of tracks *(max flow)*, but not too distant *(min cost)* from P. So, starting from A, the explorer identifies the zone adjacent to A with the max number of traces and that, however, is not too distant from A. As seen in the Fig.5 example, he will choose zone *F1'* = (1,3) rather than F1 = (2,3). In the case, however that, for example, zone (1,1), even if adjacent to A, had more traces than *F1'*, the explorer would not have chosen it because (1,1) would deviate from the original track P more than *F1'*. Subsequently, the explorer moves from *F1'* to an adjacent zone with more tracks [the *F2'* = (2,4) of *P'*, rather than the F2 = (3,4) of P] and so on. In so doing, the explorer will reach destination B and identify the route *P'* with more phone tracks than P but not much deviating from P. The details of the algorithm can be found in (Pompei 2017c).

THE BUS ALGORITHM

The Bus Algorithm is introduced to simulate the bus movements and compare, for each bus line L of the 531 urban lines, the bus performance on the original route P and the ACS redrawn route *P'*. Simulation is stepwise discrete event simulation. The algorithm receives in input the OD(h) matrices for hours h = 1, ..., 24 and the OD (L, h) matrices which, for each line L, give the original routes and the ACS redrawn routes.
(Since the OD (L, h) matrix may change at each hour of the day, the ACS might suggest different routes during the day. This would mean dynamically changing the bus itinerary during the day. It is thus the company role to fix the route basing on an average
OD (L, h) matrix, the averages being periodically made, on a monthly or bi-monthly or etc. basis to find the best compromise between economic return and quality of service.)
The algorithm also receives in input, for each of the 531 lines, the bus departure times from the start-to-run zones A. Since each zone is 400 meters long and the average bus speed is assumed i.e. of 200 meters per minute, the algorithm cyclically examines the bus status every 2 minutes.
The time is checked within the cycle: at each clock-hour (00:00 h, 01:00 h, 02: 00 h, ...), a new OD (L,h) matrix is loaded for each bus line L.

For each vehicle, the Capacity Redraw procedure (see Step 4 of Section IV) is invoked at each step to refresh of the OD (L, h) matrix. It is also considered that the bus that follows a previous one will load a smaller number of passengers, if some have been loaded by the previous one.

The Capacity Redraw procedure manages the "get-in and get-out" of passengers in/out the buses and is operated whenever a bus has to move from one zone to another. If at a certain zone passengers wish to get-in to reach one of the subsequent zones of the route and seats are available, the passengers are made to get-in (and the value n_{ij} (h)* is increased on the OD (L, h) matrix). Once at the arrival zone, passengers are made to get-out, thus allowing new passengers to get-in. The details of the algorithm can be found in (Pompei 2017c).

RESULTS

Model results compare the performance of the original routes P and of the ACS routes *P'* for the total of 531 bus lines L. Comparison is based on 6 *performance parameters* such as: 1) the amount of carried passengers, 2) the route lengths, 3) the bus seats, 4) the unitary demand (passengers/inhabitants), 5) the unitary supply (seats/inhabitants) and 6) the demand/supply ratio.
Table 1 gives the number of passengers carried along the original routes P in the 24 hours of the day (column "h"). Column "Pot.dem" gives the potential demand (total number of phone traces), column "Unused" gives the passengers who did not use the bus, column "Used" gives the passengers that used it.

Table 1: Passengers Carried along the Original Routes P

h	Pot.dem.	Unused	Used
1	239.376	190.781	48.595
2	232.449	208.462	23.987
3	155.415	141.300	14.115
4	234.360	214.633	19.727
5	171.192	155.342	15.850
6	211.821	167.828	43.993
7	393.024	304.629	88.395
8	886.587	683.206	203.381
9	1.225.332	946.864	278.468
10	1.233.795	942.229	291.566
11	1.193.826	907.655	286.171
12	1.220.847	931.916	288.931
13	1.233.009	940.036	292.973
14	1.299.210	1.001.279	297.931
15	1.196.361	920.157	276.204
16	1.188.357	914.143	274.214
17	1.160.328	889.818	270.510
18	1.151.448	886.205	265.243
19	1.083.366	832.400	250.966
20	923.106	708.305	214.801
21	707.700	538.264	169.436
22	490.767	372.509	118.258
23	346.488	262.967	83.521
24	121.320	90.997	30.323
	18.299.484	14.151.925	4.147.559

Yearly req.	6.679.311.660
Yearly trans.	1.165.671.457
Transported %	22,7%

Table 2: Passengers Carried along the Redrawn Routes *P'*.

h	Pot.dem.	Unused	Used	Difference
1	239.376	162.451	76.925	28.326
2	232.449	186.966	45.483	21.492
3	155.415	128.377	27.038	12.918
4	234.360	198.495	35.865	16.133
5	171.192	141.096	30.096	14.242
6	211.821	136.712	75.109	31.112
7	393.024	253.172	139.852	51.452
8	886.587	591.382	295.205	91.819
9	1.225.332	819.294	406.038	127.565
10	1.233.795	804.145	429.650	138.080
11	1.193.826	769.734	424.092	137.917
12	1.220.847	785.766	435.081	146.145
13	1.233.009	795.142	437.867	144.889
14	1.299.210	849.153	450.057	152.121
15	1.196.361	782.431	413.930	137.722
16	1.188.357	780.062	408.295	134.077
17	1.160.328	759.879	400.449	129.935
18	1.151.448	759.849	391.599	126.351
19	1.083.366	709.902	373.464	122.493
20	923.106	600.121	322.985	108.179
21	707.700	451.577	256.123	86.683
22	490.767	309.050	181.717	63.454
23	346.488	216.466	130.022	46.497
24	121.320	75.365	45.955	15.628
	18.299.484	12.066.587	6.232.897	2.085.230

Yearly req.	6.679.311.660
Yearly transp.	1.751.755.702
Transported %	34,1%

The results show that the amount of passengers carried in the 24 hours (and over the year) is around 22,7 % of demand, if the original routes P are used.
Table 2 instead reports similar data for passengers carried along the redrawn routes *P'* and comparisons

with Table 1. Results show that the amount of passengers carried in the 24 hours (and over the year) is increased to around 34% of demand, that means an increase of more than 2 million transported passengers per day, corresponding to an increase of about 150%
Table 3 shows the further (2 through 6) performance parameters for the P and *P'* routes.

Table 3: Further Performance Comparison of routes P and *P'*

Parameters	P	P'
Route lengths (km)	96.820.922	105.597.274
Bus seats	66.315.700	72.326.900
Demand (passg/inhabitants)	1,04	1,56
Supply (seats/inhabitants)	16,58	18,08
Demand/Supply ratio	15,99	11,60

The Table shows an increase in the route length in the redrawn case, corresponding to an effective 11% longer than the original route. Also an increased number of necessary seats. Such two elements may result in increased costs that however could be rewarded by the larger number of transported passengers. One should also consider that the largest increment in transported passengers (around 200%) occurs in the hours h=1 through 7 of lesser efficiency of the original lines.
The remaining data in Table 3 show a capacity deficit of local public transport in comparison with the expectations of the population of the area under examination.

CONCLUSIONS

A method has been introduced to improve the bus transport of a big city like Rome by redrawing routes and matching transport demand and supply. The method can be extended to any local public transport system wishing to seek smart mobility in otherwise congested towns.
The model is based on the use of OD (Origin Destination) matrices, of a modified Ant Colony System (ACS) Algorithm and of a Bus Algorithm that simulates the bus movements.
It derives information from different data sources with different formats and related to:
City area data (identification of areas to be analyzed), Population data (census data), Demand data (people moves among identified zones), Transport supply data (bus transits, stops, routes etc.), an enormous amount of heterogeneous, anonymized, encrypted and aggregated data.
The study was carried out through a 24hour/day bus movement simulation, both along the original routes and along the routes redrawn with the ACS Algorithm. The results show that the redrawn routes yield an increase of transported passengers by more than 150% with an increase in the transport demand and supply. The paper shows how mobile telephony technology and simple modeling can be used to improve the urban mobility. Limitations of the proposed method can be found in the granularity of the model, whose the level of detail is at cell-level rather than at a city route-level.

REFERENCES

Brandão J., "Iterated Local Search Algorithm for the Vehicle Routing Problem with Backhauls and Soft Time Windows", ISEG - Lisbon School of Economics and Management, REM, Universidad de Lisboa, Working Papers REM, 2014.

Dorigo M., Di Caro G., Gambardella L.M., 1999 "Ant Algorithms for Discrete Optimization". Artificial Life, Massachusetts Institute of Technology, 1999.

Dorigo M., Stützle T., 2003, "The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances". In Glover F., Kochenberger G.A. (eds), Handbook of Metaheuristics. International Series in Operations Research & Management Science, Vol. 57. Springer, Boston, MA, 2003.

Longo M., Akram Hossain C., Roscia M. "Smart Mobility for Green University Campus", 2013 PES Asia-Pacific Power and Energy Engineering Conference, 2013.

Melis A., Mirri S., Prandi C. "Crowd Sensing for Smart Mobility through a Service-Oriented Architecture", IEEE International Smart Cities Conference, 2016.

Montemanni R., Gambardella L.M, Rizzoli A.E, Rizzoli, Donati A.V, "Ant colony system for a dynamic vehicle routing problem", Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), CH-6928 Manno-Lugano, Switzerland, 2002.

Pompei F., 2017a "Ant Colony Optimization and geolocation technologies for the transportation assignment problem - Use of ACO and geolocation technologies for performance improvement and test of Local Public Transport". COMPSAC 2017, IEEE Computer Society Signature Conference on Computers, Software and Applications, July 2017.

Pompei F., 2017b, "Geolocation for LPT - Use of geolocation technologies for performance improvement and test of Local Public Transport". EEEIC 2017 – IEEE 17th International Conference on Environmental and Electrical Engineering, June 2017.

Pompei F., 2017c, "Use of geolocation technologies to optimize the Local Public Transport " (Uso di tecnologie di georeferenziazione per l'ottimizzazione del trasporto pubblico urbano), PhD Thesis in Electronics Engneering, University of Roma TorVergata, prof. Giuseppe Iazeolla teacher-guide , April 2018.

Serdar Çolak, Antonio Lima & Marta C. González "Understanding congested travel in urban areas", Nature Communications, 7, 2016.

Talbot, D., "Cell-phone data to redraw African bus routes", MIT Technology Review, April 2013.

Tokody D., Holicza P., Schuster G. "The smart mobility aspects of Intelligent RailWay", IEEE 11th IEEE International Symposium on Applied Computational Intelligence and Informatics, 2016.

Yu, B. "Optimizing bus transit network with parallel ant colony algorithm", Proceedings of the Eastern Asia Society for Transportation Studies. Vol. 5, 2005.

Williams S., White A, Waiganjo P., Orwa D., Klopp J., "The digital matatu project: Using cell phones to create an open source data for Nairobi's semi-formal bus system", Journal of Transport Geography, 49-2015, 39-51.

ACKNOWLEDGEMENTS

The authors are grateful to the ISC'2019 reviewers for their suggestions that improved the quality of paper.

POLICY APPRAISAL USING ITERATION GAMES IN ARTIFICIAL TRANSPORTATION SYSTEMS

Hajar Baghcheband, Zafeiris kokkinogenis, Rosaldo J. F. Rossetti
Artificial Intelligence and Computer Science Lab (LIACC), Department of Informatics Engineering (DEI)
Faculty of Engineering, University of Porto (FEUP)
Rua Dr. Roberto Frias, S/N – 4200-465 Porto, Portugal
E-mail: {up201808221, kokkinogenis, rossetti}@fe.up.pt

KEYWORDS
Artificial Transportation Systems, Agent-based Simulation, Minority Games, Policy Evaluation

ABSTRACT

Traffic congestion threats the vitality of cities and welfare of citizens. Transportation systems are using various technologies to allow users to adapt and make different decisions towards transportation modes. Modification and improvement of this systems affect commuters perspective and social welfare. In this study, the effect of road flow equilibrium on commuters utilities with different types of transportation modes will be discussed. A simple network with two modes of transportation will be illustrated and three different cost policies were considered to test the efficiency of reinforcement learning in commuters daily trip decision-making based on time and mode. The artificial society of agents are simulated to analyze the results.

INTRODUCTION

Transport systems entail an activity in which something is moved between the source and the destination by one or several modes of transport. There are five basic modes of transportation: road, rail, air, water, and pipeline (Ebrahimi et al. 2007). An excellent transport system is vital to support a high quality of life, making places accessible and bringing people and goods together. Information and Communication Technologies (ICT) help to achieve such a high level objective by enhancing transportation through intelligent systems.

Among all types of transport, road transport plays an important role in daily trips. People migrate into cities because of the benefits from services and employment opportunities as compared to rural areas. However, it is becoming harder to maintain road traffic in a smooth working order. Traffic congestion is a sign of a citys vitality, and policy measures like punishments or rewards often fail to create a long term remedy. The rise of ICT enables provision of travel information through advanced traveler information systems (ATIS). Current ATIS based on shortest path routing might expedite traffic to converge towards the suboptimal User Equilibrium (UE) state (Klein et al. 2018).

Traffic congestion is one reason of negative externalities, such as air pollution, time losses, noise, and decreasing safety. As more people are attracted into cities, future traffic congestion levels are not expected to decrease but rather will increase, and extending road capacity would not solve congestion problems. While private cars maximize personal mobility and comfort to some extent, various strategies have attempted to discourage car travel to the detriment of an increased public transportation use. To encourage commuters to shift from private car to public transport or experiment inter-modal changes, it is imperative to provide users with a competitive quality public transport as compared to its private counterparts. This can be measured in different aspects such as safety, comfort, information, and monetary cost, but more importantly travel times comparatively to those resulting private cars (Tlig and Bhouri 2011).

Moreover, policy measures in transportation planning aim at improving the system as a whole. Changes to the system that result in an unequal distribution of the overall welfare gain are, however, hard to implement in democratically organized societies (Grether et al. 2010). Different categories of policies can be considered in urban road transportation: negative incentives (Rouwendal and Verhoef 2006), positive incentives or rewards (Ben-Elia and Ettema 2011, Bliemer and van Amelsfort 2010), and sharing economy (Klein et al. 2018, Kaplan and Haenlein 2010). Traditional transport planning tools are not able to provide welfare analysis. In order to bridge this gap, multi-agent micro-simulations can be used. Large-scale multi-agent traffic simulations are capable of simulating the complete day-plans of several millions of individuals (agents) (Grether et al. 2009). A realistic visualization of an agent-based traffic model allows to yield visually realistic reconstructions of modern or historical road traffic. Furthermore, the development of a complex interactive environment can help scientists to open up new horizons in transport modeling by the interactive combination of a traffic simulation (to change traffic conditions or to yield emergence resulting from road interactions) and by enhanced visual analysis (Golubev et al. 2018). The main goal of this study is to de-

velop a model, based on the concept of minority games and reinforcement learning, to achieve equilibrium flow through public and private transportation and to investigate the effect of cost in on the process of selecting transportation modes. Minority game is applied to consider rewards, positive policy, for the winner whereas learning is a tool to enhance the user's perception of utility based on such rewards. To illustrate, an artificial society of commuters is considered and instantiated on simple network with two transportation modes: public (PT) and private (PR).
The remaining parts of this paper are organized as follows. In Section II we discuss the conceptual framework, which consists of definitions of user utility, minority game, and reinforcement learning algorithms. Illustration scenarios of the network and commuters, as well as the preliminary setup are explained in Section III. Experiments and results are shown in Section IV, and the related work is revisited in Section V. Conclusions on the hypothesis and on results are drawn in Section VI.

DESCRIPTION OF THE CONCEPTUAL FRAMEWORK

In this section, both theoretical and methodological aspects are described. Details on the network and model design will be discussed as well.

Traffic Simulation

Traffic simulation models are classified into macroscopic and microscopic models. The hydrodynamic approach to model traffic flow is typical for macroscopic modeling. With this kind of approach one can only make statements about the global qualities of traffic flow. For observing the behavior of an individual vehicle a microscopic simulation is necessary. Because traffic cannot be seen as a purely mechanical system, a microscopic traffic simulation should also take into consideration the capabilities of human drivers (e.g., perception, intention, driving attitudes, etc.)

Network Design

The network is formally represented as graph $G(V, L)$, in which V is the set of nodes such as Origin, Destination, and middle nodes, and L is the set of roads (edges or links) between nodes (Klein et al. 2018, Kokkinogenis et al. 2014). Each link $l_k \in L$ has some properties such as mode, length, and capacity. In addition, a volume-delay function (eq. 1 is used to describe the congestion effects, that is, how the reaching capacity of flow in a link affects the time and speed of a journey, as represented by the equation below (de Dios Ortuzar and Willumsen 2011):

$$t_k = t_0 k * [1 + \alpha(X_k/C_k)^{\beta}] \quad (1)$$

Where $t_0 k$ is the free flow travel time, X_k is the number of vehicles, and C_k represents the capacity of the link k. In this equation, α and β are controlling parameters.

Commuters Society

Commuters, agents of the artificial society, have some attributes regarding travel preferences. These attributes can have many interpretations, such as time (desired arrival time, desired travel time, mode of transportation, mode flexibility), cost (public transportation fare, waiting time cost, car cost if they have), socioeconomic features (income), and so forth.
Agents will learn with their experiences and make decision as for their daily plan based on their daily expectations and previous experiences. The iteration module generates the demand of the transportation modes and desired times. Daily trips are scheduled for a given period of the day, to which a set of origins and destinations are defined with the respective desired departure and arrival times to and from each node.
A utility-based approach is considered to evaluate travel experience and help agents make decisions. Total utility of commuter j is computed as the sum of individual contributions as follow:

$$U^j_{total} = \sum_{i=1}^{n} U^j_{perf,i} + \sum_{i=1}^{n} U^j_{time,i} + \sum_{i=1}^{n} U^j_{cost,i} \quad (2)$$

where U_{total} is the total utility for a given plan; n is the number of activities, which equals the number of trips (the first and the last activities are counted as one); $U_{pref,i}$ is the utility perceived for performing activity i; $U_{time,i}$ is the (negative) utility perceived as time, such as travel time and waiting time for activity i; and $U_{cost,i}$ is the (usually negative) utility perceived for traveling during trip i.

Performance Utility To measure the utility of selecting activity i, each mode of transportation has different variables. For public mode, comfort level and bus capacity are considered, whereas for private, pollution and comfort level are the variables accounted for.

Time Utility The measurement of the travel time quantifies the commuters perception of time based on various components like waiting and in-vehicle travelling. Waiting time indicates the service frequency of public transportation. In-vehicle travelling time is the effective time taken to travel from a given origin to a given destination.

Monetary Cost Utility Monetary cost can be defined as fare cost of public transportation, cost of fuel,

tolls (if those exist), car insurance, tax and car maintenance, for instance. This kind of cost will be measured based on the income of commuters.

Minority Games

A common assumption is that drivers choose the route between an origin-destination (OD) pair according to the principle of minimum experienced travel time (Chiu et al. 2010). As there are other drivers on the routes, the travel time between an OD pair depends on the choices of those other drivers who also aim to minimise their travel time. When all drivers succeed in choosing the optimal route that minimises their travel times, this is referred to as Equilibrium or User Equilibrium.

Challet and Zhangs Minority Game (MG) model (Challet and Zhang 1997) is one such approach in which coordination among the agents occurs through self-organisation with minimal information and without communication between agents. Route and mode choice can be seen as a problem of self-organisation, and thus iteration game agents can reach equilibrium. Therefore, the MG might be well suited for solving this problem.

The approach consists of a set of commuters, without the possibility of communication between agents who have to organise themselves while they are in a competition for a limited resource (road, bus-seat, etc.), and there is no solution deductible a priori. Here, every commuter has to choose a given transportation mode, using a predictor of the next attendance. It is given that the agents try to avoid congested situations; however, since there is no single predictor that can work for everybody at the same time, there is no deductively rational solution.

This kind of approach was originally developed as a model for financial markets, although it has been applied to different applications such as public transportation (Bouman et al. 2016), route choice (Chmura et al. 2005), road user charging scheme(Takama and Preston 2008), among many others.

Reinforcement Learning

Reinforcement learning (RL) is a class of machine learning technique concerned with how agents ought to take actions in an environment so as to maximise cumulative reward. Roth and Erev (Roth and Erev 1995) developed an algorithm to model how humans perform in competitive games against multiple strategic players. The algorithm specifies initial propensities q_0 for each of N actions and based on reward r_k for action a_k the propensities at time $t+1$ are defined as:

$$q_j(t+1) = (1-\phi)q_j(t) + E_j(\epsilon, N, k, t) \qquad (3)$$

$$E_j(\epsilon, N, k, t) = \begin{cases} r_k(t)(1-\epsilon) & \text{if } j = k \\ r_k(t)(\epsilon/N-1) & \text{otherwise} \end{cases} \qquad (4)$$

Where ϕ is a parameter that represents the recency of forgetting, whereas "" is an exploration parameter. The probability of choosing action j at time t is:

$$P_j(t) = q_j(t) / \sum_{n=1}^{N} [q_n(t)] \qquad (5)$$

ILLUSTRATIVE SCENARIO

In the simulation phase, the perspective of the conceptual framework was considered in a simple scenario where commuters make decision over transportation mode and departure time during the morning high-demand peak hour. The simulation model was implemented in the NetLogo agent-based simulation environment (Wilensky 2014).

Network and Commuters

In this study, two different links of two modes (PT or PR) encompass two middle nodes each. As shown in Figure 1 and for the sake of simplify, the upper link is for private and the other one is for public transportation where each road is composed of one-way links.

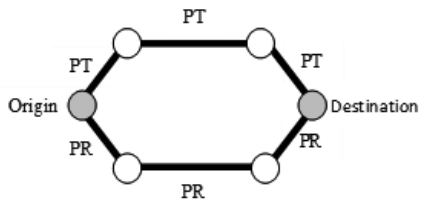


Figure 1: A bi-modal transportation network

Commuters, as a type of agents, are defined by a number of state variables which are: i) desired departure and arrival times; ii) experienced travel time; iii) the uncertainty they experienced during the trip with a given transportation mode; iv) a set of preferences about the transportation mode; v) the perceived comfort as personal satisfaction for the mode choice; and vi) a daily income variable. While the agent experience its travel activities, the costs associated with the different transportation modes, the perceived satisfaction of travelling (expressed in terms of travel times and comfort) and rewards earned by winners will have a certain impact on its mode and time choices.
Commuters can choose between travelling by PT or PR modes based on the own-car value. The decision-making process of each agent is assumed to maximise the utility and flow equilibrium on roads. They perceive current

traffic condition as well as previous experience and use this information in making other decisions.
With regard to the different utilities, the total utility of public and private modes can be measured as follow (we omit the index i indicating the activity):

$$U_{pr}^{total} = \sum_{j=1}^{n} U_{pr}^{j} \tag{6}$$

$$\begin{aligned} U_{pr}^{j} = {} & \alpha_{late}(t_{tt,exp}^{j} - t_{tt}^{j}) \\ & - (\beta_{PR} cost_{PR}/income_{j}) \\ & - \alpha_{pollution} t_{tt}^{j} pollution \\ & + \alpha_{com_{PR}} t_{tt,exp}^{j}/t_{tt}^{j} \end{aligned} \tag{7}$$

$$U_{pt}^{total} = \sum_{j=1}^{n} U_{pt}^{j} \tag{8}$$

$$\begin{aligned} U_{pt}^{j} = {} & \alpha_{late}(t_{tt,exp}^{j} - t_{tt}^{j}) \\ & - (\beta_{PT} cost_{PT}/income_{j}) \\ & + \alpha_{com_{PT}} t_{wt,exp}^{j}/t_{wt}^{j} \\ & + \alpha_{cap} t_{tt}^{j} capacity_{exp}^{j}/bus_{capacity} \end{aligned} \tag{9}$$

where t_{tt}^{j} and $t_{tt,exp}^{j}$ are total travel time and expected total travel time of agent j, $cost_{PR}$ is the monetary cost of private transportation (fuel, car maintenance, etc.), $cost_{PT}$ is the fare of public transportation, $income_{j}$ is the agents income per day, *pollution* is the amount of pollution produced by private vehicles, $capacity_{exp}^{j}$ and $bus_{capacity}$ are expected capacity of bus and total capacity of each bus respectively, $t_{wt,exp}^{j}$ is the expected waiting time and t_{wt}^{j} is the waiting time by agent j. α_{late}, β_{PT}, β_{PR}, $\alpha_{pollution}$, $\alpha_{com_{PR}}$, $\alpha_{com_{PT}}$, and α_{cap} are considered as marginal utilities or preferences for different components.
At the end of the journey each commuter memorises the experienced travel time, costs, crowding level (for PT mode users only), as well as emissions. These variables will be used to calculate the following days utility. After that each agent evaluates its own experience, comparing the expected utility to the effective utility. Based on the minority game concept, we considered the number of commuters on each road and type of transportation, and according to the Roth-Erev learning model, reward was assigned to winner who is in minority number and follows the criteria below:

- Their obtained utility $U_{effective}$ is greater than the utility prediction $U_{expected}$ as below:

  $$U_{effective} > \alpha U_{expected}$$

  where α is the marginal preference.

- The obtained utility of agent is higher than the mean utility in whole network :

  $$U_{effective} > U_{N}$$

  where $U_{N} = 1/N \sum_{j=1}^{N} U_{effective}^{j}$

Based on the reward, the effective utility they perceived in their daily trips, car-ownership and mode-flexibility, each commuter decides whether to opt for a new mode and time.

Cost Policy

In different research, the effect of cost was studied and different policies were proposed. In this paper, three different cost policies were defined, the cost of public transport is constant, and the cost of private transport will be changed as follows:

1. *Private cost is triple of public fare (Policy 1)*;
2. *Double of public fare is considered as private cost (Policy 2)*;
3. *The same cost for both mode of transportation (Policy 3)*;

Based on these policies, the simulation will be performed allowing for results to be collected and analysed.

Initial Setup

The capacity for all links of the network was considered 150 vehicles and max capacity for each bus was 70 passengers. A population consisting of 201 commuters was created, odd number to coordinate with minority game, and they iterated their daily trips in 60 days. They were characterised by the number of attributes such as departure and arrival times, mode, daily income, car-ownership and flexibility. Car-ownership is a Boolean variable and indicates whether the agent is a private or a public transportation user. Flexibility reflects the willingness of a private mode user to change its mode. See Table 1 for reference.
All plans of agents were performed in rush hours of the day from 6:30 am to 10:30 am, with a normal distribution to simulate peak times. It was observed a high demand in peak duration between 8- 9:30 am, on both roads. The range of income was 20 to 70 Euro per day. The routes between nodes Origin and Destination had both a length of 19 km.
The free-flow travel time from node Origin to Destination was approximately 25 minutes in the PR mode, whereas for the public transportation it was around 35 minutes plus the waiting time at the bus stop and walking time. The bus frequency service was 10 minutes before the rush hour and 5 minutes during the rush hour.

Table 1: Default value of network and learning parameters

Variable	Value
Number of commuters	N=201
Capacity of links	L=150
Capacity of bus	B=70
Time	6:30am to 10:30am
Range of income	20 to 70 €per day
Simulation period	116 days
Recency (ϕ)	0.3
Exploration (ϵ)	0.6

EXPERIMENTS AND RESULTS

We performed 60 iterations of the model, in which the Roth-Erev learning approach was used to establish the commuter equilibrium between both roads along the departure time interval. During simulation steps, we monitored agents expected and effective utilities, average travel times of public and private transportation, average total travel times, number of commuters on each mode and differences between averages of total travel time in public and private transportation.

Propensity of commuters to select public and private modes were set following a normally random distribution and updated based on recency and exploration learning parameters. Earned scores and two propensities were observed during all days.

In Figure 2, the distribution of all commuters is depicted among all days, where green line and red line show the number of agents on roads with public and private transportation respectively. The number of commuters on different mode of transportation converged upon using learning tools and rewards. On the first day, most people had tendency to use public transportation, which was decreased on the last day of the simulation period.

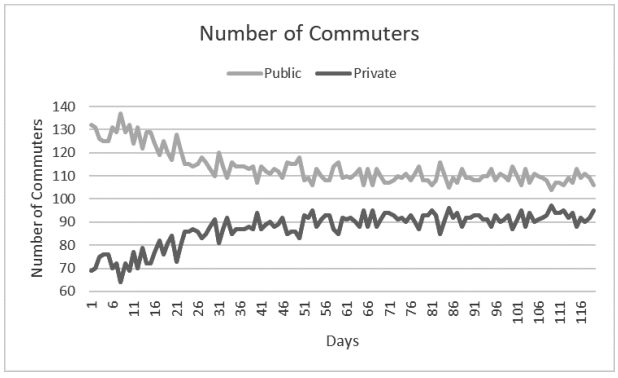


Figure 2: Number of commuters evolution between the 2 modes

Total times of daily trips for both mode of transportation selected by the agents were measured and the differences between these two times for all day long were calculated. In Figure 3, the result is shown for the simulation period. This fluctuation was related to different factors such as traffic on road, departure time and waiting time for public transportation on each day. However, on the final days, the difference time between public and private transportation was less than 10 minutes by reaching equilibrium flow on transportation modes which seemed to be stable.

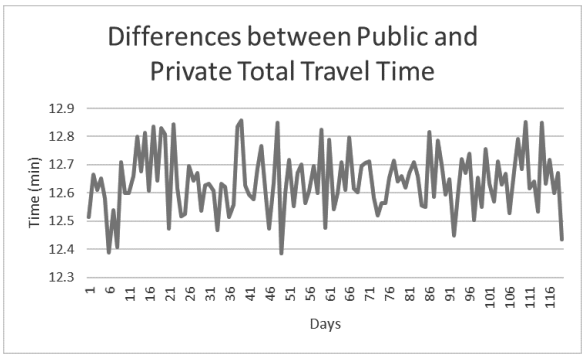


Figure 3: Differences between Public and Private Total Travel Time

Based on rewards and decision-making of departure time and transportation modes, commuters utilities were changed daily. Figure 4 represents daily changes in effective utilities perceived by each of the commuters within the whole period. In this chart, it is shown that both public and private utilities were increased with day-to-day variations.

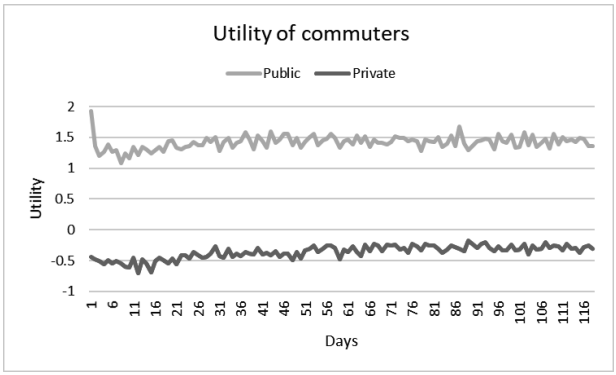


Figure 4: Utility of commuters in both transportation

The number of commuters on different modes, effective utility, average time of public and private modes, and average total time which were observed on the last day along with values for some parameters are described in Table 2. The values there rendered show that the average total travel times of each mode were roughly similar to the average total travel time of both modes, as well as that effective utilities of commuters had a bit difference from the ones they had expected (all margins were consider as 0.1 except $\alpha_{com_{PR}} = 0.65$)

To compare the effect of cost on commuter's mode de-

Table 2: Default Value of run time parameters

Variable	Value
No. of commuters on PT	106
No. of commuters on PT	95
Average total time on PT	34.7 min
Average total time on PT	22.26 min
Average total time of both mode	28.82 min
Average effective utility on PT	1.35
Average effective utility on PR	-0.38
Average expected utility on PR	-0.74
Cost of PT	3 Euros per Day
Cost of PR	8 Euros per Day

cision, three different categories of cost were defined, whose impact on the system performance was simulated and analysed. The cost categories are shown in Table 3.

Table 3: Cost Category

Type	Cost Value of PT (Euro/Day)	Cost Value of PR (Euro/Day)
Cat. 1	3	8
Cat. 2	3	6
Cat. 3	3	3

Figure 5 shows that cost of transportation can change the number of commuters on different modes, as it is expected, and the same costs of public and private transportation result the closer number of commuters on each road. Increasing the cost of private transportation also increases the number of commuters on public transportation.

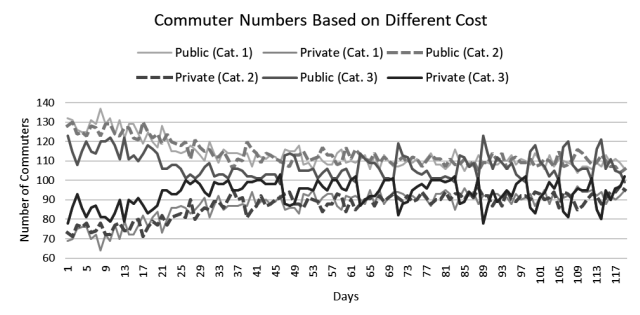


Figure 5: Number of Commuters with different Category of Cost

Total time of both modes in each category was monitored and depicted in Figure 6. The highest difference in cost resulted in lower total time differences. However, in consequence of the same price of public and private transportation, total travel time in both modes is better than a smoother cost difference.

Through decreasing cost of private transportation, the utility of commuters will increase, even though the utilities of commuters on public transportation was approximately the same. The fluctuation of utilities is shown in Figure 7.

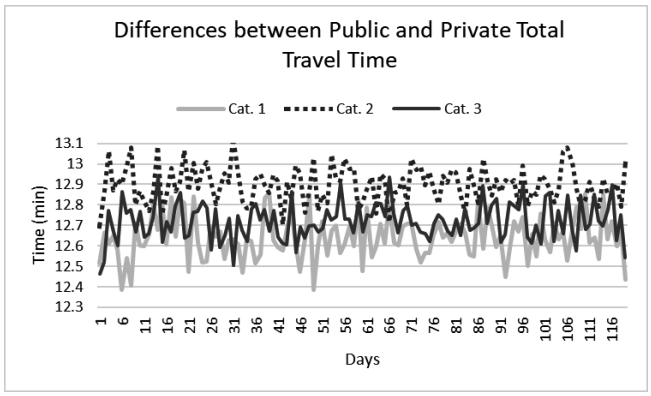


Figure 6: Differences between Public and Private Total Travel Time in different Category

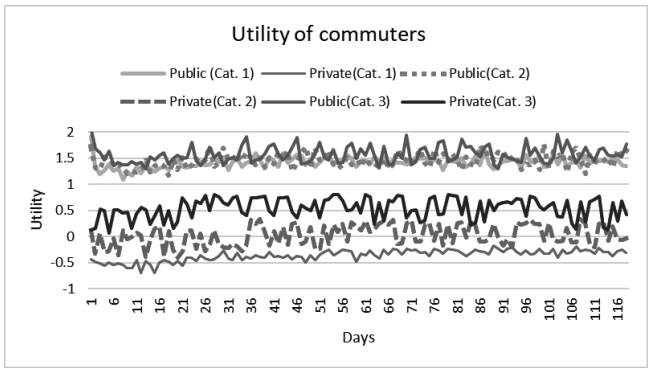


Figure 7: Utility with different Category of Price

RELATED WORK

Kokkinogenis et al. discussed on a social-oriented modelling and simulation framework for Artificial Transportation Systems, which accounts for different social dimensions of the system in the assessment and application of policy procedures. They illustrated how a social agent-based model can be a useful tool to test the appropriateness and efficiency of transportation policies (Kokkinogenis et al. 2014). Nallur et al. introduced the mechanism of algorithm diversity for nudging system to reach distributive justice in a decentralised manner. They use minority game as an exemplar of an artificial transportation network and their result showed how algorithm diversity leads to a fair reward distribution (Nallur et al. 2016). Klein et al. developed a multi-agent simulation model for analysing daily evolution of traffic on roads, in which the behaviour of agents was reinforced by their previous experiences. They considered various network designs, information recommendations, and incentive mechanisms, and evaluated their models based on efficiency, stability, and equity criteria. Their

results suggest that punishments or rewards were useful incentives (Klein et al. 2018). In (Rossetti et al. 2002) is detailed an agent-based demand model following the beliefs-desires-intentions (BDI) architecture, emphasising on decision-making processes such as departure time, route selection, and itinerary deviation. The model was then implemented and tested elsewhere (Rossetti and Liu 2005).
The authors in (Tsekeris and Voβ 2009) underline the potential of agent-based models based on their bottom-up approach with signicant degree of disaggregation, intelligence, autonomy, and ability to capture interactions among individuals. Travel demand emerges from the interactions of agents in the transportation system. The work in (Monteiro et al. 2014) combine the four-step model with an agent-based framework to model such a demand in multi-modal transportation networks. In Zhang et al. (2008), an agent-based model is used to analyse price competition, capacity choice, and willingness to pay for different services on congested networks. Grether and colleagues (Grether et al. 2010) show how multi-agent simulation can be used in road pricing policy evaluation adding an individual income attribute to each agent so that personalised utilities are considered. In (Kickhfer and Nagel 2013), a toll-based policy for air pollution reduction is evaluated, and the long-term user reactions are discussed. (An et al. 2009) proposes an agent-based model that considers individual characteristics and collective group behaviours in the evaluation of bus service performance from the perspective of passengers. In (Li et al. 2010), authors describe an articial urban transit system as an instance of artificial transportation systems (ATS) for public transport. Authors present their model as a set of interactions of different types of agents for simulating transport operations and planning. A similar approach for modelling ATS is proposed in (Soares et al. 2013). The A qualitative evaluation of several traffic simulation frameworks is presented in (Passos et al. 2011), with respect to their ability to model various aspects of modern ATS.

CONCLUSIONS

In this paper, we have proposed the framework for evaluating the effect of reinforcement learning and minority games on the equilibrium of traffic flow on road networks. We suggested to apply agent-based modelling and simulation as a platform to implement our framework. To illustrate such an approach, a simple network consisting of two different modes of transport (PT and PR) was considered, and a population of commuters with memory of previous travel experiences were generated. They performed their daily plan in morning high-demand hours and their activities iterated for sixty days. Their experiences, expected and effective utilities, expected and effective travel times and rewards were observed and analysed.

Regarding the results, commuters learned to predict total travel time in both modes, and their exceptions were similar to obtained total travel time on each mode. By balancing the number of commuters on each type of transportation, they gained higher utilities rather than on the first days. From the illustrative example, the hypothesis of the study, which was to use reinforcement learning and minority game to reach equilibrium flow, was reached and it is concluded that equilibrium flow can follow higher utilities and more precise time prediction of daily trips.
As for future work, we will consider a realistic large-scale network and demand, different types of incentives and roads with a wider combination of transportation modes so as to better study and analyse commuter behaviour, as well as the performance of the transportation system as a whole. With such improvements, we are confident that our framework can be proper and accurate to enhance commuters experience and also improve the performance of the road transportation system.

ACKNOWLEDGEMENT

This work partially results from the C-ROADS Portugal project, funded by the European Union's Connecting Europe Facility programme, under grant agreement No INEA/CEF/TRAN/M2016/13613245, Action No 2016-PT-TM-0259-S.

REFERENCES

An J.; Liu H.; Yang X.; and Teng J., 2009. *Modeling single-route bus service with information environment: A multi-agent based approach.* In *Intelligent Transportation Systems, 2009. ITSC'09. 12th International IEEE Conference on.* IEEE, 1–6.

Ben-Elia E. and Ettema D., 2011. *Rewarding rush-hour avoidance: A study of commuters travel behavior. Transportation Research Part A: Policy and Practice*, 45, no. 7, 567–582.

Bliemer M. and van Amelsfort D.H., 2010. *Rewarding instead of charging road users: a model case study investigating effects on traffic conditions. European Transport\ Trasporti Europei*, , no. 44, 23–40.

Bouman P.; Kroon L.; Vervest P.; and Maróti G., 2016. *Capacity, information and minority games in public transport. Transportation Research Part C: Emerging Technologies*, 70, 157–170.

Challet D. and Zhang Y.C., 1997. *Emergence of cooperation and organization in an evolutionary game. Physica A: Statistical Mechanics and its Applications*, 246, no. 3-4, 407–418.

Chiu Y.C.; Bottom J.; Mahut M.; Paz A.; Balakrishna R.; Waller T.; and Hicks J., 2010. *A primer for dy-*

namic traffic assignment. Transportation Research Board, 2–3.

Chmura T.; Pitz T.; and Schreckenberg M., 2005. *Minority game-experiments and simulations.* In *Traffic and Granular Flow03*, Springer. 305–315.

de Dios Ortuzar J. and Willumsen L.G., 2011. *Modelling transport.* John Wiley & Sons.

Ebrahimi M.; Monsefi R.; Ildarabadi S.; Akbarzadeh M.; and Habini M., 2007. *License plate location based on multi agent systems.* In *2007 11th International Conference on Intelligent Engineering Systems.* IEEE, 155–159.

Golubev K.; Zagarskikh A.; and Karsakov A., 2018. *A framework for a multi-agent traffic simulation using combined behavioural models. Procedia Computer Science*, 136, 443–452.

Grether D.; Chen Y.; Rieser M.; and Nagel K., 2009. *Effects of a simple mode choice model in a large-scale agent-based transport simulation.* In *Complexity and Spatial Networks*, Springer. 167–186.

Grether D.; Kickhöfer B.; and Nagel K., 2010. *Policy evaluation in multiagent transport simulations. Transportation Research Record*, 2175, no. 1, 10–18.

Kaplan A.M. and Haenlein M., 2010. *Users of the world, unite! The challenges and opportunities of Social Media. Business horizons*, 53, no. 1, 59–68.

Kickhfer B. and Nagel K., 2013. *Towards High-Resolution First-Best Air Pollution Tolls - An Evaluation of Regulatory Policies and a Discussion on Long-Term User Reactions. Networks and Spatial Economics*, 1–24.

Klein I.; Levy N.; and Ben-Elia E., 2018. *An agent-based model of the emergence of cooperation and a fair and stable system optimum using ATIS on a simple road network. Transportation research part C: emerging technologies*, 86, 183–201.

Kokkinogenis Z.; Monteiro N.; Rossetti R.J.F.; Bazzan A.L.C.; and Campos P., 2014. *Policy and incentive designs evaluation: A social-oriented framework for Artificial Transportation Systems.* In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC).* IEEE, 151–156.

Li L.; Zhang H.; Wang X.; Lu W.; and Mu Z., 2010. *Urban Transit Coordination Using an Artificial Transportation System*, 1–10.

Monteiro N.; Rossetti R.; Campos P.; and Kokkinogenis Z., 2014. *A framework for a multimodal transportation network: an agent-based model approach. Transportation Research Procedia*, 4, 213–227.

Nallur V.; O'Toole E.; Cardozo N.; and Clarke S., 2016. *Algorithm diversity: A mechanism for distributive justice in a socio-technical MAS.* In *Proceedings of the 2016 international conference on autonomous agents & multiagent systems.* International Foundation for Autonomous Agents and Multiagent Systems, 420–428.

Passos L.S.; Rossetti R.J.F.; and Kokkinogenis Z., 2011. *Towards the next-generation traffic simulation tools: a first appraisal.* In *6th Iberian Conference on Information Systems and Technologies (CISTI 2011).* IEEE, 1–6.

Rossetti R.J.F.; Bordini R.H.; Bazzan A.L.C.; Bampi S.; Liu R.; and Van Vliet D., 2002. *Using BDI agents to improve driver modelling in a commuter scenario. Transportation Research Part C: Emerging Technologies*, 10, no. 5–6, 373–398.

Rossetti R.J.F. and Liu R., 2005. *An Agent-Based Approach to Assess Drivers' Interaction with Pre-Trip Information Systems. Journal of Intelligent Transportation Systems*, 9, no. 1, 1–10.

Roth A.E. and Erev I., 1995. *Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. Games and economic behavior*, 8, no. 1, 164–212.

Rouwendal J. and Verhoef E.T., 2006. *Basic economic principles of road pricing: From theory to applications. Transport policy*, 13, no. 2, 106–114.

Soares G.; Kokkinogenis Z.; Macedo J.L.; and Rossetti R.J.F., 2013. *Agent-based Traffic Simulation using SUMO and JADE: an integrated platform for Artificial Transportation Systems.* In *Simulation of Urban MObility User Conference.* Springer, 44–61.

Takama T. and Preston J., 2008. *Forecasting the effects of road user charge by stochastic agent-based modelling. Transportation Research Part A: Policy and Practice*, 42, no. 4, 738–749.

Tlig M. and Bhouri N., 2011. *A multi-agent system for urban traffic and buses regularity control. Procedia-Social and Behavioral Sciences*, 20, 896–905.

Tsekeris T. and Voβ S., 2009. *Design and evaluation of road pricing: state-of-the-art and methodological advances. Netnomics*, 10, no. 1, 5–52.

Wilensky U., 2014. *NetLogo: Center for connected learning and computer-based modeling, Northwestern University. Evanston, IL, 1999.*

Zhang L.; Levinson D.M.; and Zhu S., 2008. *Agent-based model of price competition, capacity choice, and product differentiation on congested networks. Journal of Transport Economics and Policy (JTEP)*, 42, no. 3, 435–461.

USING SIMULATION GAMES FOR TRAFFIC MODEL CALIBRATION

Gonçalo Leão, João Ferreira, Pedro Amaro, Rosaldo J. F. Rossetti
Artificial Intelligence and Computer Science Lab, Department of Informatics Engineering
Faculty of Engineering, University of Porto (FEUP)
Rua Dr. Roberto Frias, S/N – 4200-465 Porto, Portugal
E-mail: {up201406036, up201405163, up201405210, rossetti}@fe.up.pt

KEYWORDS
Calibration, Krauß model, Traffic simulation, Vehicle driving

ABSTRACT

Microscopic simulation requires accurate car-following models so that they can properly emulate real-world traffic. In order to define these models, calibration procedures can be used. The main problem with reliable calibration methods is their high cost, either in terms of the time they need to produce a model or due to high resource requirements. In this paper, we examine a method based on virtual driving simulation to calibrate the Krauß car-following model by coupling the Unity 3D game engine with SUMO. In addition, we present a means based on the fundamental diagrams of traffic flow for validating the instances of the model obtained from the calibration. The results show that our method is capable of producing instances with parameters close to those found in the literature. We conclude that this method is a promising, cost-efficient calibration technique for the Krauß model. Further investigation will be required to define a more general approach to calibrate a broader range of car-following models and to improve their accuracy.

INTRODUCTION

In a microscopic traffic simulator, there is an explicit representation for the vehicles. Thus, these simulators allow the study of the behaviors and interactions between individual vehicles.
These simulators have numerous real-life applications. From a traffic engineering perspective, microscopic simulation can be used to understand complex phenomena (such as how vehicles behave in a crossway), where a mathematical or analytical treatment is either inadequate or infeasible (Mathew 2012). Using a traffic flow model, simulations can also be used in both traffic planning and prediction in order to explore different solutions for a given problem (for instance, traffic light scheduling) and to forecast traffic on a network, respectively. Lastly, simulators can be used for the real-time monitoring of the actual traffic within a network, in order to detect problems such as congestions.
These simulators take into account two perpendicular axis. The longitudinal axis considers the vehicles' position and speed in the same direction as the lane they are traveling on. The lateral axis considers the movements of vehicles when they wish to change lane.
There are many models available to simulate the motion of vehicles. Many of them take into account the human aspect of their drivers, such as their reaction time. Car-following models describe how cars are spaced between each other and how drivers react to changes caused by the vehicle ahead of them. Notable examples of this type of models are the Krauß', Gipps' and Wiedemann models. Lane changing models determine how vehicles choose the lane they will travel on in roads with multiple parallel lanes and the speed adjustments required to change lanes. Intersection models handle the way vehicles behave when they reach a junction namely in terms of right-of-way rules, gap acceptance and avoiding deadlocks.
Simulation of Urban MObility (SUMO) is an open-source, microscopic simulator that is well suited to represent a traffic road network of the size of a city (Behrisch et al. 2011).
Currently, for the simulation of the longitudinal movement of vehicles in a single lane, the SUMO simulator uses a modified version of the Krauß' car-following model, so that a vehicle sets its speed depending on its desired speed, the speed of the vehicle in front of it, the distance between both cars and the driver's reaction time (Krajzewicz et al. 2002). This model is stochastic, as it subtracts a random human error from the desired speed. One important constraint of this model is that it is collision-free.
In order for a simulator to be representative of the real world, the models it is using must be properly calibrated. In other words, the model's parameters must be properly defined so that the simulator's output data is adequately close to reality. This is a major challenge during the implementation stage of any model and requires the gathering of data (Mathew 2012).
One of the traditional methods used to collect data is on the field, by using chronometers and other measuring devices to measure the speed and distance between vehicles traversing a road.
A more modern approach for traffic surveillance are Automated Vehicle Counting (AVC) systems. These real-time systems use image processing algorithms to detect passing vehicles (Wang et al. 2007).
Another alternative is using physical driving simulators, where the driver is placed on a cockpit controlled by software models that create an environment that emulates real driving. One notable example is the Toyota Driving Simulator, built at Toyota's Higashifuji Technical Center, whose goal is to conduct driving tests that are too

dangerous to be performed in the real world, such as studying the effects of fatigue or illness on the driver's performance (Slob 2008).

The problem with all these methods is that they are highly resource demanding. For instance, a large amount of time and human resources are required in order to collect a substantial volume of data with the traditional approach. In addition, AVC systems might face accuracy problems, including struggling to detect vehicles at night and to count certain types of vehicles, such as motorcycles (Wang et al. 2007). Many physical driving simulators do not emulate the physical motion (namely the pitch and roll movements) experienced by real-life drivers, which may lead to motion sickness (Cheng et al. 2006).

Therefore, having more cost-efficient means of acquiring data will allow the proper calibration of more traffic models using less time and other resources, as well as reducing the cost of adapting existing models.

Our hypothesis is that simulation games can be used to extract the data needed for traffic models calibration, as an alternative to the methods previously described. These games have the potential of being a more cost-efficient means of collecting data (Gonçalves and Rossetti 2013).

In order to demonstrate this hypothesis, we developed a methodology to calibrate the modified Krauß car-following model used by SUMO, that encompasses connecting the traffic simulator with a 3D cockpit driving simulator, and collecting data from drivers in the simulated environment in order to find appropriate values for the model's parameters.

Section II consists of a review of the available literature related to the calibration of car-following models. The background presents an overview of the body of knowledge and history related to the study of car-following models, whereas the related work and gap analysis subsections establish a parallel with similar works that aim to calibrate and validate this type of models. Section III formalizes the Krauß car-following model and provides a brief introduction to the fundamental diagrams of traffic flow. Section IV details our methodology to calibrate Krauß' model using a game engine. Section V explains how we evaluated our model's performance and reports the results of this analysis. Finally, section VI concludes with some discussion and further developments.

LITERATURE REVIEW

Background

Traffic problems began attracting the attention of the scientific community with the rise of the automobile, when owning a vehicle became affordable to the general population and traffic jams became an issue. Due to the economic and social impact caused by traffic congestion, it became increasingly important to find means of tackling and mitigating this problem.

The earliest contributions to traffic theory date back to the early 50's. On the one hand, Reuschel (Reuschel 1950) and Pipes (Pipes 1953) described the movement of two vehicles driving close to one another in a single lane. Their studies led to a a microscopic model of traffic. On the other hand, Lighthill and Whitham (Whitham and Lighthill 1955) proposed a "macroscopic" traffic model that established a parallel with fluid mechanics. Their model provided a good description of several basic traffic phenomena such as the propagation of "shock waves" (Gazis 2002).

The first car-following model was put forward in the late 50's by three researchers from the General Motors Research Lab in Detroit (Chandler et al. 1958). This work would eventually lead to one of the first and perhaps most well-known car-following model, the Gazis-Herman-Rothery (GHR) model (Gazis et al. 1961). It is based in the hypothesis that a following vehicle's acceleration is proportional to the relative speed between the leading and following vehicles.

During the 60's and 70's, several researchers proposed various small modifications to this model's formulas and conducted experiments in order to find the optimal values for the model's parameters. However, since the late 70's, the GHR model has had less attention, possibly due to the large number of contradictory findings for the correct values of certain parameters (Brackstone and McDonald 1999).

Another major family of car-following models are the collision avoidance models, that were first introduced by Kometani and Sasaki in 1959 (Kometani and Sasaki 1959), when they proposed a model that adds an additional 'safety' reaction time which is a sufficient condition to avoid a collision under all circumstances. One of the most attractive properties of this model is that its parameters correspond to self-evident characteristics of vehicles, and thus most of them can be assigned values without the use of elaborated calibration procedures (Gipps 1981).

Therefore, due to its simplicity, various prominent traffic simulators use modified versions of the Gipps model. In particular, SUMO uses Krauß' model which is essentially a stochastic variation of the Gipps model. We will present this model in further detail in the following section (Krauß 1998).

Nowadays, new car-following models are being formulated based on modern trends. On the one hand, researchers have created models that consider the latest advancements in technology, which bring both new capabilities for vehicles, such as inter-vehicle communication (Tang et al. 2014), but also new threats, namely cyberattacks (Wang et al. 2017b). On the other hand, an increasingly large number of models are taking into account the environmental factors related to traffic, such as fuel consumption and exhaust emissions, as important performance metrics since traffic pollution has become one serious problem (Wang et al. 2017a). Moreover, the increasing popularity of electric vehicles has lead to models that factor their driving range (Tang et al. 2015). Finally, modern car-following models also taken into account multiple modes of transport, namely public transport (Shen et al. 2017).

Related work

A significant amount of research has been conducted on the topic of calibrating the parameters of car-following models. To perform this task, data related to the drivers' and vehicles' properties must be collected. On the one hand, there are cost efficient ways to collect the data for calibration, such as capturing trajectory data with a GPS and a cellphone, as detailed by Zeyu (Zeyu et al. 2016), or using a car equipped with a LIDAR (Light Detection and Ranging) rangefinder, which measures the distance to the various objects around it, as shown by Neto (Vasconcelos et al. 2014). On the other hand, Hinsbergen has shown that the trajectory of many vehicles can be captured using an helicopter and a camera (Van Hinsbergen et al. 2015), which is another approach to capture calibration data, albeit more expensive, comparatively.

Afterwards, the captured data must be processed, and then applied to the models as parameters. Models with a great number of parameters face the problem of overfitting, since the model's parameters are prone to become biased by the datasets used to calibrate them. To overcome this issue, Hinsbergen uses Bayesian techniques to calibrate more complex models. These techniques balance a given model's complexity with how well it fits to a dataset (Van Hinsbergen et al. 2015).

One key aspect of most of the recent work on calibration is that they do not focus on calibrating the Krauß car-following model, and prefer other models, such as the GM (Zeyu et al. 2016), CHM, Helly, OVM, IDM, Generalized Helly-3-1, Lenz-2, HDM (Van Hinsbergen et al. 2015) and Gipps' car-following models (Vasconcelos et al. 2014).

Older work on calibration and validation includes Ranjitkar (Ranjitkar et al. 2005) and Kesting (Kesting and Treiber 2008). Ranjitkar benchmarks various car-following models, including the Krauß car-following model. Kesting uses publicly available trajectory datasets to calibrate the IDM and VDM car-following models. Then, he validates them by comparing the empirical data and the simulated data with the known fundamental diagrams.

Another objective of our work is to show a simulated traffic environment which the user can interact with and from where we can collect data. Gonçalves described a methodology that involves using a driving simulator in Unity 3D, and explained how the data was collected and processed to cluster several kinds of drivers (Gonçalves and Rossetti 2013). The main difference with our work is that we will not cluster drivers, and will additionally use the data collected to calibrate and validate the Krauß car-following model in SUMO.

There are many other, less recent, papers on the subject of comparing the behavior of test subjects in a real driving environment to a simulated driving environment, although there is no calibration of traffic models using the collected data. Mullen explains that most driving simulators are able to approximate, but not exactly replicate, on-road driving behavior (Mullen et al. 2011). For instance, the distance to the closest wall in a tunnel tends to be lower in simulators, whereas the vehicle's speed tends to be higher. However, similar constraints applied to both the real driving situation and the simulated one, such as removing the speedometer, showed the same result in both situations: an increase of speed. Thus, while there is no direct correspondence between the behavior data in the driving simulator and in the real situation, the correlation between effects of different variations in the driving situation is satisfactory (Törnros 1998).

Gap Analysis

Table 1: Gap Analysis from Related Work

	Calibrates a model	Calibrates Krauß model	Uses car simulator	Validates the calibration
Neto	x			
Hinsbergen	x			
Zeyu	x			
Gonçalves			x	
Ranjitkar	x	x		
Kesting	x			x
Our approach	x	x	x	x

To the best of our knowledge, there has been little research regarding the calibration and the validation of instances of the Krauß model, despite how frequent it is used in traffic simulators. We speculate that this is due to how close this model is to the well-known models, namely the Gipps model, which has been extensively used in various works.

PROBLEM FORMALIZATION

Krauß car-following model

The purpose of this project is to calibrate the Krauß car-following model. Although a modified version of this model is used as default within SUMO, the input parameters are the same as the original Krauß model. The input parameters are defined on Table 2. The table includes estimates for the typical values of the parameters in SI units, as given by the Gipps' car-following model (Krauß 1998), which we can use for an initial calibration of the Krauß' model.

Table 2: Gipps Model Parameters

Variable	Name	Estimate	Restrictions
a	Acceleration	$0.8ms^{-2}$	$a > 0$
b	Deceleration	$4.5ms^{-2}$	$b > 0$
τ	Reaction Time	$1s$	$\tau \geq 0$
η	Driver deviation		$\eta \geq 0$
v_{max}	Maximum speed	$36ms^{-1}$	$vmax \geq 0$

The most simple parameters for any car-following model are the acceleration a and deceleration b, which must abide by inequality (1).

$$-b \leq \frac{dv}{dt} \leq a \tag{1}$$

For the more common of the car-following models, it is assumed that there is a single lane, with a leading

car, which is in the front, being followed by a following car, which attempts to keep a minimum distance from the leading car. This distance is related to τ, the reaction time, which is intended to model the time a human driver takes to react to the leading car braking. The speed of the following car should, given enough time, match the speed of the leading car.
To ensure that vehicles can move without colliding with each other, both models use the gap between vehicles g and the desired gap between vehicles g_{des}, both in meters. These variables can be calculated by using the position of the leading car x_l, the position of the following car x_f, the length of the leading vehicle l_l, and its speed v_l, using the equations (2) and (3).

$$g = x_l - x_f - l_l \tag{2}$$

$$g_{des} = \tau v_l \tag{3}$$

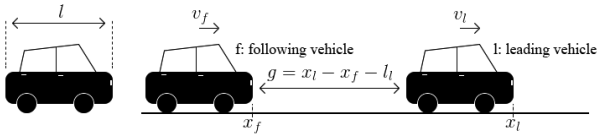


Figure 1: Illustration of the Gap between two Consecutive Vehicles

It is important to note that the width of the car is not considered in the car-following models, and that the model remains consistent even if the vehicles have different lengths.
The Krauß car-following model then uses the gaps from figure 1, along with the time scale τb, which can be calculated using equation (4), where $\bar{v}$ is the average speed of both vehicles, and b is the deceleration of the cars, to define the safe speed, which is the speed at which the following vehicle can safely brake when the leading vehicle comes to an abrupt stop.

$$\tau_b = \bar{v}/b \tag{4}$$

$$v_{safe}(t) = v_l(t) + \frac{g(t) - g_{des}(t)}{\tau_b + \tau} \tag{5}$$

From definition of v_{safe} from equation (5), we can define the desired speed, v_{des}, which is the minimum between the maximum speed of the following car v_{max}, the current speed of the following car, plus a small contribution due to acceleration $a(v)\Delta t$, and the safe speed, as seen in equation (6).

$$v_{des}(t) = min[v_{max}, v(t) + a(v)\Delta t, v_{safe}(t)] \tag{6}$$

Finally, assuming we have a given vehicle speed v and position x at an instant t, we can define these output parameters of the model at the instant $t + \Delta t$ as seen in equations (7) and (8).

$$v(t + \Delta t) = max[0, v_{des}(t) - \eta] \tag{7}$$

$$x(t + \Delta t) = x(t) + v\Delta t \tag{8}$$

In the equation above, η represents a random deviation from optimal driving. It is used to represent the fact that humans are not perfect drivers, and it adds a stochastic factor to the Krauß model.
In the particular case where a given vehicle has no vehicle at its front, $v_{safe}(t)$ is evaluated to $+\infty$. Therefore, the driver's desired speed, $v_{des}(t)$, will eventually correspond to the maximum speed the vehicle can drive at.
The fundamental property of Krauß' car-following model, including the modified one in SUMO, is that it is collision free, as the equations above guarantee that drivers always stay within safe velocities that enable them to stop their vehicles with enough time so that they don't collide with the leading vehicle.

Fundamental relations of traffic flow

For the final phase of our project, where we will validate our model, we will resort to a quantitative analysis that uses the fundamental relation of traffic flow. This relation is given by formula (9).

$$q = kv \tag{9}$$

where q is the traffic flow (measured in vehicles/h), k is the traffic density (measured in vehicles/km) and v is the mean speed (measured in km/h) (Pipes 1966).
All three of these measures vary with time and location. These measures have some important properties:

- If the density k is zero, then the flow q will also be zero since there are no vehicles on the road.
- When the number of vehicles surpasses a given threshold, the road will reach a state of maximum density, known as the "jam density" k_{max}, where the vehicles can't move. In this scenario, both the flow q and mean speed v will be zero.
- There is a value for the density k (between the state of zero and maximum density) such that the flow q is maximum, known as the "critical density" k_c. When this value is reached, the mean speed reaches a value known as the "maximum flow speed" v_c.
- As the number of vehicles increases, both the traffic density k and the traffic flow q will increase (Hoogendoorn 2016), until the critical density k_c is reached. After this point, an increase in the number of vehicles (and thus to the density) will decrease the traffic flow.

There are various models that give an explicit formula for the mean speed as a function for the density ($v = f(k)$).
One of the simplest and earliest models is due to Greenshields, who suggests that the speed-density relation is linear and can be given by the formula (10).

$$v(k) = v_{max} - \frac{kv_{max}}{k_{max}} \tag{10}$$

where v_{max} is the maximum speed the vehicle can travel in the network (not to be confused with v_c) (Greenshields et al. 1935).

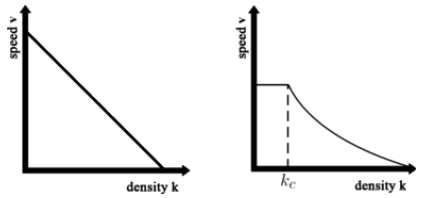


Figure 2: Speed-Density Fundamental Diagrams according to Greenshields (left) and Greenberg (right)

Greenshields' model is univariate since the uncongested and congested traffic regimes are described by the same formula (Knoop and Daamen 2017).

One of the problems of Greenshields' model is that it is rather unrealistic for small traffic densities since the mean speed does not decrease as strongly as described by the model (Hoogendoorn 2016). One possible solution for this issue is to use two-variate models, which are more flexible.

For the validation of our instance of the Krauß model, we will also use a modified, two-variate model based on Greenberg's model, which states that the speed-density relation is logarithmic (Greenberg 1959). Formula (11) is our modified version of Greenberg's model (Hoogendoorn 2016).

$$v(k) = \begin{cases} v_{max} & k \leq k_c \\ v_c \ln \frac{k_{max}}{k_c} & k > k_c \end{cases} \quad (11)$$

where $\ln x$ is the natural logarithm of x.

In formula (11), v_c can be derived if we set k to be equal to k_c. In this case, we obtain formula (12).

$$v_c = \frac{v_{max}}{\ln \frac{k_{max}}{k_c}} \quad (12)$$

METHODOLOGICAL APPROACH

Our approach can be divided into two main phases. First of all, we developed a virtual driving simulator by integrating a 3D cockpit built in Unity with SUMO. Secondly, using this newly developed simulator, we collected the data required to calibrate the Krauß model's parameters. This calibration produced a new instance of the model.

Data collection

We designed a track in SUMO as well as in the Unity 3D driving simulator. Since our goal is to calibrate a car-following model (and not an intersection model, for instance), for the sake of simplicity, this track is a straight path with two lanes.

During the experiments, there were two vehicles present in this track: the avatar controlled by the driving simulator (following vehicle) and a vehicle in SUMO, controlled by keyboard input (leading vehicle).

In order to collect all the data required, 16 subjects used the game engine to drive the vehicle associated with the avatar until they accomplished a specific goal: reaching the end of the track.

The experiment was divided into two phases. In the first phase, both vehicles had to remain on the right lane and, at certain points throughout the track, we made the leading vehicle abruptly brake until it completely stopped, in order to measure the driver's reaction time and maximum deceleration. In the second phase, after having stopped the leading vehicle at least two times, the test subject was instructed to overtake the vehicle controlled by SUMO by using the left lane. The goal of this phase was to evaluate the driver's maximum speed and acceleration.

There was a warm-up period, to allow the subjects to get acquainted with the driving simulator, along with the leading vehicle's motion.

Exceptionally, in the cases where an unexpected event happened, such as the avatar crashing against the leading vehicle or overturning, the data collected for that subject was discarded, as we assume the data gathered was not relevant to the Krauß model.

After each simulation ended, we computed the test subject's acceleration a and deceleration b values as the absolute value of the maximum and minimum amount of acceleration measured in each frame, respectively. For the maximum speed v_{max}, we used the maximum value recorded for the speed among all frames. For the reaction time τ, we measured the average time it took for the driver to begin braking when the leading vehicle started braking. In other words, for each driver, the reaction time parameter was measured as seen in equation (13).

$$\tau = \frac{\sum_{n=1}^{N} t_f(n) - t_l(n)}{N} \quad (13)$$

where N is the number of times the leading vehicle brakes, $t_l(n)$ is the instant (in s) when the leading vehicle brakes for the n-th time, and $t_f(n)$ is the instant (in s) when the user begins to brake as a consequence of the leading vehicle braking for the n-th time. $t_f(n)$ was computed as an instant when the user abruptly applied more pressure to the brakes in order to react to the leading vehicle's sudden behavior.

For the sake of simplicity, we assumed that there is no driver deviation η, which will be set to zero in our model's instance.

As future work, it would be interesting to cluster the values of the parameters according to different simulation scenarios, such as driving in circles or in a road with a large amount of curves.

In order to produce an instance of the Krauß model, we set the acceleration, deceleration, reaction time and maximum speed parameters to be the average of the respective variables amongst all drivers.

RESULTS AND ANALYSIS

We tested both qualitatively and quantitatively our instance of the Krauß model to assess its validity. In order to perform this validation, we used a second instance of the Krauß model that was parametrized using the values from Table 2. Table 3 summarizes the values for both instances of the Krauß model we tested.

Table 3: Values for both Instances of the Krauß Model

	a	b	v_{max}	τ	η
Our instance	7.027	10.77	36.23	0.7226	0.0
Krauß' instance	0.8	4.5	36.0	1.0	0.0

Quantitative analysis

We ran simulations in a circular track with both instances of the Krauß model and then plotted values for the fundamental diagram that relates the traffic density k to the mean speed v.

In particular, we generated different amounts of vehicles in the network, in order to obtain different values for the density. For each scenario, we computed the average of the vehicles' speed v after a certain amount of time (in order for the system to stabilize).

The results are summarized in figures 3 and 4.

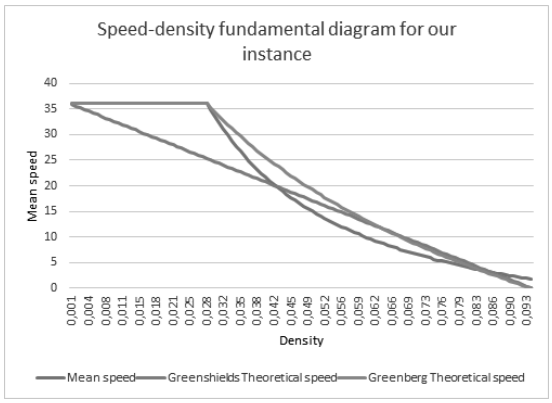


Figure 3: Fundamental Diagram for our Instance

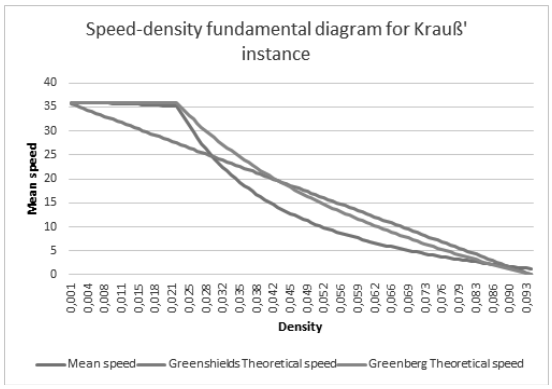


Figure 4: Fundamental Diagram for the Instance from Table 2

In order to evaluate the performance of both instances of the Krauß model in a quantitative manner, we measured the deviation of the plotted values with respect to the theoretical fundamental curves given by the Greenshields and Greenberg models (that were presented in section III). It should be noted that our intention is not to minimize these deviations (since both models are only approximations of the speed-density relation in real-life traffic networks). Instead, ideally, the difference of the deviations between both instances will be relatively small, which may serve as an indicator that our calibration method is able to produce instances with coherent values for the parameters of the Krauß model. We measured this deviation using the chi-square goodness of fit test between the actual speed-density curve and the theoretical curve given by each model, for both instances. In order to use this measure, we need to split the values for the mean speed into bins, where a given bin stores the number of values for the mean speed that are in a certain range, and with the sum of the frequency of all bins being equal to the total number of plots in our experiment. The width of each bin was computed using Scott's rule (Scott 1979) which is presenting in equation (14).

$$width_{bin} = \frac{3.5\hat{\sigma}}{n^{1/3}} \tag{14}$$

where $\hat{\sigma}$ is the standard deviation of the series and n is the number of values.

The chi-square value χ^2 can be computed using formula (15).

$$\chi^2 = \sum_{i=1}^{B} \frac{(o_i - e_i)^2}{e_i} \tag{15}$$

where B is the total number of bins, and o_i and e_i are the number of values for the i-th bin of the observed and expected values, respectively.

Table 4: Chi-Square Values of both Instances Relative to Speed-Density Models

	Deviation from Greenshields	Deviation from Greenberg
Our instance	41.592	4.126
Krauß' instance	51.304	8.342

As it can be seen in the figures 3 and 4, and in Table 4, the Greenberg model fits the speed-density curves for both instances much better than the Greenshields model (since the chi-square values are significantly smaller). This is mainly due to the fact that the actual mean speed only begins to decrease significantly once the number of vehicles surpasses a given threshold. We speculate that this threshold corresponds to when the leading vehicle starts to decrease its speed due to its close proximity to the last vehicle in the queue.
The chi-square values for both instances have the same order of magnitude in both models, which suggests that our instance behaves similarly to the instance with the parameter values from Table 2.
As future work, it would be interesting to develop a more rigorous alternative to analyze these deviations either by using the chi-square or other relevant curve fitness metrics.

Qualitative analysis

To evaluate the performance of our instances in a qualitative manner, we generated traffic shock waves in SUMO by making the leading vehicle brake in a given sequence of ten vehicles, in a straight road. Then, we built the graphs of the position, speed and acceleration of each vehicle with respect to time for both instances of the Krauß' model.
In the figures 5 to 10, instant $t = 0$ corresponds to when the leading vehicle started braking.

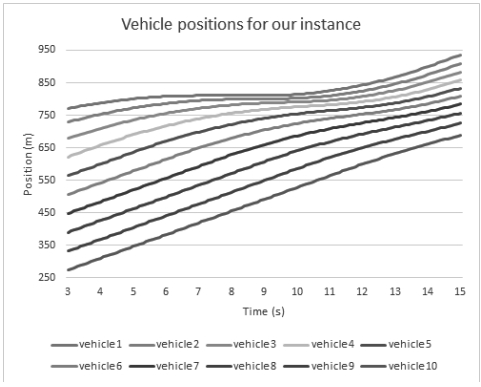


Figure 5: Position curves for our instance

The results clearly show the delays between consecutive vehicles to start braking when the leading vehicle brakes. It is also interesting to notice how the first vehicles in

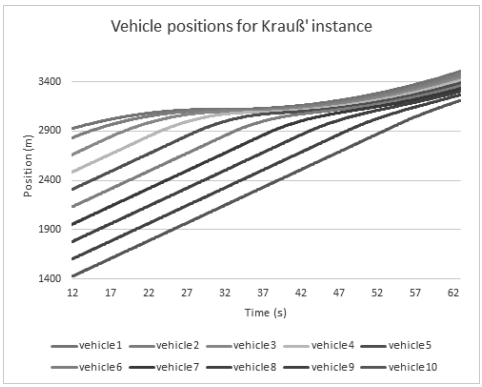


Figure 6: Position Curves for the Instance from Table 2

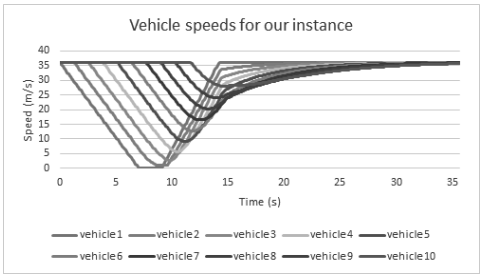


Figure 7: Speed Curves for our Instance

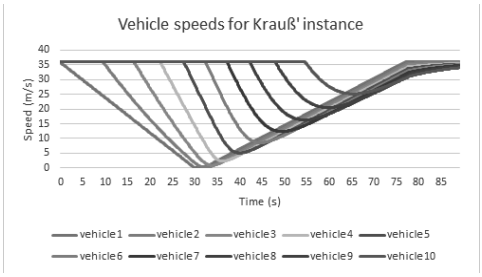


Figure 8: Speed Curves for the Instance from Table 2

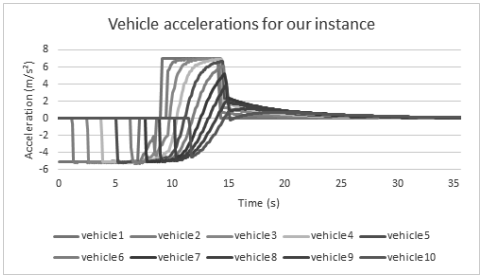


Figure 9: Acceleration Curves for our Instance

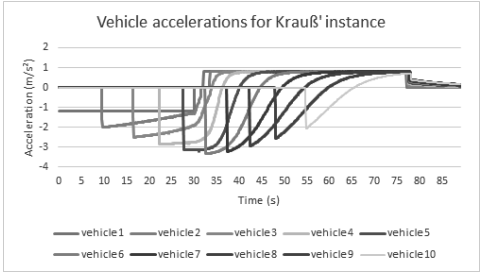


Figure 10: Acceleration Curves for the Instance from Table 2

the queue brake with slightly more intensity than the ones in front of them.

The main difference between both instances is that the vehicles using Krauß' instance take a longer time to brake and to reach their maximum speed. This is coherent with the fact that the values for the acceleration a and deceleration b are both smaller on Krauß' instance than in our instance.

Apart from this difference, both instances appear to have the same behavior which is an indicator that the instance we produced is well calibrated.

CONCLUSIONS AND FUTURE RESEARCH

We have developed a means of calibrating a car-following model using a game engine.

To begin with, we integrated a well-known traffic simulator, SUMO, with a game engine, Unity 3D, which is capable of simulating and rendering in a three dimensional space the vehicle controlled by the user as well as other vehicles and roads.

Afterwards, we used this newly developed driving simulator to collect data to be used for Krauß' model calibration. Most of the parameters were rather trivial to calibrate, with the exception of the reaction time, which required making the leading vehicle abruptly stop at several moments of the simulation and measuring how quickly the test subject was able to brake.

Our approach presents several weaknesses. Firstly, we assumed that the driver deviation η was 0. This could lead to models that are not able to simulate real traffic with accuracy. Future work can include finding an accurate way of calibrating this parameter.

In addition, we can question the simple method we used for computing the average speed and acceleration with respect to all drivers in order to produce values for the model's parameters. This approach does not distinguish the different types of drivers. As future work, we could cluster data we collect in order to extract stereotypes that distinguish different types of behavior, such as nervous or more relaxed driving.

Finally, the procedures used to validate Krauß' model can be improved in further iterations, in order to increase our confidence that our calibration technique produces coherent, well-behaved instances.

Regarding the strengths of our approach, we can mention that our driving simulator has a low cost comparatively to other methods, such as using an AVC, attaching measuring devices to vehicles and driving them in real-life roads or using physical driving simulators.

Moreover, our 3D simulation game is able to render a realistic virtual cockpit, which enables it to have a look and feel closer to a real vehicle than other virtual driving simulators.

Lastly, it has the potential of being extensible to other car-following models, by adapting the type of data collected during the simulation.

REFERENCES

Behrisch M.; Bieker L.; Erdmann J.; and Krajzewicz D., 2011. *SUMO–simulation of urban mobility: an overview.* In *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation.* ThinkMind.

Brackstone M. and McDonald M., 1999. *Car-following: a historical review. Transportation Research Part F: Traffic Psychology and Behaviour*, 2, no. 4, 181–196.

Chandler R.E.; Herman R.; and Montroll E.W., 1958. *Traffic dynamics: studies in car following. Operations research*, 6, no. 2, 165–184.

Cheng M.; Irawan P.; Kwak H.E.; and Putra P., 2006. *Motion Base for Driving Simulator.*

Gazis D.C., 2002. *The origins of traffic theory. Operations Research*, 50, no. 1, 69–77.

Gazis D.C.; Herman R.; and Rothery R.W., 1961. *Nonlinear follow-the-leader models of traffic flow. Operations research*, 9, no. 4, 545–567.

Gipps P.G., 1981. *A behavioural car-following model for computer simulation. Transportation Research Part B: Methodological*, 15, no. 2, 105–111.

Gonçalves J. and Rossetti R.J.F., 2013. *Extending sumo to support tailored driving styles.* In *1st SUMO User Conference, DLR, Berlin-Adlershof, Germany.* vol. 21, 205–211.

Greenberg H., 1959. *An analysis of traffic flow. Operations research*, 7, no. 1, 79–85.

Greenshields B.; Channing W.; Miller H.; et al., 1935. *A study of traffic capacity.* In *Highway research board proceedings.* National Research Council (USA), Highway Research Board, vol. 1935.

Hoogendoorn S.P., 2016. *Chapter 4 - Fundamental diagrams.*

Kesting A. and Treiber M., 2008. *Calibrating car-following models by using trajectory data: Methodological study. Transportation Research Record: Journal of the Transportation Research Board*, , no. 2088, 148–156.

Knoop V.L. and Daamen W., 2017. *Automatic fitting procedure for the fundamental diagram. Transportmetrica B: Transport Dynamics*, 5, no. 2, 129–144.

Kometani E. and Sasaki T., 1959. *Dynamic behavior of traffic with a nonlinear spacing-speed relationship. Theory of Traffic Flow (Proc of Sym on TTF (GM))*, 105–119.

Krajzewicz D.; Hertkorn G.; Rössel C.; and Wagner P., 2002. *SUMO (Simulation of Urban MObility)-an open-source traffic simulation.* In *Proceedings of the 4th middle East Symposium on Simulation and Modelling (MESM20002).* 183–187.

Krauß S., 1998. *Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics.* Ph.D. thesis.

Mathew T.V., 2012. *Microscopic Traffic Simulation. Retrieved from National Programme on Technology Enhanced Learning: http://nptel iitm ac in/courses/105101008/downloads/cete_16 pdf.*

Mullen N.; Charlton J.; Devlin A.; and Bedard M., 2011. *Simulator validity: Behaviors observed on the simulator and on the road.*

Pipes L.A., 1953. *An operational analysis of traffic dynamics. Journal of applied physics*, 24, no. 3, 274–281.

Pipes L.A., 1966. *Car following models and the fundamental diagram of road traffic. Transportation Research/UK/.*

Ranjitkar P.; Nakatsuji T.; and Kawamua A., 2005. *Car-following models: an experiment based benchmarking. Journal of the Eastern Asia Society for Transportation Studies*, 6, 1582–1596.

Reuschel A., 1950. *Fahrzeugbewegungen in der Kolonne. Osterreichisches Ingenieur Archiv*, 4, 193–215.

Scott D.W., 1979. *On optimal and data-based histograms. Biometrika*, 66, no. 3, 605–610.

Shen J.; Qiu F.; Li R.; and Zheng C., 2017. *An extended car-following model considering the influence of bus. Tehnički vjesnik*, 24, no. 6, 1739–1747.

Slob J., 2008. *State-of-the-art driving simulators, a literature survey. DCT report*, 107.

Tang T.Q.; Chen L.; Yang S.C.; and Shang H.Y., 2015. *An extended car-following model with consideration of the electric vehicles driving range. Physica A: Statistical Mechanics and its Applications*, 430, 148–155.

Tang T.Q.; Shi W.F.; Shang H.Y.; and Wang Y.P., 2014. *An extended car-following model with consideration of the reliability of inter-vehicle communication. Measurement*, 58, 286–293.

Törnros J., 1998. *Driving behaviour in a real and a simulated road tunnela validation study. Accident Analysis & Prevention*, 30, no. 4, 497–503.

Van Hinsbergen C.; Schakel W.; Knoop V.; van Lint J.; and Hoogendoorn S., 2015. *A general framework for calibrating and comparing car-following models. Transportmetrica A: Transport Science*, 11, no. 5, 420–440.

Vasconcelos L.; Neto L.; Santos S.; Silva A.B.; and Seco Á., 2014. *Calibration of the gipps car-following model using trajectory data. Transportation Research Procedia*, 3, 952–961.

Wang J.; Rakha H.A.; and Fadhloun K., 2017a. *Comparison of car-following models: A vehicle fuel consumption and emissions estimation perspective.* Tech. rep.

Wang K.; Li Z.; Yao Q.; Huang W.; and Wang F.Y., 2007. *An automated vehicle counting system for traffic surveillance.* In *Vehicular Electronics and Safety, 2007. ICVES. IEEE International Conference on.* IEEE, 1–6.

Wang P.; Yu G.; Wu X.; Qin H.; and Wang Y., 2017b. *An extended car-following model to describe connected traffic dynamics under cyberattacks. Physica A: Statistical Mechanics and its Applications.*

Whitham G.B. and Lighthill M.J., 1955. *On kinematic waves II. A theory of traffic flow on long crowded roads. Proc R Soc Lond A*, 229, no. 1178, 317–345.

Zeyu J.; Gangqiao W.; Han X.; Yong W.; Xiangpeng Z.; and Yi L., 2016. *Calibrating car-following model with trajectory data by cell phone.* In *Proceedings of the Second ACM SIGSPATIALInternational Workshop on the Use of GIS in Emergency Management.* ACM, 12.

AUTHOR LISTING

AUTHOR LISTING

Abreu A. **35**
Alfrink M. **90**
Alves V. **19**
Amaro P. **131**
Antunes M. **67**

Baghcheband H. **123**
Brown K.N. **67**

Calado J.M.F. **35**
Camilleri L. **5**
Camilleri M. **5**
Castañé G.G. **67**
Charneca B. **19**
Chaves H. **19**
Conley W. **11/46**
Crespo A. **19**

Damiani L. **98**
Dias A. **35**
Dozortsev V. **26**
Dumond Y. **59**

Ferreira J. **131**
Fridman A. **41**

Gonçalves P.J.S. **77**
Guerry M.-A. **53**

Iazeolla G. **117**

Kocerka J. **111**
Kokkinogenis Z. **123**
Komenda T. **85**
Krześlak M. **111**

Leão G. **131**
Lin Y. **67**

Mazari-Abdessameud O. **53**
Mironova A. **26**
Morra E. **98**

Neves J. **19**

Oleynik A. **41**
Ozturk C. **67**

Pompei F. **117**

Ranz F. **85**
Requeijo J. **35**
Revetria R. **98**
Ribeiro J. **19**
Rojek I. **105**
Rossetti R.J.F. **123/131**
Rozhok A. **98**
Roβmann J. **90**

Santos F.M.S. **77**
Santos V. **19**
Sihn W. **85**
Simonis H. **67**
Spiteri L. **5**
Studziński J. **105**
Szelag B. **105**

Van Kerckhoven J. **53**
Van Utterbeeck F. **53**
Vicente H. **19**