International Journal on Advances in Systems and Measurements

2013 vol. 6 nr. 1&2

The *International Journal on Advances in Systems and Measurements* is published by IARIA.
ISSN: 1942-261x
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Responsibility for the contents rests upon the authors and not upon IARIA, nor on IARIA volunteers, staff, or contractors.

IARIA is the owner of the publication and of editorial aspects. IARIA reserves the right to update the content for quality improvements.

Abstracting is permitted with credit to the source. Libraries are permitted to photocopy or print, providing the reference is mentioned and that the resulting material is made available at no cost.

Reference should mention:

International Journal on Advances in Systems and Measurements, issn 1942-261x
vol. 6, no. 1 & 2, year 2013, http://www.iariajournals.org/systems_and_measurements/

The copyright for each included paper belongs to the authors. Republishing of same material, by authors or persons or organizations, is not allowed. Reprint rights can be granted by IARIA or by the authors, and must include proper reference.

Reference to an article in the journal is as follows:

<Author list>, "<Article title>"
International Journal on Advances in Systems and Measurements, issn 1942-261x
vol. 6, no. 1 & 2, year 2013, <start page>:<end page> , http://www.iariajournals.org/systems_and_measurements/

IARIA journals are made available for free, proving the appropriate references are made when their content is used.

Sponsored by IARIA
www.iaria.org

Copyright © 2013 IARIA

International Journal on Advances in Systems and Measurements
Volume 6, Number 1 & 2, 2013

Editor-in-Chief

Constantin Paleologu, University 'Politehnica' of Bucharest, Romania

Editorial Advisory Board

Vladimir Privman, Clarkson University - Potsdam, USA
Go Hasegawa, Osaka University, Japan
Winston KG Seah, Institute for Infocomm Research (Member of A*STAR), Singapore
Ken Hawick, Massey University - Albany, New Zealand

Editorial Board

Jemal Abawajy, Deakin University, Australia
Ermeson Andrade, Universidade Federal de Pernambuco (UFPE), Brazil
Al-Khateeb Anwar, Politecnico di Torino, Italy
Francisco Arcega, Universidad Zaragoza, Spain
Tulin Atmaca, Telecom SudParis, France
Rafic Bachnak, Texas A&M International University, USA
Lubomír Bakule, Institute of Information Theory and Automation of the ASCR, Czech Republic
Nicolas Belanger, Eurocopter Group, France
Lotfi Bendaouia, ETIS-ENSEA, France
Partha Bhattacharyya, Bengal Engineering and Science University, India
Karabi Biswas, Indian Institute of Technology - Kharagpur, India
Jonathan Blackledge, Dublin Institute of Technology, UK
Dario Bottazzi, Laboratori Guglielmo Marconi, Italy
Diletta Romana Cacciagrano, University of Camerino, Italy
Javier Calpe, Analog Devices and University of Valencia, Spain
Jaime Calvo-Gallego, University of Salamanca, Spain
Maria-Dolores Cano Baños, Universidad Politécnica de Cartagena,Spain
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain
Berta Carballido Villaverde, Cork Institute of Technology, Ireland
Vítor Carvalho, Minho University & IPCA, Portugal
Irinela Chilibon, National Institute of Research and Development for Optoelectronics, Romania
Soolyeon Cho, North Carolina State University, USA
Hugo Coll Ferri, Polytechnic University of Valencia, Spain
Denis Collange, Orange Labs, France
Noelia Correia, Universidade do Algarve, Portugal
Pierre-Jean Cottinet, INSA de Lyon - LGEF, France
Marc Daumas, University of Perpignan, France
Jianguo Ding, University of Luxembourg, Luxembourg
António Dourado, University of Coimbra, Portugal

Daniela Dragomirescu, LAAS-CNRS / University of Toulouse, France
Matthew Dunlop, Virginia Tech, USA
Mohamed Eltoweissy, Pacific Northwest National Laboratory / Virginia Tech, USA
Paulo Felisberto, LARSyS, University of Algarve, Portugal
Miguel Franklin de Castro, Federal University of Ceará, Brazil
Mounir Gaidi, Centre de Recherches et des Technologies de l'Energie (CRTEn), Tunisie
Eva Gescheidtova, Brno University of Technology, Czech Republic
Tejas R. Gandhi, Virtua Health-Marlton, USA
Marco Genovese, Italian Metrological Institute (INRIM), Italy
Teodor Ghetiu, University of York, UK
Franca Giannini, IMATI - Consiglio Nazionale delle Ricerche - Genova, Italy
Gonçalo Gomes, Nokia Siemens Networks, Portugal
João V. Gomes, University of Beira Interior, Portugal
Luis Gomes, Universidade Nova Lisboa, Portugal
Antonio Luis Gomes Valente, University of Trás-os-Montes and Alto Douro, Portugal
Diego Gonzalez Aguilera, University of Salamanca - Avila, Spain
Genady Grabarnik,CUNY - New York, USA
Craig Grimes, Nanjing University of Technology, PR China
Stefanos Gritzalis, University of the Aegean, Greece
Richard Gunstone, Bournemouth University, UK
Jianlin Guo, Mitsubishi Electric Research Laboratories, USA
Mohammad Hammoudeh, Manchester Metropolitan University, UK
Petr Hanáček, Brno University of Technology, Czech Republic
Go Hasegawa, Osaka University, Japan
Henning Heuer, Fraunhofer Institut Zerstörungsfreie Prüfverfahren (FhG-IZFP-D), Germany
Paloma R. Horche, Universidad Politécnica de Madrid, Spain
Vincent Huang, Ericsson Research, Sweden
Friedrich Hülsmann, Gottfried Wilhelm Leibniz Bibliothek - Hannover, Germany
Travis Humble, Oak Ridge National Laboratory, USA
Florentin Ipate, University of Pitesti, Romania
Imad Jawhar, United Arab Emirates University, UAE
Terje Jensen, Telenor Group Industrial Development, Norway
Liudi Jiang, University of Southampton, UK
Teemu Kanstrén, VTT Technical Research Centre of Finland, Finland
Kenneth B. Kent, University of New Brunswick, Canada
Fotis Kerasiotis, University of Patras, Greece
Andrei Khrennikov, Linnaeus University, Sweden
Alexander Klaus, Fraunhofer Institute for Experimental Software Engineering (IESE), Germany
Andrew Kusiak, The University of Iowa, USA
Vladimir Laukhin, Institució Catalana de Recerca i Estudis Avançats (ICREA) / Institut de Ciencia de Materials de Barcelona (ICMAB-CSIC), Spain
Kevin Lee, Murdoch University, Australia
Andreas Löf, University of Waikato, New Zealand
Jerzy P. Lukaszewicz, Nicholas Copernicus University - Torun, Poland
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France
Sathiamoorthy Manoharan, University of Auckland, New Zealand

Stefano Mariani, Politecnico di Milano, Italy
Paulo Martins Pedro, Chaminade University, USA / Unicamp, Brazil
Daisuke Mashima, Georgia Institute of Technology, USA
Don McNickle, University of Canterbury, New Zealand
Mahmoud Meribout, The Petroleum Institute - Abu Dhabi, UAE
Luca Mesin, Politecnico di Torino, Italy
Marco Mevius, HTWG Konstanz, Germany
Marek Miskowicz, AGH University of Science and Technology, Poland
Jean-Henry Morin, University of Geneva, Switzerland
Fabrice Mourlin, Paris 12th University, France
Adrian Muscat, University of Malta, Malta
Mahmuda Naznin, Bangladesh University of Engineering and Technology, Bangladesh
George Oikonomou, University of Bristol, UK
Arnaldo S. R. Oliveira, Universidade de Aveiro-DETI / Instituto de Telecomunicações, Portugal
Aida Omerovic, SINTEF ICT, Norway
Victor Ovchinnikov, Aalto University, Finland
Telhat Özdoğan, Recep Tayyip Erdogan University, Turkey
Gurkan Ozhan, Middle East Technical University, Turkey
Constantin Paleologu, University Politehnica of Bucharest, Romania
Matteo G A Paris, Universita` degli Studi di Milano,Italy
Vittorio M.N. Passaro, Politecnico di Bari, Italy
Giuseppe Patanè, CNR-IMATI, Italy
Marek Penhaker, VSB- Technical University of Ostrava, Czech Republic
Juho Perälä, VTT Technical Research Centre of Finland, Finland
Florian Pinel, T.J.Watson Research Center, IBM, USA
Ana-Catalina Plesa, German Aerospace Center, Germany
Miodrag Potkonjak, University of California - Los Angeles, USA
Alessandro Pozzebon, University of Siena, Italy
Vladimir Privman, Clarkson University, USA
Konandur Rajanna, Indian Institute of Science, India
Stefan Rass, Universität Klagenfurt, Austria
Candid Reig, University of Valencia, Spain
Teresa Restivo, University of Porto, Portugal
Leon Reznik, Rochester Institute of Technology, USA
Gerasimos Rigatos, Harper-Adams University College, UK
Luis Roa Oppliger, Universidad de Concepción, Chile
Ivan Rodero, Rutgers University - Piscataway, USA
Lorenzo Rubio Arjona, Universitat Politècnica de València, Spain
Claus-Peter Rückemann, Leibniz Universität Hannover / Westfälische Wilhelms-Universität Münster / North-German Supercomputing Alliance, Germany
Subhash Saini, NASA, USA
Mikko Sallinen, University of Oulu, Finland
Christian Schanes, Vienna University of Technology, Austria
Rainer Schönbein, Fraunhofer Institute of Optronics, System Technologies and Image Exploitation (IOSB), Germany
Guodong Shao, National Institute of Standards and Technology (NIST), USA
Dongwan Shin, New Mexico Tech, USA

Larisa Shwartz, T.J. Watson Research Center, IBM, USA
Simone Silvestri, University of Rome "La Sapienza", Italy
Diglio A. Simoni, RTI International, USA
Radosveta Sokullu, Ege University, Turkey
Junho Song, Sunnybrook Health Science Centre - Toronto, Canada
Leonel Sousa, INESC-ID/IST, TU-Lisbon, Portugal
Arvind K. Srivastav, NanoSonix Inc., USA
Grigore Stamatescu, University Politehnica of Bucharest, Romania
Raluca-Ioana Stefan-van Staden, National Institute of Research for Electrochemistry and Condensed Matter, Romania
Pavel Šteffan, Brno University of Technology, Czech Republic
Monika Steinberg, University of Applied Sciences and Arts Hanover, Germany
Chelakara S. Subramanian, Florida Institute of Technology, USA
Sofiene Tahar, Concordia University, Canada
Jaw-Luen Tang, National Chung Cheng University, Taiwan
Muhammad Tariq, Waseda University, Japan
Roald Taymanov, D.I.Mendeleyev Institute for Metrology, St.Petersburg, Russia
Francesco Tiezzi, IMT Institute for Advanced Studies Lucca, Italy
Theo Tryfonas, University of Bristol, UK
Wilfried Uhring, University of Strasbourg // CNRS, France
Guillaume Valadon, French Network and Information and Security Agency, France
Eloisa Vargiu, Barcelona Digital - Barcelona, Spain
Miroslav Velev, Aries Design Automation, USA
Dario Vieira, EFREI, France
Stephen White, University of Huddersfield, UK
M. Howard Williams, Heriot-Watt University, UK
Shengnan Wu, American Airlines, USA
Xiaodong Xu, Beijing University of Posts & Telecommunications, China
Ravi M. Yadahalli, PES Institute of Technology and Management, India
Yanyan (Linda) Yang, University of Portsmouth, UK
Shigeru Yamashita, Ritsumeikan University, Japan
Patrick Meumeu Yomsi, INRIA Nancy-Grand Est, France
Alberto Yúfera, Centro Nacional de Microelectronica (CNM-CSIC) - Sevilla, Spain
Sergey Y. Yurish, IFSA, Spain
David Zammit-Mangion, University of Malta, Malta
Guigen Zhang, Clemson University, USA
Weiping Zhang, Shanghai Jiao Tong University, P. R. China
J Zheng-Johansson, Institute of Fundamental Physic Research, Sweden

International Journal on Advances in Systems and Measurements
Volume 6, Numbers 1 & 2, 2013

CONTENTS

pages: 1 - 25
Characterizing and Fulfilling Traceability Needs in the PREDIQT Method for Model-based Prediction of System Quality
Aida Omerovic, SINTEF ICT, Norway
Ketil Stølen, SINTEF ICT & University of Oslo, Department of Informatics, Norway

pages: 26 - 39
Augmented Reality Visualization of Numerical Simulations in Urban Environments
Sebastian Ritterbusch, Karlsruhe Institute of Technology (KIT), Germany
Staffan Ronnas, Karlsruhe Institute of Technology (KIT), Germany
Irina Waltschlaeger, Karlsruhe Institute of Technology (KIT), Germany
Philipp Gerstner, Karlsruhe Institute of Technology (KIT), Germany
Vincent Heuveline, Karlsruhe Institute of Technology (KIT), Germany

pages: 40 - 56
An Explorative Study of Module Coupling and Hidden Dependencies based on the Normalized Systems Framework
Dirk van der Linden, University of Antwerp, Belgium
Peter De Bruyn, University of Antwerp, Belgium
Herwig Mannaert, University of Antwerp, Belgium
Jan Verelst, University of Antwerp, Belgium

pages: 57 - 71
Magnitude of eHealth Technology Risks Largely Unknown
Hans Ossebaard, RIVM National Institute for Public Health and the Environment,, Netherlands
Lisette van Gemert-Pijnen, University of Twente, Netherlands
Adrie de Bruijn, RIVM National Institute for Public Health and the Environment,, Netherlands
Robert Geertsma, RIVM National Institute for Public Health and the Environment,, Netherlands

pages: 72 - 81
Optimized Testing Process in Vehicles Using an Augmented Data Logger
Karsten Hünlich, Steinbeis Interagierende Systeme GmbH, Germany
Daniel Ulmer, Steinbeis Interagierende Systeme GmbH, Germany
Steffen Wittel, Steinbeis Interagierende Systeme GmbH, Germany
Ulrich Bröckl, University of Applied Sciences Karlsruhe, Germany

pages: 82 - 91
Modeling and Synthesis of mid- and long-term Future Nanotechnologies for Computer Arithmetic Circuits
Bruno Kleinert, Chair of Computer Architecture, University of Erlangen-Nürnberg, Germany
Dietmar Fey, Chair of Computer Architecture, University of Erlangen-Nürnberg, Germany

pages: 92 - 111
Developing an ESL Design Flow and Integrating Design Space Exploration for Embedded Systems

Falko Guderian, TU-Dresden, Germany
Gerhard Fettweis, TU-Dresden, Germany

pages: 112 - 123
6LoWPAN Gateway System for Wireless Sensor Networks and Performance Analysis
Gopinath Rao Sinniah, MIMOS Berhad, Malaysia
Zeldi Suryady Kamalurradat, MIMOS Berhad, Malaysia
Usman Sarwar, MIMOS Berhad, Malaysia
Mazlan Abbas, MIMOS Berhad, Malaysia
Sureswaran Ramadass, Universiti Sains Malaysia, Malaysia

pages: 124 - 136
Silicon Photomultiplier: Technology Improvement and Performance
Roberto Pagano, CNR-IMM, Italy
Sebania Libertino, CNR-IMM, Italy
Domenico Corso, CNR-IMM, Italy
Salvatore Lombardo, CNR-IMM, Italy
Giuseppina Valvo, STMicroelectronics, Italy
Delfo Sanfilippo, STMicroelectronics, Italy
Giovanni Condorelli, STMicroelectronics, Italy
Massimo Mazzillo, STMicroelectronics, Italy
Angelo Piana, STMicroelectronics, Italy
Beatrice Carbone, STMicroelectronics, Italy
Giorgio Fallica, STMicroelectronics, Italy

pages: 137 - 148
Application of the Simulation Attack on Entanglement Swapping Based QKD and QSS Protocols
Stefan Schauer, AIT Austrian Institute of Technology GmbH, Austria
Martin Suda, AIT Austrian Institute of Technology GmbH, Austria

pages: 149 - 165
Maximizing Utilization in Private IaaS Clouds with Heterogenous Load through Time Series Forecasting
Tomas Vondra, Dept. of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, Czech Republic
Jan Sedivy, Dept. of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, Czech Republic

pages: 166 - 177
RobustMAS: Measuring Robustness in Hybrid Central/Self-Organising Multi-Agent Systems
Yaser Chaaban, Institute of Systems Engineering, Leibniz University of Hanover, Germany
Christian Müller-Schloer, Institute of Systems Engineering, Leibniz University of Hanover, Germany
Jörg Hähner, Institute of Organic Computing, University of Augsburg, Germany

pages: 178 - 189
Optimization and Evaluation of Bandwidth-Efficient Visualization for Mobile Devices
Andreas Helfrich-Schkarbanenko, Karlsruhe Institute of Technology (KIT), Germany
Roman Reiner, Karlsruhe Institute of Technology (KIT), Germany
Sebastian Ritterbusch, Karlsruhe Institute of Technology (KIT), Germany
Vincent Heuveline, Karlsruhe Institute of Technology (KIT), Germany

pages: 190 - 199
LUT Saving in Embedded FPGAs for Cache Locking in Real-Time Systems

Antonio Martí Campoy, Universitat Politècnica de València, Spain
Francisco Rodríguez-Ballester, Universitat Politècnica de València, Spain
Rafael Ors Carot, Universitat Politècnica de València, Spain

pages: 200 - 213

Archaeological and Geoscientific Objects used with Integrated Systems and Scientific Supercomputing Resources

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster (WWU), Leibniz Universität Hannover, North-German Supercomputing Alliance (HLRN), Germany, Germany

pages: 214 - 223

Quantifying Network Heterogeneity by Using Mutual Information of the Remaining Degree Distribution

Lu Chen, Osaka University, Japan
Shin'ichi Arakawa, Osaka University, Japan
Masayuki Murata, Osaka University, Japan

pages: 224 - 234

An FPGA Implementation of OFDM Transceiver for LTE Applications

Tiago Pereira, Instituto de Telecomunicações, Portugal
Manuel Violas, Instituto de Telecomunicações, Universidade de Aveiro, Portugal
João Lourenço, Instituto Telecomunicações, Portugal
Atílio Gameiro, Instituto de Telecomunicações; Universidade de Aveiro, Portugal
Adão Silva, Instituto de Telecomunicações; Universidade de Aveiro, Portugal
Carlos Ribeiro, Instituto de Telecomunicações; Escola Superior de Tecnologia e Gestão, Instituto Politécnico de Leiria, Portugal

pages: 235 - 244

Comparison of Single-Speed GSHP Controllers with a Calibrated Semi-Virtual Test Bench

Tristan Salque, CSTB, France
Dominique Marchio, Mines Paristech, France
Peter Riederer, CSTB, France

Characterizing and Fulfilling Traceability Needs in the PREDIQT Method for Model-based Prediction of System Quality

Aida Omerovic* and Ketil Stølen*†
*SINTEF ICT, Pb. 124, 0314 Oslo, Norway
†University of Oslo, Department of Informatics, Pb. 1080, 0316 Oslo, Norway
Email: {aida.omerovic,ketil.stolen}@sintef.no

***Abstract*—Our earlier research indicated the feasibility of the PREDIQT method for model-based prediction of impacts of architectural design changes, on the different quality characteristics of a system. The PREDIQT method develops and makes use of a multi-layer model structure, called prediction models. Usefulness of the prediction models requires a structured documentation of both the relations between the prediction models and the rationale and assumptions made during the model development. This structured documentation is what we refer to as trace-link information. In this paper, we first propose a traceability scheme for PREDIQT. The traceability scheme specifies the needs regarding the information that should be traced and the capabilities of the traceability approach. An example-driven solution that addresses the needs specified through the scheme is then presented. Moreover, we propose an implementation of the solution in the form of a prototype traceability tool, which can be used to define, document, search for and represent the trace-links needed. The tool-supported solution is applied on prediction models from an earlier PREDIQT-based analysis of a real-life system. Based on a set of success criteria, we argue that our traceability approach is useful and practically scalable in the PREDIQT context.**

Keywords-traceability; system quality prediction; modeling; architectural design; change impact analysis; simulation.

I. INTRODUCTION

ICT systems are involved in environments which are constantly evolving due to changes in technologies, standards, users, business processes, requirements, or the ways systems are used. Both the systems and their operational environments frequently change over time and are shared. The new needs are often difficult to foresee, as their occurrence and system life time are insufficiently known prior to system development. Architectural adaptions are inevitable for accommodating the systems to the new services, processes, technologies, standards, or users. However, due to criticality of the systems involved, planning, implementation, testing and deployment of changes can not involve downtime or similar degradation of quality of service. Instead, the systems have to quickly and frequently adapt at runtime, while maintaining the required quality of service.

Independent of whether the systems undergoing changes are in the operation or in the development phase, important architectural design decisions are made often, quickly and with lack of sufficient information. When adapting the system architecture, the design alternatives may be many and the design decisions made may have unknown implications on the system and its quality characteristics (such as availability, security, performance or scalability). A change involving increased security may, for example, compromise performance or usability of a system.

The challenge is therefore how to achieve the necessary flexibility and dynamism required by software, while still preserving the necessary overall quality. Thus, there is a need for decision-making support which facilitates the analysis of effects of architectural adaptions, on the overall quality of a system as a whole.

In order to facilitate decision making in the context of what-if analyses when attempting to understand the implications of architectural design changes on quality of a system, models are a useful means for representing and analyzing the system architecture. Instead of implementing the potential architectural changes and testing their effects, model-based prediction is an alternative. Model-based prediction is based on abstract models which represent the relevant aspects of the system. A prediction based on models may address a desired number of architectural changes, without affecting the target system. As such, it is a quicker and less costly alternative to traditional implementation and testing performed in the context of understanding the effects of changes on system quality.

Important preconditions for model-based prediction are correctness and proper usage of the prediction models. In addition, the development and use of the prediction models has to be properly documented. In practice, traceability support requires process guidance, tool support, templates and notations for enabling the user to eventually obtain sufficiently certain predictions and document the underlying conditions. Our recent work has addressed this issue by proposing an approach to traceability handling in model-based prediction of system quality [1]. This paper provides refinements and several extensions of the approach, and elaborates further on the current state of the art with respect to traceability in the context of model-based prediction of system quality.

In addressing the above outlined needs and challenges re-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

lated to managing architectural changes, we have developed and tried out the PREDIQT method [2] [3] [4] aimed for predicting impacts of architectural design changes on system quality characteristics and their trade-offs. PREDIQT has been developed to support the planning and analyzing the architecture of the ICT systems in general, and to facilitate the reasoning about alternatives for potential improvements, as well as for the reasoning about existing and potential weaknesses of architectural design, with respect to individual quality characteristics and their trade-offs. The predictions obtained from the models provide propagation paths and the modified values of the estimates which express the degree of quality characteristic fulfillment at the different abstraction levels.

The process of the PREDIQT method guides the development and use of the prediction models, but the correctness of the prediction models and the way they are applied are also highly dependent on the creative effort of the analyst and his/her helpers. In order to provide additional help and guidance to the analyst, we propose in this paper a traceability approach for documenting and retrieving the rationale and assumptions made during the model development, as well as the dependencies between the elements of the prediction models. This paper proposes a traceability solution for PREDIQT to be used for predicting system quality. To this end, we provide guidance, tool support, templates and notations for correctly creating and using the prediction models. The major challenge is to define accurate and complete trace information while enabling usability and effectiveness of the approach.

The approach is defined by a traceability scheme, which is basically a feature diagram specifying capabilities of the solution and a meta-model for the trace-link information. As such, the traceability scheme specifies the needs regarding the information that should be traced and the capabilities of the traceability approach. The proposed traceability scheme deals with quality indicators, model versioning, cost and profit information, as well as the visualization of the impact on such values of different design choices. An example-driven solution that addresses the needs specified through the scheme is then presented.

Moreover, a prototype traceability tool is implemented in the form of a relational database with user interfaces which can be employed to define, document, search for and represent the trace-links needed. The tool-supported solution is illustrated on prediction models from an earlier PREDIQT-based analysis conducted on a real-life industrial system [5]. We argue that our approach is, given the success criteria for traceability in PREDIQT, practically useful and better than any other traceability approach we are aware of.

This paper is a revised and extended version of a full technical report [6]. The latter is an extended version of a paper [1] originally presented at and published in the proceedings of the SIMUL'11 conference. With respect to the SIMUL'11 conference paper [1], this paper is extended with:

1) An outline of the PREDIQT method.
2) Guidelines for application of the prediction models. The guidelines are used for eliciting the traceability scheme for our approach.
3) Further extensions and refinements of the traceability approach in PREDIQT with special focus on specification and handling of indicators during development and use of prediction models; handling of quality characteristic fulfillment acceptance levels; handling of timing aspects; versioning of prediction models; cost-benefit aspects in PREDIQT; and handling of usage profile in relation to the prediction models.
4) A way of practically visualizing the design decision alternatives has been proposed and exemplified.
5) Preliminary requirements for integration of the existing PREDIQT tool with the prototype traceability tool, have been specified and exemplified.

The paper is organized as follows: Section II provides background on traceability. An overview of the PREDIQT method is provided in Section III. Guidelines for application of both the prediction models and the trace-link information are provided in Section IV. The challenge of traceability handling in the context of the PREDIQT method is characterized in Section V. The traceability scheme is presented in Section VI. Our traceability handling approach is presented in Section VII. Section VIII illustrates the approach on an example. Section IX argues for completeness and practicability of the approach, by evaluating it with respect to the success criteria. Section X substantiates why our approach, given the success criteria outlined in Section V, is preferred among the alternative traceability approaches. The concluding remarks and future work are presented in Section XI.

II. Background on traceability

Traceability is the ability to determine which documentation entities of a software system are related to which other documentation entities according to specific relationships [7]. IEEE [8] also provides two definitions of traceability:

1) Traceability is the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another; for example, the degree to which the requirements and design of a given software component match.
2) Traceability is the degree to which each element in a software development product establishes its reason for existing.

Traceability research and practice are most established in fields such as requirements engineering and model-driven

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

engineering (MDE). Knethen and Paech [7] argue: "Dependency analysis approaches provide a fine-grained impact analysis but can not be applied to determine the impact of a required change on the overall software system. An imprecise impact analysis results in an imprecise estimate of costs and increases the effort that is necessary to implement a required change because precise relationships have to be identified during changing. This is cost intensive and error prone because analyzing the software documents requires detailed understanding of the software documents and the relationships between them." Aizenbud-Reshef et al. [9] furthermore state: "The extent of traceability practice is viewed as a measure of system quality and process maturity and is mandated by many standards" and "With complete traceability, more accurate costs and schedules of changes can be determined, rather than depending on the programmer to know all the areas that will be affected by these changes."

IEEE [8] defines a trace as "A relationship between two or more products of the development process." According to the OED [10], however, a trace is defined more generally as a "(possibly) non-material indication or evidence showing what has existed or happened". As argued by Winkler and von Pilgrim [11]: "If a developer works on an artifact, he leaves traces. The software configuration management system records who has worked on the artifact, when that person has worked on it, and some systems also record which parts of the artifacts have been changed. But beyond this basic information, the changes themselves also reflect the developer's thoughts and ideas, the thoughts and ideas of other stakeholders he may have talked to, information contained in other artifacts, and the transformation process that produced the artifact out of these inputs. These influences can also be considered as traces, even though they are usually not recorded by software configuration management systems."

A traceability link is a relation that is used to interrelate artifacts (e.g., by causality, content, etc.) [11]. In the context of requirements traceability, Winkler and von Pilgrim [11] argue that "a trace can in part be documented as a set of meta-data of an artifact (such as creation and modification dates, creator, modifier, and version history), and in part as relationships documenting the influence of a set of stakeholders and artifacts on an artifact. Particularly those relationships are a vital concept of traceability, and they are often referred to as traceability links. Traceability links document the various dependencies, influences, causalities, etc. that exist between the artifacts. A traceability link can be unidirectional (such as depends-on) or bidirectional (such as alternative-for). The direction of a link, however, only serves as an indication of order in time or causality. It does not constrain its (technical) navigability, so traceability links can always be followed in both directions".

In addition to the different definitions, there is no commonly agreed basic classification [11], that is, a classification of traceability links. A taxonomy of the main concepts within traceability is suggested by Knethen and Paech [7].

An overview of the current state of traceability research and practice in requirements engineering and model-driven development is provided by Winkler and von Pilgrim [11], based on an extensive literature survey. Another survey by Galvao and Goknil [12] discusses the state-of-the-art in traceability approaches in MDE and assesses them with respect to five evaluation criteria: representation, mapping, scalability, change impact analysis and tool support. Moreover, Spanoudakis and Zisman [13] present a roadmap of research and practices related to software traceability and identify issues that are open for further research. The roadmap is organized according to the main topics that have been the focus of software traceability research.

Traces can exist between both model- and non-model artifacts. The means and measures applied for obtaining traceability are defined by so-called traceability schemes. A traceability scheme is driven by the planned use of the traces. The traceability scheme determines for which artifacts and up to which level of detail traces can be recorded [11]. A traceability scheme thus defines the constraints needed to guide the recording of traces, and answers the core questions: what, who, where, how, when, and why. Additionally, there is tacit knowledge (such as why), which is difficult to capture and to document. A traceability scheme helps in this process of recording traces and making them persistent.

As argued by Aizenbud-Reshef et al. [9], the first approach used to express and maintain traceability was cross-referencing. This involves embedding phrases like "see section x" throughout the project documentation. Thereafter, different techniques have been used to represent traceability relationships including standard approaches such as matrices, databases, hypertext links, graph-based approaches, formal methods, and dynamic schemes [9]. Representation, recording and maintenance of traceability relations are classified by Spanoudakis and Zisman [13] into five approaches: single centralized database, software repository, hypermedia, mark-up, and event-based.

According to Wieringa [14], representations and visualizations of traces can be categorized into matrices, cross-references, and graph-based representations. As elaborated by Wieringa, the links, the content of the one artifact, and other information associated with a cross reference, is usually displayed at the same time. This is, however, not the case with traceability matrices. So, compared to traceability matrices, the user is (in the case of cross-references) shown more local information at the cost of being shown fewer (global) links. As models are the central element in MDE, graph-based representations are the norm. A graph can be transformed to a cross-reference. Regarding the notation, there is, however, no common agreement or standard, mostly because the variety and informality of different artifacts is not suitable for a simple, yet precise

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

notation. Requirements traceability graphs are usually just plain box-and-line diagrams [14].

Knethen and Paech [7] argue that the existing traceability approaches do not give much process support. They specify four steps of traceability process: 1) define entities and relationships, 2) capture traces, 3) extract and represent traces, and 4) maintain traces. Similarly, Winkler and von Pilgrim [11] state that traceability and its supporting activities are currently not standardized. They classify the activities when working with traces into: 1) planning for traceability, 2) recording traces, 3) using traces, and 4) maintaining traces. Traceability activities are generally not dependent on any particular software process model.

Trace models are usually stored as separate models, and links to the elements are (technically) unidirectional in order to keep the connected models or artifacts independent. Alternatively, models can contain the trace-links themselves and links can be defined as bidirectional. While embedded trace-links pollute the models, navigation is much easier [11]. Thus, we distinguish between external and internal storage, respectively. Anquetil at al. [15] argue: "Keeping link information separated from the artifacts is clearly better; however, it needs to identify uniquely each artifact, even fined-grained artifacts. Much of the recent research has focused on finding means to automate the creation and maintenance of trace information. Text mining, information retrieval and analysis of trace links techniques have been successfully applied. An important challenge is to maintain links consistency while artifacts are evolving. In this case, the main difficulty comes from the manually created links, but scalability of automatic solution is also an issue."

As outlined by Aizenbud-Reshef et al. [9], automated creation of trace-links may be based on text mining, information retrieval, analysis of existing relationships to obtain implied relations, or analysis of change history to automatically compute links.

Reference models are an abstraction of best practice and comprise the most important kinds of traceability links. There is nothing provably correct about reference models, but they derive their relevance from the slice of practice they cover. Nevertheless, by formalizing a reference model in an appropriate framework, a number of elementary desirable properties can be ensured. A general reference model for requirements traceability is proposed by Ramesh and Jarke [16], based on numerous empirical studies.

Various tools are used to set and maintain traces. Surveys of the tools available are provided by Knethen and Paech [7], Winkler and von Pilgrim [11], Spanoudakis and Zisman [13], and Aizenbud-Reshef et al. [9]. Bohner and Arnold [17] found that the granularity of documentation entities managed by current traceability tools is typically somewhat coarse for an accurate impact analysis.

III. An overview of the PREDIQT method

PREDIQT is a tool-supported method for model-based prediction of quality characteristics (performance, scalability, security, etc.). PREDIQT facilitates specification of quality characteristics and their indicators, aggregation of the indicators into functions for overall quality characteristic levels, as well as dependency analysis. The main objective of a PREDIQT-based analysis is prediction of system quality by identifying different quality aspects, evaluating each of these, and composing the results into an overall quality evaluation. This is useful, for example, for eliciting quality requirements, evaluating the quality characteristics of a system, run-time monitoring of quality relevant indicators, as well as verification of the overall quality characteristic fulfillment levels.

The PREDIQT method produces and applies a multi-layer model structure, called prediction models, which represent system relevant quality concepts (through "Quality Model"), architectural design (through "Design Model"), and the dependencies between architectural design and quality (through "Dependency Views"). The Design Model diagrams are used to specify the architectural design of the target system and the changes whose effects on quality are to be predicted. The Quality Model diagrams are used to formalize the quality notions and define their interpretations. The values and the dependencies modeled through the Dependency Views (DVs) are based on the definitions provided by the Quality Model. The DVs express the interplay between the system architectural design and the quality characteristics. Once a change is specified on the Design Model diagrams, the affected parts of the DVs are identified, and the effects of the change on the quality values are automatically propagated at the appropriate parts of the DV. This section briefly outlines the PREDIQT method in terms of the process and the artifacts.

A. Process and models

The process of the PREDIQT method consists of three overall phases: *Target modeling*, *Verification of prediction models*, and *Application of prediction models*. Each phase is decomposed into sub-phases, as illustrated by Figure 1.

Based on the initial input, the stakeholders involved deduce a high level characterization of the target system, its scope and the objectives of the prediction analysis, by formulating the system boundaries, system context (including the usage profile), system lifetime and the extent (nature and rate) of design changes expected.

As mentioned above, three interrelated sets of models are developed during the process of the PREDIQT method: Design Model which specifies system architecture, Quality Model which specifies the system quality notions, and Dependency Views (DVs) which represent the interrelationship between the system quality and the architectural design. Quality Model diagrams are created in the form of trees,

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

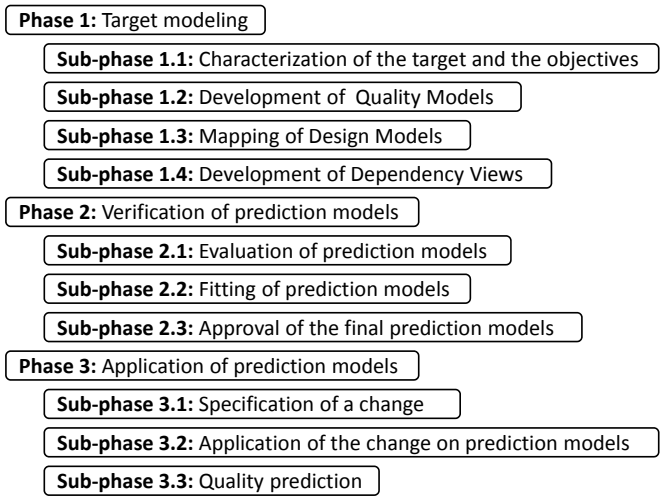


Figure 1. A simplified overview of the process of the PREDIQT method

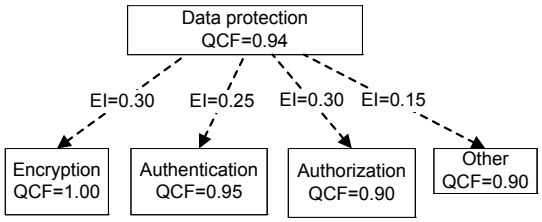


Figure 2. Excerpt of an example DV with fictitious values

by defining the quality notions with respect to the target system. The Quality Model diagrams represent a taxonomy with interpretations and formal definitions of system quality notions. The total quality of the system is decomposed into characteristics, sub-characteristics and quality indicators. The Design Model diagrams represent the architectural design of the system.

For each quality characteristic defined in the Quality Model, a quality characteristic specific DV is deduced from the Design Model diagrams and the Quality Model diagrams of the system under analysis. This is done by modeling the dependencies of the architectural design with respect to the quality characteristic that the DV is dedicated to, in the form of multiple weighted and directed trees. A DV comprises two notions of parameters:

1) EI: Estimated degree of Impact between two nodes, and
2) QCF: estimated degree of Quality Characteristic Fulfillment.

Each arc pointing from the node being influenced is annotated by a quantitative value of EI, and each node is annotated by a quantitative value of QCF.

Figure 2 shows an excerpt of an example DV with fictitious values. In the case of the *Encryption* node of Figure 2, the QCF value expresses the goodness of encryption with respect to the quality characteristic in question, e.g., security. A quality characteristic is defined by the underlying system specific Quality Model, which may for example be based on the ISO 9126 product quality standard [18]. A QCF value in a DV expresses to what degree the node (representing system part, concern or similar) is realized so that it, within its own domain, fulfills the quality characteristic. The QCF value is based on the formal definition of the quality characteristic (for the system under analysis), provided by the Quality Model. The EI value on an arc expresses the degree of impact of a child node (which the arc is directed to) on the parent node, or to what degree the parent node depends on the child node, with respect to the quality characteristic under consideration.

"Initial" or "prior" estimation of a DV involves providing QCF values to all leaf nodes, and EI values to all arcs. Input to the DV parameters may come in different forms (e.g., from domain expert judgments, experience factories, measurements, monitoring, logs, etc.), during the different phases of the PREDIQT method. The DV parameters are assigned by providing the estimates on the arcs and the leaf nodes, and propagating them according to the general DV propagation algorithm. Consider for example the *Data protection* node in Figure 2 (denoting: DP: Data protection, E: Encryption, AT: Authentication, AAT: Authorization, and O:Other):

$$QCF_{(DP)} = QCF_{(E)} \cdot EI_{(DP \rightarrow E)} + QCF_{(AT)} \cdot EI_{(DP \rightarrow AT)} + QCF_{(AAT)} \cdot EI_{(DP \rightarrow AAT)} + QCF_{(O)} \cdot EI_{(DP \rightarrow O)} \quad (1)$$

The DV-based approach constrains the QCF of each node to range between 0 and 1, representing minimal and maximal characteristic fulfillment (within the domain of what is represented by the node), respectively. This constraint is ensured through the formal definition of the quality characteristic rating (provided in the Quality Model). The sum of EIs, each between 0 (no impact) and 1 (maximum impact), assigned to the arcs pointing to the immediate children must be 1 (for model completeness purpose). Moreover, all nodes having a common parent have to be orthogonal (independent). The dependent nodes are placed at different levels when structuring the tree, thus ensuring that the needed relations are shown at the same time as the tree structure is preserved.

The general DV propagation algorithm, exemplified by (1), is legitimate since each quality characteristic specific DV is complete, the EIs are normalized and the nodes having a common parent are orthogonal due to the structure. A DV is complete if each node which is decomposed, has children nodes which are independent and which together fully represent the relevant impacts on the parent node, with respect to the quality characteristic that the DV is dedicated to. Two main means can be applied in order to facilitate that the children nodes fully represent the relevant impacts. First, in case not all explicit nodes together express the total impact, an additional node called "other" can

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

be added to each relevant sub-tree, thus representing the overall dependencies. Second, once the EI and QCF values have been assigned within a subtree, a possible lack of completeness will become more explicit. In such a case, either the EI estimates have to be modified, or additional nodes (for the missing dependencies) need to be added either explicitly, or in the form of an "other" node. In case "other" is used, it is particularly important to document the rationale (and other trace-link information) related to it.

The rationale for the orthogonality is that the resulting DV structure is tree-formed and easy for the domain experts to relate to. This significantly simplifies the parametrization and limits the number of estimates required, since the number of interactions between the nodes is minimized. Although the orthogonality requirement puts additional demands on the DV structuring, it has shown to represent a significant advantage during the estimation.

The "Verification of prediction models" is an iterative phase that aims to validate the prediction models, with respect to the structure and the individual parameters, before they are applied. A measurement plan with the necessary statistical power is developed, describing what should be evaluated, when and how. Both system-as-is and change effects should be covered by the measurement plan. Model fitting is conducted in order to adjust the DV structure and the parameters to the evaluation results. The objective of the "Approval of the final prediction models" sub-phase is to evaluate the prediction models as a whole and validate that they are complete, correct and mutually consistent after the fitting. If the deviation between the model and the new measurements is above the acceptable threshold after the fitting, the target modeling phase is re-initiated.

The "Application of the change on prediction models" phase involves applying the specified architectural design change on the prediction models. During this phase, a specified change is applied to the Design Model diagrams and the DVs, and its effects on the quality characteristics at the various abstraction levels are simulated on the respective DVs. When an architectural design change is applied on the Design Model diagrams, it is according to the definitions in the Quality Model, reflected to the relevant parts of the DV. Thereafter, the DV provides propagation paths and quantitative predictions of the new quality characteristic values, by propagating the change throughout the rest of each one of the modified DVs, based on the general DV propagation algorithm.

We have earlier developed tool support [5] based on Microsoft Excel for development of the DVs, as well as automatic simulation and sensitivity analysis in the context of the DVs. This tool was originally developed in order to serve as an early version providing a "proof-of-concept" and supporting the case studies on PREDIQT. Based on the PREDIQT method specification, and the early tool support, a new and enriched version of the PREDIQT tool has been developed, as presented in [19]. The former tool was developed on proprietary software, since MS Excel provided a rather simple and sufficient environment for quick prototyping. The last version of the tool, is however developed in the form of an Eclipse Modeling Framework (EMF) plugin. Both tools have recently been applied in full scale realistic industrial case studies. The existing PREDIQT tool support will in the following be referred to as the "PREDIQT tool."

B. Structure of the prediction models

Figure 3 provides an overview of the elements of the prediction models, expressed as a UML [20] class diagram. A Quality Model is a set of tree-like structures, which clearly specify the system-relevant quality notions, by defining and decomposing the meaning of the system-relevant quality terminology. Each tree is dedicated to a target system-relevant quality characteristic. Each quality characteristic may be decomposed into quality sub-characteristics, which in turn may be decomposed into a set of quality indicators. As indicated by the relationship of type aggregation, specific sub-characteristics and indicators can appear in several Quality Model trees dedicated to the different quality characteristics. Each element of a Quality Model is assigned a quantitative normalized metric and an interpretation (qualitative meaning of the element), both specific for the target system. A Design Model represents the relevant aspects of the system architecture, such as for example process, data flow, structure, and rules.

A DV is a weighted dependency tree dedicated to a specific quality characteristic defined through the Quality Model. As indicated by the attributes of the Class *Node*, the nodes of a DV are assigned a name and a QCF. A QCF (*Quality Characteristic Fulfillment*) is, as explained above, the value of the degree of fulfillment of the quality characteristic, with respect to what is represented by the node. The degree of fulfillment is defined by the metric (of the quality characteristic) provided in the Quality Model. Thus, a complete prediction model has as many DVs as the quality characteristics defined in the Quality Model. Additionally, as indicated by the *Semantic* dependency relationship, semantics of both the structure and the weights of a DV are given by the definitions of the quality characteristics, as specified in the Quality Model. A DV node may be based on a Design Model element, as indicated by the *Based on* dependency relationship. As indicated by the self-reference on the *Node* class, one node may be decomposed into children nodes. Directed arcs express dependency with respect to quality characteristic by relating each parent node to its immediate children nodes, thus forming a tree structure. Each arc in a DV is assigned an EI (*Estimated Impact*), which is a normalized value of degree of dependence of a parent node, on the immediate child node. Thus, there is a quantified dependency relationship from each parent node, to its immediate children. The values on the nodes and the arcs are

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

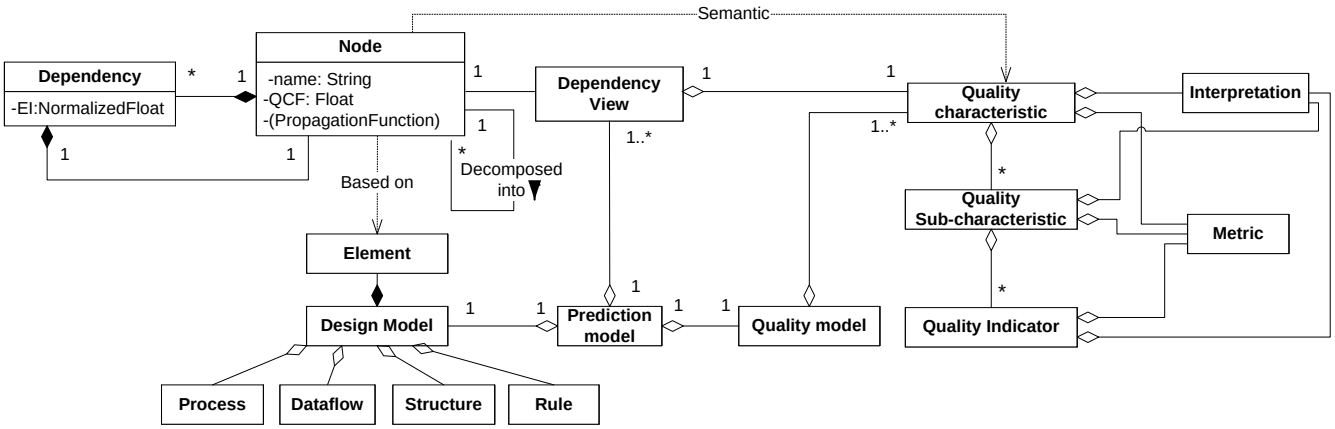


Figure 3. An overview of the elements of the prediction models, expressed as a UML class diagram

referred to as parameter estimates. We distinguish between prior and inferred parameter estimates. The former ones are, in the form of empirical input, provided on leaf nodes and all arcs, while the latter ones are deduced using the above presented DV propagation model for PREDIQT. For further details on PREDIQT, see Omerovic et al. [2], Omerovic and Stølen [21], Omerovic et al. [22], and Omerovic [4].

IV. Guidelines for application of prediction models

In order to facilitate quality and correct use of prediction models, this section provides guidelines for application of the prediction models and the trace-link information, with the analyst as the starting point. Thus, unless otherwise specified, all the guidelines are directed towards the analyst. Overall guidelines for the "Application of prediction models" – phase (i.e., Phase 3 of the PREDIQT process, see Figure 1) are presented first, followed by detailed guidelines for each one of its sub-phases: "Specification of a change", "Application of the change on prediction models" and "Quality prediction", respectively. The guidelines for each phase and sub-phase follow a standard structure:

- objective – specifies the goals of the phase
- prerequisites – specifies the conditions for initiating the phase
- how conducted – presents the detailed instructions for performing the steps that have to be undergone
- input documentation – lists the documentation that is assumed to be ready and available upon the initialization of the phase
- output documentation – lists the documentation that is assumed to be available upon the completion of the (sub)phase
- modeling guideline – lists the sequence of steps needed to be undergone in the context of modifying or applying the relevant prediction models.

The guidelines are based on the authors' experiences from industrial trials of PREDIQT [5] [3]. As such, the guidelines are not exhaustive but serve as an aid towards a more structured process of applying the prediction models and accommodating the trace information during the model development, based on the needs of the "Application of prediction models"-phase.

It should be noted that the guidelines presented in this section only cover Phase 3 of the PREDIQT process. This is considered as the essential phase for obtaining the predictions in a structured manner with as little individual influence of the analyst as possible. It would of course be desirable to provide corresponding guidelines for the first two phases of the PREDIQT process as well. For our current purpose, however, Phase 3 is essential and critical, while the guidance for carrying out phases 1 and 2 currently relies on the presentation of PREDIQT [4] and documentation of the case studies [2] [3].

It should also be noted that in the guidelines presented in this section, sub-phase 2 ("Application of the change on prediction models") is the most extensive one. In this phase, the specified change is first applied on the Design Model. Then, the dependencies within the Design Model are identified. Thereafter, the change is, based on the specification and the modified Design Model, reflected on the DVs. Once the DVs are modified, the modifications are verified. The modifications of both the Design Model and the DVs strongly depends on the semantics of the Quality Model which is actively used (but not modified) throughout the sub-phase. As such, the sub-phase involves modification of the Design Model and the DVs, based on the change specification and the Quality Model. Rather that splitting this sub-phase into two separate ones, we believe that it is beneficial to include all tasks related to application of a change on the prediction models in one (although extensive, yet) coherent sub-phase.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

A. Guidelines for the "Application of prediction models" – phase

Objective

During this phase, a specified change is applied to the prediction models, and its effects on the quality characteristics at the various abstraction levels are simulated on the respective Dependency Views (DVs). The simulation reveals which design parts and aspects are affected by the change and the degree of impact (in terms of the quality notions defined by the Quality Model).

Prerequisites

- The fitted prediction models are approved.
- The changes applied are assumed to be independent relative to each other.
- The "Quality prediction" sub-phase presupposes that the change specified during the "Specification of a change" sub-phase can be fully applied on the prediction models, during the "Application of the change on prediction models" sub-phase.

How conducted

This phase consists of the three sub-phases:

1) Specification of a change
2) Application of the change on prediction models
3) Quality prediction

Input documentation

- Prediction models: Design Model diagrams, Quality Model diagrams, and Dependency Views
- Trace-links

Output documentation

- Change specification
- Pre- and post-change Design Model diagrams
- DVs.

People that should participate

- Analysis leader (Required). Analysis leader is also referred to as analyst.
- Analysis secretary (Optional)
- Representatives of the customer:
  - Decision makers (Optional)
  - Domain experts (Required)
  - System architects or other potential users of PREDIQT (Required)

Modeling guideline

1) Textually specify the architectural design change of the system.
2) Modify the Design Model diagrams with respect to the change proposed. Modify the structure and the values of the prior parameters, on the affected parts of the DVs.
3) Run the simulation and display the changes on the Design Model diagrams and the DVs, relative to their original (pre-change) structure and values.

B. Guidelines for the "Specification of a change" sub-phase

Objective

The change specification should clearly state all deployment relevant facts necessary for applying the change on the prediction models. The specification should include the current and the new state and characteristics of the design elements/properties being changed, the rationale and the assumptions made.

Prerequisites

The fitted prediction models are approved.

How conducted

Specify the change by describing type of change, the rationale, who should perform it, when, how and in which sequence of events. In the case that the change specification addresses modifications of specific elements of the Design Model diagrams or the DVs, the quality characteristics of the elements before and after the change have to be specified, based on the definitions provided by the Quality Model. The change specification has to be at the abstraction level corresponding to the abstraction level of a sufficient subset of the Design Model diagrams or DVs.

Input documentation

- Prediction models
- Design Model
- Quality Model
- Dependency Views.

Output documentation

Textual specification of a change.

Modeling guideline

1) Textually specify an architectural design change of the system represented by the approved prediction models.
2) Specify the rationale and the process related to the change deployment.

C. Guidelines for the "Application of the change on prediction models" sub-phase

Objective

This sub-phase involves applying the specified change on the prediction models.

Prerequisites

- The change is specified.
- The specified change is, by the analyst and the domain experts, agreed upon and a common understanding is reached.

How conducted

Detailed instructions for performing the six steps specified in "Modeling guideline," are provided here.

1) This first step of relating the change to the Design Model diagram(s) and their elements is a manual effort. The analyst and the domain experts confirm that a common understanding of the specification has been reached. Then, they retrieve the diagrams and the respective elements of the Design Model and

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

identify which elements are potentially affected by the change, with respect to the system quality in general. The identified elements are marked, and their post-change status specified. The status may be of three types: update, delete or add. The update may involve change of a property related to design or a quality characteristic. In the case of delete, the diagram element is marked and its new status is visible. In the case of add, a new diagram element is introduced.

2) The trace-links between diagrams and diagram elements are (during the "Target modeling" phase) documented in the form of a database, which they can be retrieved from. Each one of the above identified Design Model diagrams and diagram elements (except the added ones) is searched in the existing trace-link database (created during the model development). The result displays those searched items which have the role of the origin or the target element, and all the elements that depend on them or that they are dependent on, respectively. The result also displays overall meta-data, e.g., the kinds of the trace-links and their rationale. The domain experts and the analyst identify those retrieved (linked) elements that are affected by the specified change. Depending on the nature of the change and the trace-link type and rationale, each diagram or element which, according to the search results is linked to the elements identified in the previous step, may be irrelevant, deleted or updated. The updated and the deleted elements are, within the diagrams, assigned the new (post-change) status and meta-data.
3) The trace-link database is searched for all the above identified elements which have been updated or deleted. The trace-links between those elements and the DV model elements are then retrieved. Then, the overall DV model elements that may be affected by the change are manually identified. The rationale for the DV structure and the node semantics regarding all the retrieved and manually identified DV model elements, are retrieved from the trace-link database. It is considered whether the added design element models require new DV nodes. The DV structure is manually verified, based on the retrieved trace-link information.
4) The domain experts and the analyst manually verify the updated structure (completeness, orthogonality, and correctness) of each DVs, with respect to the i) quality characteristic definitions provided by the Quality Model and ii) the modified Design Model.
5) The estimates of the prior parameters have to be updated due to the modifications of the Design Model and the DV structure. Due do the structural DV modification in the previous step, previously internal nodes may have become prior nodes, and the EIs on the arcs may now be invalid. New nodes and arcs may have been introduced. All the earlier leaf nodes which have become internal nodes, and all new internal nodes are assumed to automatically be assigned the function for the propagation model, by the PREDIQT tool. All the new or modified arcs and leaf nodes have to be marked so that the values of their parameters can be evaluated. The overall unmodified arcs and the leaf nodes whose values may have been affected by the change, are manually identified. In the case of the modified arcs and leaf nodes, trace-links are used to retrieve the previously documented rationale for the estimation of the prior parameter values and node semantics. The parameter values on the new and the modified arcs and leaf nodes are estimated based on the Quality Model.

The leaf node QCFs of a sub-tree are estimated before estimating the related EIs. The rationale is to fully understand the semantics of the nodes, through reasoning about their QCFs first. In estimating a QCF, two steps have to be undergone:

a) interpretation of the node in question – its contents, scope, rationale and relationship with the Design Model, and
b) identification of the relevant metrics from the Quality Model of the quality characteristic that the DV is addressing, as well as evaluation of the metrics identified.

When estimating a QCF the following question is posed (to the domain experts): "*To what degree is the quality characteristic fulfilled, given the contents and the scope of the node?*" The definition of the rating should be recalled, along with the fact that zero estimate value denotes no fulfillment, while one denotes maximum fulfillment.

In estimating an EI, two steps have to be undergone:

a) interpretation of the two nodes in question, and
b) determination of the degree of impact of the child node, on the parent node. The value is assigned relative to the overall EIs related to the same parent node, and with a consistent unit of measure, prior to being normalized. The normalized EIs on the arcs from the same parent node have to sum up to one, due to the requirement of model completeness.

When estimating an EI the following question is posed (to the domain experts): "*To what degree does the child node impact the parent node, or how dependent is the parent node on child node, with respect to the quality characteristic that the DV is dedicated to?*" The definition of the quality characteristic provided by its Quality Model, should be recalled and the estimate is provided relative to the impact of the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

overall children nodes of the parent node in question. Alternatively, an impact value is assigned using the same unit of measure on all arcs of the sub-tree, and normalized thereafter.

Once one of the above specified questions is posed, depending on the kind of the DV parameter, the domain expert panel is asked to provide the estimate with an interval so that the correct value is within the interval with a probability given by the confidence level [23].

6) Manually verify the updated prior parameter values, so that the relative QCF values are consistent to each other and the rest of the estimates, and so that EIs on the arcs from a common parent sum up to one.

If the specified change can be fully applied, it is within the scope of the prediction models, which is a prerequisite for proceeding to the next sub-phase. Otherwise, the modifications are canceled and the change deemed not predictable by the models as such.

Input documentation

- Prediction models: Design Model, Quality Model, Dependency Views
- Specification of the change
- The trace-links.

Output documentation

- Design Model
- DVs modified with respect to the change.

Modeling guideline

1) Relate the specified change to manually identifiable Design Model diagram(s) and their elements.
2) Use the trace-links to identify the affected parts (diagrams and diagram elements) of the Design Model. Apply the change by modifying (updating, deleting or adding) the identified affected parts of the Design Model.
3) Use the trace-links to identify the affected parts (nodes and dependency links) of each DV, by retrieving the traces from the modified and the deleted parts of the Design Model to the DVs, as well as the rationale for the DV structure and the node semantics. Modify the structure of the affected parts of the DVs.
4) Manually verify the updated structure (completeness, orthogonality, and correctness) of the DVs, with respect to the Quality Model and the modified Design Model.
5) Use trace-links to identify the documented rationale for the estimation of the prior parameter values. Manually identify the overall prior parameters that have been affected by the change. Use Quality Model to modify the values of the affected prior parameters (i.e., EIs and leaf node QCFs).
6) Manually verify the updated prior parameter values (that QCFs are consistent relative to each other and that EIs on the arcs from a common parent sum up to one).

D. Guidelines for the "Quality prediction" sub-phase

Objective

The propagation of the change throughout the rest of each one of the modified DVs, is performed. The propagation paths and the modified parameter values are obtained.

Prerequisites

The specified change is within the scope of and fully applied on the prediction models.

How conducted

Use the PREDIQT tool support to propagate the change. The tool explicitly displays the propagation paths and the modified parameter values, as well as the degrees of parameter value change. Obtain the predictions, in terms of the propagation paths and the parameter value modification. The result must explicitly express the changes with respect to the pre-change values. The propagation of the change throughout each one of the modified DVs, is performed based on the general DV propagation model, according to which the QCF value of each parent node is recursively calculated by first multiplying the QCF and EI value for each closest child and then summing up these products. Such a model is legitimate since each quality characteristic DV is complete, the EIs are normalized and the nodes having a common parent are orthogonal (with respect to the quality characteristic that the DV is dedicated to) due to the structure. The root node QCF values on the quality characteristic specific DVs represent the system-level rating value of the quality characteristic that the DV is dedicated to. If the predicted parameter values are beyond a pre-defined uncertainty threshold, the modifications are canceled and the change deemed not predictable by the input data and the models as such.

Input documentation

DVs.

Output documentation

- The change is propagated throughout the DVs, based on the DV propagation model.
- Propagation paths and parameter value changes (relative to the original ones) are displayed.

Modeling guideline

1) Run the simulation on the PREDIQT tool, in order to obtain the change propagation paths and the modified QCF values of the affected non-leaf nodes of the DVs.
2) Display the changes performed on the Design Model and the DVs (structure and the prior parameter values).

V. THE CHALLENGE

This section motivates and specifies the success criteria for the traceability handling approach in PREDIQT.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

A. Balancing the needs

Trace-link information can be overly detailed and extensive while the solution needed in a PREDIQT context has to be applicable in a practical real-life setting within the limited resources allocated for a PREDIQT-based analysis. Therefore, the traceability approach should provide sufficient breadth and accuracy for documenting, retrieving and representing of the trace-links, while at the same time being practically applicable in terms of comprehensibility and scalability. The right balance between the completeness and accuracy of the trace information on the one side, and practical usability of the approach on the other side, is what characterizes the main challenge in proposing the appropriate solution for traceability handling in PREDIQT. Therefore, the trace-link creation efforts have to be concentrated on the traces necessary during the application of the prediction models.

It is, as argued by Winkler and von Pilgrim [11], an open issue to match trace usage and traceability schemes, and to provide guidance to limit and fit traceability schemes in a such way that they match a projects required usage scenarios for traces. One of the most urgent questions is: what requirements a single scenario imposes on the other activities (in particular planning and recording) in the traceability process.

Moreover, it is argued by Aizenbud-Reshef et al. [9] that the lack of guidance as to what link information should be produced and the fact that those who use traceability are commonly not those producing it, also diminishes the motivation of those who create and maintain traceability information. In order to avoid this trap, we used the PREDIQT guidelines (as documented in Section IV) for the analyst as a starting point, for deriving the specific needs for traceability support.

B. Success criteria

The specific needs for traceability support in PREDIQT are summarized below:

1) There is need for the following kinds of trace-links:
   - Links between the Design Model elements to support identification of dependencies among the elements of the Design Model.
   - Links from the Design Model elements to DV elements to support identification of DV nodes which are based on specific elements of the Design Model.
   - Links from DV elements to Quality Model elements to support acquisition of traces from the prior estimates of the DV to the relevant quality indicators.
   - Links to external information sources (documents, cost information, profit information, usage profile, indicator definitions, indicator values, measurements, domain expert judgments) used during the development of DV structure and estimation of the parameters to support documenting the traces from the DV to the more detailed information sources available outside the prediction models.
   - Links to rationale and assumptions for:
     – Design Model elements
     – the semantics of the DV elements
     – the structure of the DVs
     – prior parameter estimates of the DVs

     The objective of these links is to support documenting the relevant aspects of the development of the prediction models, particularly the understanding and interpretations that the models are based on. Part of rationale and assumptions are also specifications of the acceptable values of quality characteristic fulfillment (also called quality characteristic fulfillment acceptance criteria/levels) as well as validity of input and models w.r.t. time (timing validity applies to Design Model and the DVs).
2) The traceability approach should have facilities for both searching with model types and model elements as input parameters, as well as for reporting linked elements and the link properties
3) The traceability approach should be flexible with respect to granularity of trace information
4) The traceability approach should be practically applicable on real-life applications of PREDIQT

These needs are in the sequel referred to as the success criteria for the traceability approach in PREDIQT.

VI. Traceability scheme

We propose a traceability scheme in the form of a meta-model for trace-link information and a feature diagram for capabilities of the solution. The traceability scheme specifies the needs regarding the information that should be traced and the capabilities of the traceability approach. Thus, our traceability scheme is based on the guidelines for application of the prediction models and the success criteria for the traceability approach specified in the two previous respective sections.

The types of the trace-links and the types of the traceable elements are directly extracted from Success Criterion 1 and represented through a meta-model shown by Figure 4. The *Element* abstract class represents a generalization of a traceable element. The *Element* abstract class is specialized into the five kinds of traceable elements: *Design Model Element*, *DV Element*, *Quality Model Element*, *External Information Source*, and *Rationale and Assumptions*. Similarly, the *Trace Link* abstract class represents a generalization of a trace-link and may be assigned a rationale for the trace-link. The *Trace Link* abstract class is specialized into the six kinds of trace-links.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

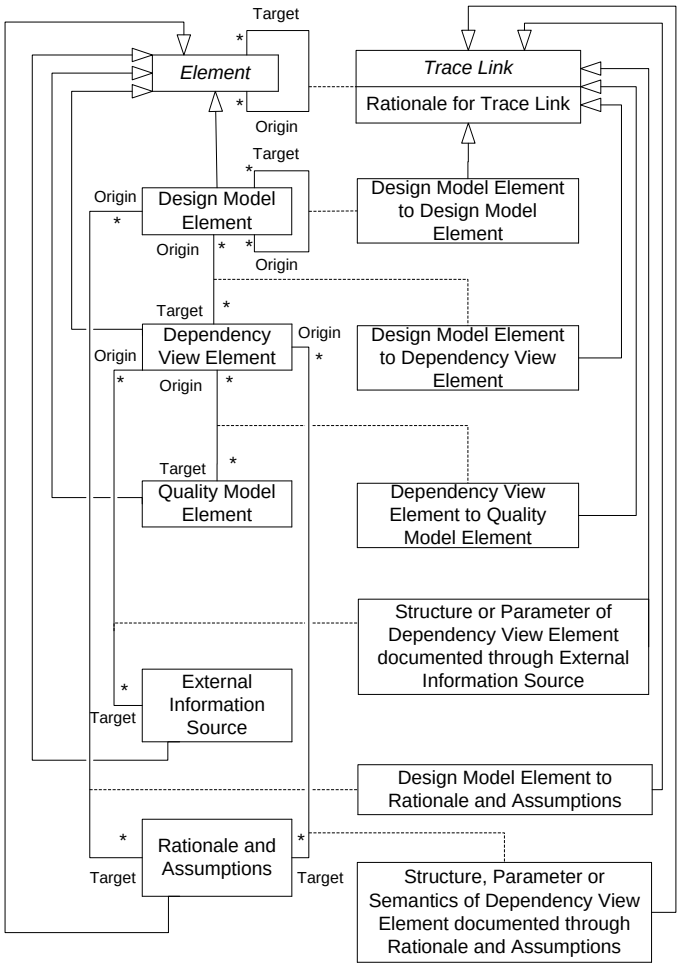


Figure 4. A meta model for trace-link information, expressed as a UML class diagram

Pairs of certain kinds of traceable elements form binary relations in the form of unidirectional trace-links. Such relations are represented by the UML-specific notations called association classes (a class connected by a dotted line to a link which connects two classes). For example, trace-links of type *Design Model Element to Design Model Element* may be formed from a *Design Model Element* to a *Dependency View Element*. The link is annotated by the origin (the traceable element that the trace-link goes from) and the target (the traceable element that the trace-link goes to) in order to indicate the direction. Since only distinct pairs (single instances) of the traceable elements (of the kinds involved in the respective trace-links defined in Figure 4) can be involved in the associated specific kinds of trace-links, uniqueness (property of UML association classes) is present in the defined trace-links. Due to the binary relations (arity of value 2) in the defined trace-links between the traceable elements, only two elements can be involved in any trace-link. Furthermore, multiplicity of all the traceable elements involved in the trace-links defined is of type "many," since an element can participate in multiple associations (given they are defined by the meta-model and unique).

The main capabilities needed are represented through a feature diagram [11] shown by Figure 5. Storage of trace-links may be internal or external, relative to the prediction models. A traceable element may be of type prediction model element (see Figure 3) or non-model element. Reporting and searching functionality has to be supported. Trace-link info has to include link direction, link meta-data (e.g., date, creator, strength) and cardinality (note that all links are binary, but a single element can be origin or target for more than one trace-link). Typing at the origin and the target ends of a trace-link, as well as documenting the rationale for the trace-link, are optional.

VII. Example-driven solution

This section presents the main aspects of our traceability approach for PREDIQT. We focus particularly on traceability of indicators by elaborating on the role of indicators in the Quality Model and the DVs and proposing a template for specification of indicators. Moreover, we elaborate on how to specify quality characteristic fulfillment acceptance criteria within the traceability approach. This is followed by a proposal for how to handle validity of models w.r.t time in the form of model versions. Furthermore, traceability of cost and profit information is discussed. Our traceability approach also includes handling of usage profile in the prediction models. The usage profile handling is presented before proposing how to visualize the impacts of the different the decision alternatives on quality characteristics, cost and profit. Additionally, a prototype traceability tool for trace-link management, implementing the needs specified through the traceability scheme, is presented. Finally, we propose the preliminary steps for integration of the prototype traceability tool with the existing PREDIQT tool.

A. Traceability of indicators

As stated above in relation to Success Criterion 1, links to external information sources include definitions and values of indicators. In PREDIQT, indicators are used as a part of the Quality Model in order to define the quality notions for the system being considered. The Quality Model, however, only defines the meaning of the terminology (i.e., quantitative and qualitative aspects of the quality notions specific to the target of analysis). Therefore, in addition to the Quality Model, indicator definitions and values are also associated with the DVs, through the traceability information. The indicators defined in relation to the DVs may be the same or additional w.r.t. the ones defined in the Quality Model. The reason for this is the fact that the DVs are an instantiation of the architectural dependency specific to the system in question. Hence, indicators may be attached to both QCFs and EIs at any part of the DVs. Most common use of an

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

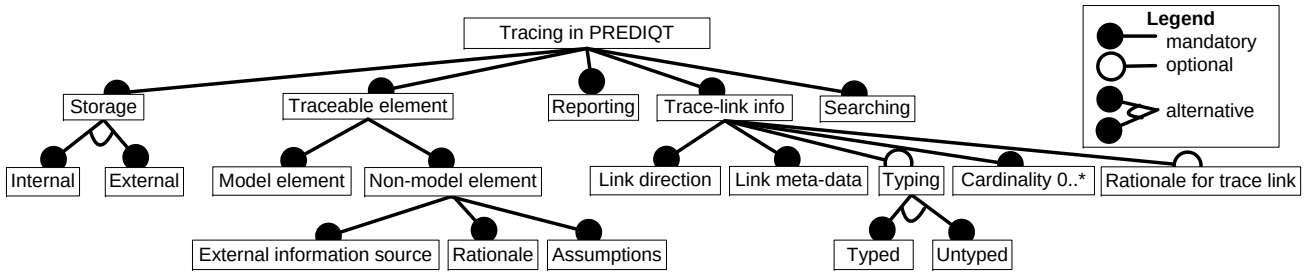


Figure 5. Main capabilities of the traceability approach, expressed as a feature diagram

indicator in the DV context is in relation to a leaf node QCF, where the indicator serves as a partial evaluator of the QCF value. The indicator value may be subject to dynamic change. The relationship between the indicator and the QCF may be linear or non-linear, and a mapping function should be defined. There may also be exceptions concerning the impact of the indicator value on the QCF which the indicator is related to. Moreover, one indicator may be related to several DV parameters. The dynamics of the indicators, their measurability in terms of empirical input, the loose relationship with the DV parameters, their possible relationship with several DV parameters simultaneously, and possible deviation of the mapping function from the general DV propagation model, distinguish the indicators from the regular DV parameters.

In order to make the indicator specification and evaluation as precise and streamlined as possible, we propose a template for specification of indicators, as well as a template for documenting the indicator measurement results. Table I provides a template for the specification of an indicator. The first column lists the names of the attributes relevant for the specification, while the second column provides the explanation and the guidelines regarding the input needed. Not all the attributes will be as relevant in a practical context. For example, the ISO 9126 product quality standard [18] defines a set of quality characteristic metrics using a similar but smaller set of attributes. The precision of the specification will also depend on how automatized the acquisition of the indicator values is, as well as how often the indicator values have to be retrieved. For example, a real-time monitoring environment automatically collecting dynamic indicators in order to capture irregularities in measurement patterns, will depend on a more precise definition of an indicator than a static value being evaluated between long intervals. Importance of the indicator also depends on the impact of its value (and the related DV parameter) on the rest of the model, acceptance values for the quality levels propagated, as well as the effect of the uncertainty on the rest of the model.

Table II provides a template for documenting the revision history concerning an indicator specification (defined in Table I). The relevant information regarding the revision of a specification is included here. The first column lists the names of the attributes relevant for the revision history, while the second column provides the explanation and guidelines regarding the input needed.

Table III provides a template for documenting the measurement history of an indicator (specified through the template in Table I). Each measurement is documented, and the value in the first attribute represents the instantiation of the indicator according to its latest specification.

Both the specification and the instantiation of an indicator has to be documented by a traceability approach. The process of identifying the relevant indicators and specifying them is a part of the development of the Quality Model and the DVs. The measurement of the indicator values is however only relevant in the context of the development, validation and application of the DVs. Therefore, Table I and Table II may be used in relation to both the Quality Model and the DVs, while Table III will only be used in the DV context.

B. Traceability of quality characteristic fulfillment acceptance levels

As mentioned in relation to Success Criterion 1, a part of the trace-link information regarding the rationale and assumptions are also specifications of the acceptable values of quality characteristic fulfillment. This basically means that for each quality characteristic defined in the Quality Model and instantiated through a DV, the acceptance levels for the QCF of the DV root node should be defined. As the acceptance level may vary at the different levels of a DV, it may also be defined w.r.t. other nodes than the root. The intervals between the acceptance levels depend on the risk attitude and the utility function of the decision maker, as well as on the predefined goals of the organization/stakeholders.

The advantage of defining the acceptance levels at the different nodes of a DV, is that early symptoms of irregularities or weaknesses can be captured by the model (as a part of, for example, run-time monitoring where indicator values are mapped to the DV-parameters), instead of waiting until a significant deviation has been propagated on the root node and then detected in relation to a higher abstraction level. In practice, this means that the acceptance scale can be even

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Table I
TEMPLATE FOR SPECIFICATION OF AN INDICATOR

Specification attributes for the indicator	Explanation of the specification attributes
Unique indicator id	Give each indicator a unique identifier.
Name of the indicator	State a concise, result-oriented name for the indicator. The name should reflect what the indicator expresses.
Definition	Specify the qualitative and the quantitative definition of the indicator. The definition should include the qualitative and the quantitative definitions of the variables.
Created by	Specify the name and the affiliation of the person that the indicator has been specified by.
Date created	Specify the date for the specification of the indicator.
Purpose of the indicator	Specify the purpose of the indicator, i.e., what it will be used for.
Assumptions	Specify any assumptions made for the indicator specification and its values.
Measurement guidelines	Specify how to obtain the indicator values and who is responsible for that.
Data source	Specify where the indicator values are stored, or where they are to be retrieved or measured from.
Measurement frequency	Specify how often the indicator values should be retrieved.
Trigger for measurement	Identify the events, states or values that initiate a new measurement of this indicator.
Preconditions	List any activities that must take place, or any conditions that must be true, before the indicator can be measured. Number each precondition sequentially.
Postconditions	Describe the state of the system at the conclusion of the indicator measurement. Number each postcondition sequentially.
Expected change frequency	Specify how often the value of the indicator is expected to change, i.e., the dynamics of the indicator.
Unit of measure	Specify the unit of measure of the indicator.
Interpretation of the value measured	Specify which indicator values are: preferred, realistic, extreme, within the normal range, and on the border to the unacceptable.
Scale	Provide the scale that should be used for the indicator measurement. (Scale types: nominal, ordinal, interval, or ratio).
Uncertainty	Specify degree of uncertainty and sources of uncertainty. Express uncertainty in the form of interval, confidence level, variance or similar.
How related to the relevant diagram parameters (function and instantiation coefficients)	Specify which diagrams and parameters of the diagrams the indicator is related to. Specify the mapping function, any exceptions and what values the possible coefficients of the indicator function should be instantiated with.
Notes and issues	Specify any additional notes or issues.

Table II
TEMPLATE FOR DOCUMENTING REVISION HISTORY CONCERNING AN INDICATOR SPECIFICATION

Revision attributes	Explanation of the revision attributes
Specification last updated by	Provide the name of the person who was the last one to update the specification.
Specification last updated date	Provide the date when the specification was last updated.
Reason for changes	Provide the reason to the update.
Version	Provide a version number of the specification.

Table III
TEMPLATE FOR DOCUMENTING MEASUREMENT HISTORY CONCERNING AN INDICATOR

Measurement attributes	Explanation of the measurement attributes
Measured value	Provide the indicator value from the latest measurement.
Measured by	Provide the name of the person/service that the measurement was performed by.
Date of measurement	Provide the date/time of the measurement.
Remarks	Provide and any additional info if appropriate.

more fine grained and more context specific, when mapped to several abstraction levels of a DV.

Note that the length of the intervals between the different acceptance levels may very significantly. Note also that the interpretation of a certain value of a quality characteristic (as defined through the Quality Model) is constant, while what is the acceptable value may vary, depending on which DV node a QCF is related to. Therefore, acceptance level and interpretation of a QCF value are two different notions. It is up to the stakeholders (mainly the decision makers) how fine or coarse grained the acceptance scale for a quality characteristic fulfillment (at the selected parts of a DV) should be. An example of a specification of the acceptance levels for root node QCF (always ranging between 0 and 1) of a DV representing quality characteristic *availability* is:

- $0.999 \leq$QCF – Very good
- $0.990 \leq$QCF<0.999 – Acceptable and compliant with the SLA goals
- $0.90 \leq$QCF<0.990 – According to the sector standards, but not sufficiently high for all services

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

- QCF<0.90 – Not acceptable

Consolidated traceability information regarding interval specification, interval measurement and the acceptance levels, allows for relating the interval values to the acceptance levels of the QCFs. Therefore, the sensitivity and dynamics (i.e., the frequency of change) of the indicator value, as well as the granularity of the acceptance level of the related QCF, will be among the factors influencing how often the indicator value should be measured in order to capture the irregular patterns and generally achieve the observability of the system and its aimed quality fulfillment level.

C. Traceability of model versions

As mentioned in relation to Success Criterion 1, a part of the trace-link information regarding the rationale and assumptions is also an explicit specification of validity of the input and the models w.r.t. time. The objective is to document when and for how long a model version of elements/parameters of a model are valid. The timing validity in the PREDIQT context applies to the Design Model and the DVs; the Quality Model is assumed to be static.

In order to address the timing aspect in the prediction models, we introduce the model versioning. A model or a trace-link information which has time-dependent validity is annotated with the versions which are valid at specified intervals of time. As such, versioning of both the Design Model and the DVs as well as versioning of the traceability info, is a tool for mapping the states of the system to the time.

The degree of the variation of models over time provides understanding of the needs for scalability as well as the overhead related to maintenance of an architecture. The reason is that an architecture which seems to be optimal at a certain point of time, may not represent the generally optimal solution, due to the changes expected in the long term. Therefore, in order to accommodate the long-term needs for scaling and adoptions, the relevant prediction models should be specified in terms of their time-dependent versions.

To support versioning, a set of attributes should be added to a trace-link or a model. Table IV presents the attributes needed and provides a template for specification of timing validity of models and trace-links. Not all the attributes specified will be as relevant in a practical context, but among the mandatory fields should be: “applies to trace-link element”, “version number”, and at least one of the following: “valid from”, “valid until”, “precondition for validity”, “postcondition for validity.”

D. Traceability of cost and profit information

As stated above in relation to Success Criterion 1, links to external information sources also include cost information. Often, the decision making around the architecture design alternatives has to take into account not only impact of changes on quality characteristics, but also on cost and profit. We argue that the traceability approach in the PREDIQT context can accommodate such a multi-dimensional cost-benefit analysis.

A prerequisite for including cost in the prediction models, is a cost model. By cost we mean a monetary amount that represents the value of resources that have to be used in relation to a treatment or deployment of a measure. A cost model should define and decompose the notion of cost for the architecture in question. As such, the cost model will have the same role in the context of cost, that the Quality Model has in the context of quality. An example of a Cost Model is shown in Figure 6. The rightmost nodes represent possible indicators, which should be specified using Table I and Table II. The decomposition of the cost notions is based on the architecture design models, and particularly the process models related to the deployment of a measure.

Once the cost notions are defined and decomposed, the cost information may be added in the form of trace-link information and attached to the relevant parts of the DVs. A preferred way of instantiating the cost model, is however by developing a dedicated DV for cost, according to the same principles as the ones used for developing quality characteristic specific DVs. Thus, cost will become a new explicit and separate concern, treated equally as each quality characteristic. Consequently, the cost specific DVs will provide predictions of impact of changes on monetary cost.

However, the profit may also be of monetary kind and it will not necessarily only be related to improved quality characteristics. Therefore, the profit should be treated in the same manner as cost and the respective quality characteristics, i.e., as a separate concern in the form of a Profit Model and a dedicated DV. Finally, the benefit of a decision alternative should be represented as a function of both the cost and the profit according to a specified utility function.

E. Traceability of usage profile

As mentioned in relation to Success Criterion 1, usage profile is a part of the trace-link information classified under the external information sources. Some of the DV parameters are in fact based on the usage profile. For example, the expected licensing costs as well as scalability needs, may be subject to to the usage profile. Moreover, the uncertainty of the estimates will be based on to what degree the usage profile is known and relevant for the parameters under consideration. Most importantly, when considering the alternative solutions for deployment of an architecture design, the usage profile information will be crucial, in order to meet the needs for accommodating the operational environment to the expected usage. The characteristics of the usage profile should be specified in terms of for example:

- number of clients
- number of servers
- number of data messages
- number of logons

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Table IV
TEMPLATE FOR DOCUMENTING TIMING VALIDITY OF MODELS AND TRACE-LINKS

Validity relevant attributes	Explanation of attributes
Applies to trace-link element	Specify which trace-link element this version specification applies to.
Version number	Provide a unique version number.
Valid from	Specify exactly when the trace-link or the model element in question is valid from.
Valid until	Specify exactly when the trace-link or the model element in question is valid until.
Precondition for validity	List any events or states that must take place, or any conditions that must be true, before this version can become valid. Number each precondition sequentially.
Postcondition for validity	Describe any events or states at the conclusion of the validity of this version. Number each postcondition sequentially.
Preceding version	If appropriate, specify which version should be succeeded by this one.
Version which succeeds this one	If appropriate, specify the version that should become valid after this one.
Rationale for the timing limitation	Explain and substantiate why the validity of this trace-link element is limited w.r.t. time.
Assumptions for the validity	Specify the assumptions for this specification, if any.

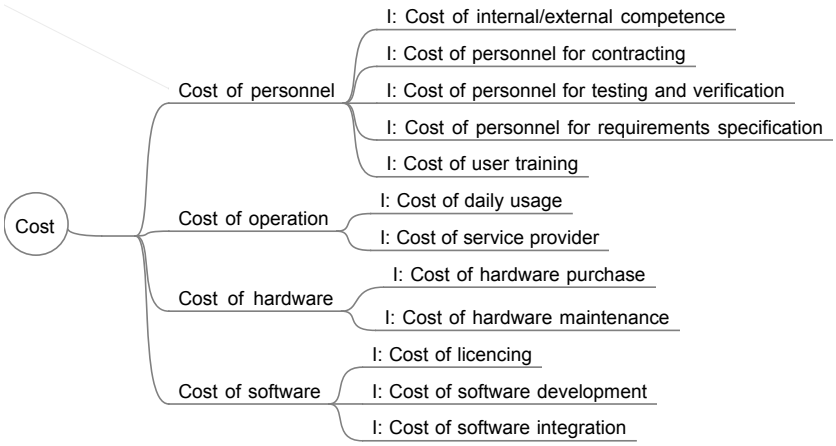


Figure 6. An example of a cost model

- number of users
- number of retrievals per user and per unit of time
- size of messages.

F. Visualization of the decision alternatives

Once the complete prediction models have been developed with the trace-link information, the application of the prediction models will result in predictions w.r.t three kinds of concerns:

- each quality characteristic as defined by the Quality Model
- cost as defined by the Cost Model
- profit as defined by the Profit Model.

As a result, the impacts of a decision alternative w.r.t. the current values of these three kinds of concerns may be difficult to compare. In order to facilitate the comparison, we propose a tabular visualization of the impacts of the alternative design decisions on each quality characteristic, as well as cost and profit. A simplified example of such a representation is illustrated in Table V. Thus, we distinguish between alternatives based on:

- value of each quality characteristic (i.e., the root node QCF of each quality characteristic specific DV)
- cost value (i.e., the root node value of the cost specific DV)
- profit value (i.e., the root node value of the profit specific DV).

In order to compare the alternatives with the current solution, one should take into account the risk attitude and the utility function of the decision maker. A simple way of doing this, is by weighting the quality characteristics, cost and profit with respect to each other. The constraints of the utility function will be the quality characteristic fulfillment acceptance levels, proposed in Section VII-B.

G. Prototype traceability tool

We have developed a prototype traceability tool in the form of a database application with user interfaces, on the top of Microsoft Access [24]. Similarly as for the first version of the PREDIQT tool, the proprietary development environment (Microsoft Access) was found suitable since it offers a rather simple and sufficient toolbox for quick prototyping of the proof-of-concept. A later version of the traceability tool may however use another (open source or similar) environment. The current prototype traceability tool includes a structure of tables for organizing the trace

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Table V
A POSSIBLE VISUALIZATION OF THE IMPACTS OF THE DIFFERENT ARCHITECTURE DESIGN ALTERNATIVES ON QUALITY, COST AND PROFIT

Architecture design alternative	Availability QCF	Scalability QCF	Usability QCF	Cost	Profit
Current architecture	0.999	0.90	0.95	85 000 EUR	120 000 EUR
Alternative 1	0.92	0.95	0.80	55 000 EUR	85 000 EUR
Alternative 2	0.90	0.85	0.99	60 000 EUR	90 000 EUR
Alternative 3	0.85	0.99	0.90	95 000 EUR	130 000 EUR

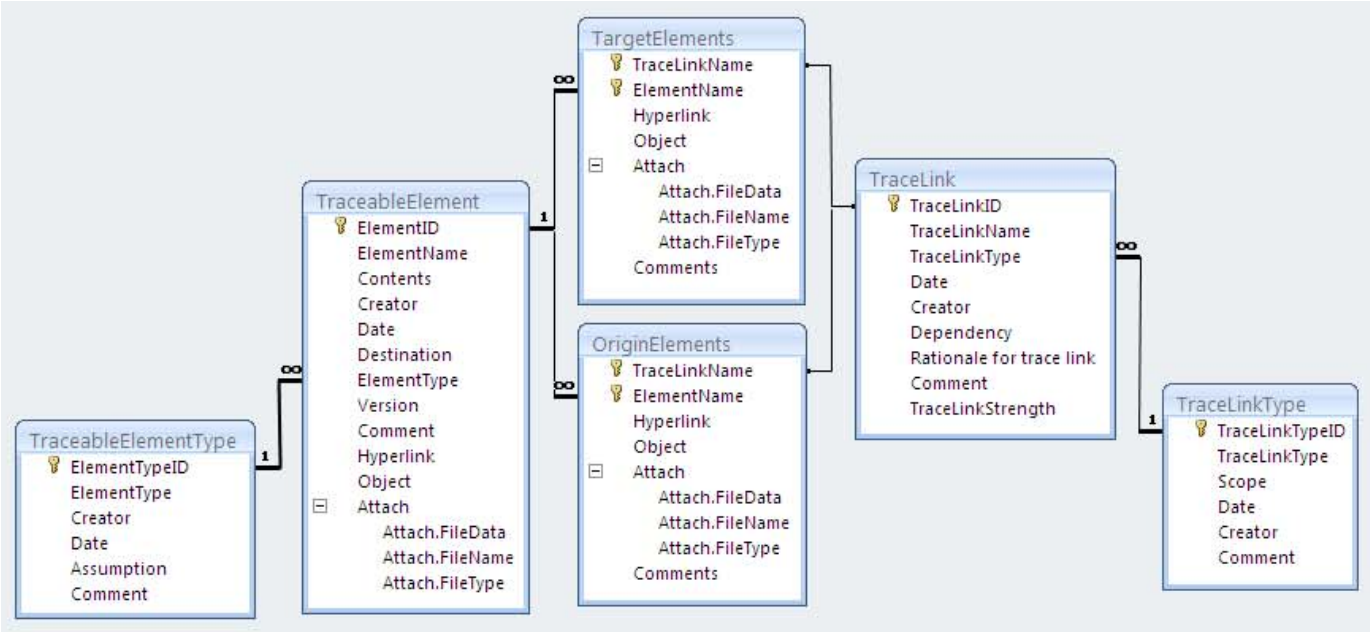


Figure 7. Entity-relationship diagram of the trace-link database of the prototype traceability tool

information, queries for retrieval of the trace info, a menu for managing work flow, forms for populating trace-link information, and facilities for reporting trace-links. A screen shot of the entity-relationship (ER) diagram of the trace-link database is shown by Figure 7. The ER diagram is normalized, which means that the data are organized with minimal needs for repeating the entries in the tables. Consistency checks are performed on the referenced fields. The data structure itself (represented by the ER diagram) does not cover all the constraints imposed by the meta-model (shown by Figure 4). However, constraints on queries and forms as well as macros can be added in order to fully implement the logic, such as for example which element types can be related to which trace-link types.

The five traceable element types defined by Figure 4 and their properties (name of creator, date, assumption and comment), are listed in Table *TraceableElementType*. Similarly, the six trace-link types defined by Figure 4 and their properties (scope, date, creator and comment), are listed in Table *TraceLinkType*. Table *TraceableElement* specifies the concrete instances of the traceable elements, and assigns properties (such as the pre-defined element type, hyperlink, creator, date, etc.) to each one of them. Since primary key attribute in Table *TraceableElementType* is foreign key in Table *TraceableElement*, multiplicity between the two respective tables is one-to-many.

Most of the properties are optional, and deduced based on: i) the core questions to be answered by traceability scheme [11] and ii) the needs for using guidelines for application of prediction models, specified in Section IV. The three Tables *TargetElements*, *OriginElements* and *TraceLink* together specify the concrete instances of trace-links. Each link is binary, and directed from a concrete pre-defined traceable element – the origin element specified in Table *OriginElements*, to a concrete pre-defined traceable element – the target element specified in Table *TargetElements*. The trace-link itself (between the origin and the target element) and its properties (such as pre-defined trace-link type) are specified in Table *TraceLink*. Attribute *TraceLinkName* (associated with a unique *TraceLinkId* value) connects the three tables *TraceLink*, *OriginElements* and *TargetElements* when representing a single trace-link instance, thus forming a cross-product when relating the three tables. The MS Access environment performs reference checks on the cross products, as well as on the values of the foreign key attributes. Target elements and origin elements participating

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

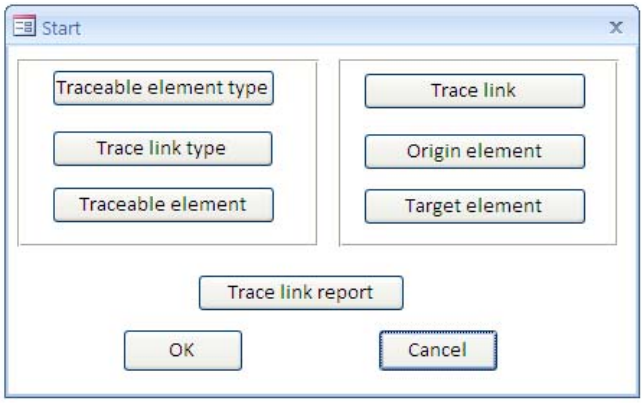


Figure 8. A screen shot of the start menu of the prototype traceability tool

in a trace-link, are instances of traceable elements defined in Table *TraceableElement*. They are connected through the Attribute *ElementId*. Note that in the Tables *OriginElements* and *TargetElements*, the Attribute *ElementId* has the role of a foreign key and is displayed as *ElementName*. In Tables *OriginElements* and *TargetElements*, the *ElementName* is through the *ElementId* retrieved from the Table *TraceableElement* and therefore exactly the same as the one in the table it originates from (i.e., *TraceableElement*). Thus, multiplicity between Table *TraceableElement* and Table *TargetElements*, as well as between Table *TraceableElement* and Table *OriginElements*, is one-to-many. Similarly, since primary key attribute in Table *TraceLinkType* is foreign key in Table *TraceLink*, multiplicity between the two respective tables is one-to-many.

A screen shot of the start menu is shown by Figure 8. The sequence of the buttons represents a typical sequence of actions of an end-user (the analyst), in the context of defining, documenting and using the trace-links. The basic definition of the types of the traceable elements and the trace-links are provided first. Then, concrete traceable elements are documented, before defining specific instances of the trace-links and their associated specific origin and target elements, involved in the binary trace-link relations. Finally, reports can be obtained, based on search parameters such as for example model types, model elements, or trace-link types.

H. Integrating the prototype traceability tool with the existing PREDIQT tool

In order to fully benefit from the traceability approach, the prototype traceability tool should be integrated with the existing PREDIQT tool. In addition, the traceability tool should be extended with the indicator templates and the above proposed visualization of the impacts. The traceability tool should moreover guide the user in the PREDIQT process and verify that the necessary prerequisites for each phase are fulfilled. The result should be seamless handling of the trace-link information in the traceability tool during the simultaneous development and use of DVs in the PREDIQT tool. Moreover, exchange of the trace-link information between the traceability tool and the PREDIQT tool, as well as a consolidated quality-cost-profit visualization of the decision alternatives in an integrated tool, is needed.

A preliminary result is exemplified in Figure 9, which shows a screen shot of the existing PREDIQT tool. The trace-link information is shown on demand. In this particular illustrative example with fictitious values, the user is evaluating the benefit of increasing the QCF of the root node by 0.006 (i.e., from 0.919 to 0.925). To this end, he is comparing cost of two possible alternatives: increase QCF of "Message Routing" by 0.04 (i.e., from 0.93 to 0.97), or increase of "Performance of the related services" by 0.025 (i.e., from 0.80 to 0.825). Both alternatives have the same impact on the root node QCF, but the cost of the measures (or treatments) related to achievement of the two alternatives, is different. Note that the cost information is a part of the trace-link information and not explicitly displayed on the DV shown in Figure 9. The integration of the traceability tool with the existing PREDIQT tool should therefore involve exchange of standardized messages regarding the trace-link information, functionality for running queries from the existing PREDIQT tool, and possibility of retrieving the prediction model elements (stored in the PREDIQT tool) from the traceability tool.

VIII. SUMMARY OF EXPERIENCES FROM APPLYING A PART OF THE SOLUTION ON PREDICTION MODELS FROM AN INDUSTRIAL CASE STUDY

This section reports on the results from applying our tool-supported traceability approach on prediction models, which were originally developed and applied during a PREDIQT-based analysis [5] on a real-life industrial system. The analysis targeted a system for managing validation of electronic certificates and signatures worldwide. The system analyzed was a so-called "Validation Authority" (VA) for evaluation of electronic identifiers (certificates and signatures) worldwide. In that case study, the prediction models were applied for simulation of impacts of 14 specified architecture design changes on the VA quality. Each specified architecture design change was first applied on the affected parts of the Design Model, followed by the conceptual model and finally the DVs. Some of the changes (e.g., change 1) addressed specific architecture design aspects, others referred to the system in general, while the overall changes (e.g., changes 6 through 14) addressed parameter specifications of the DVs. The specification suggested each change being independently applied on the approved prediction models.

The trace-link information was documented in the prototype traceability tool, in relation to the model development. The trace-links were applied during change application,

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

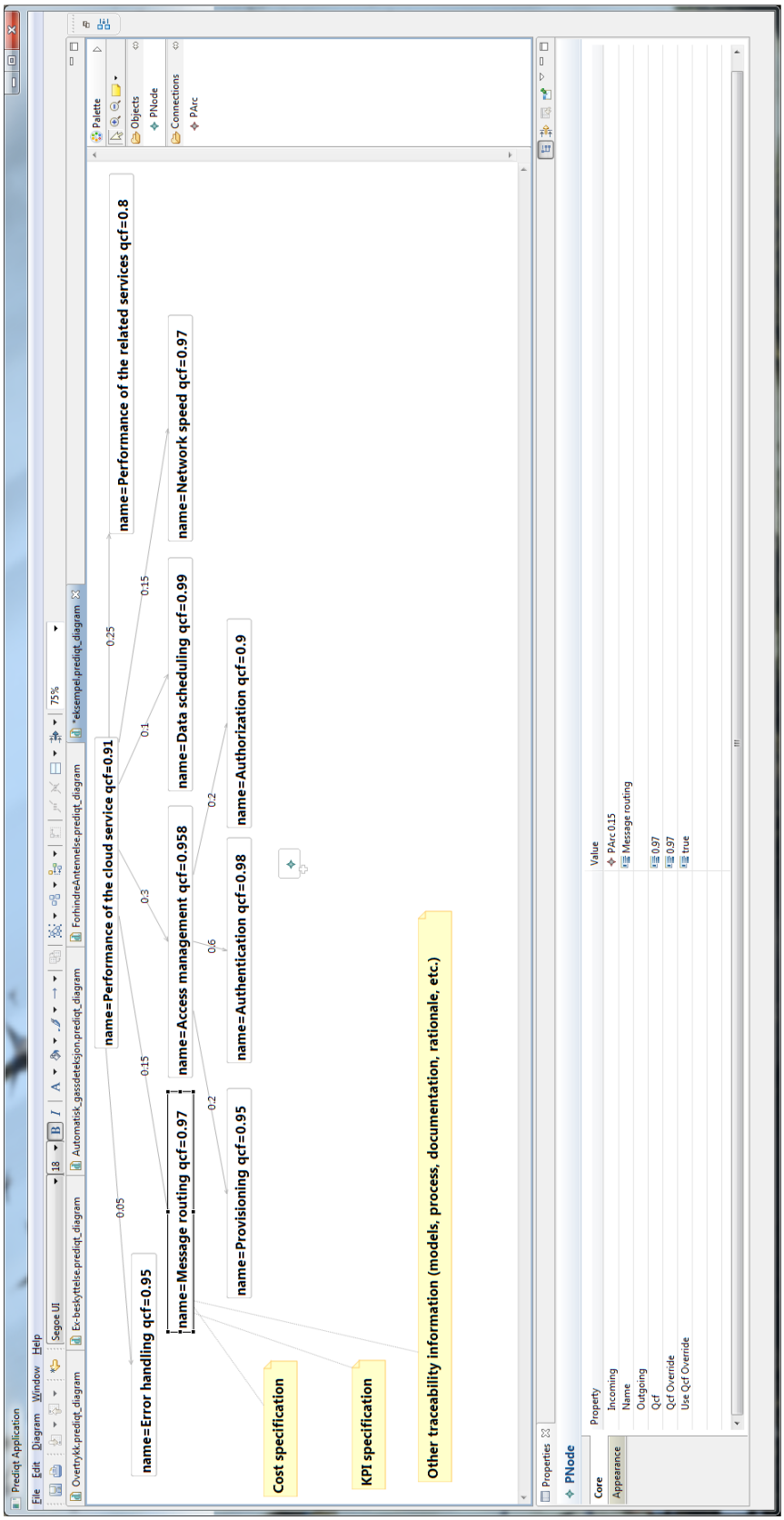


Figure 9. An illustrative example (with fictitious values) of displaying the trace-links in the PREDIQT tool

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Trace-link Report

Trace-link Type	Origin Element	Target Element	Trace-link Name
Design Model Element to Design Model Element			
	Signature Verification Comp-Interface	Signature Verification Comp-Interface	Signature Verification Comp-Interface
	Signature Verification Components	Signature Verification Components	Signature Verification Interface-Port
	Signature Verification Interface-Port	Signature Verification Interface-Port	VA Root Node Semantics

Figure 10. A screen shot of an extract of a trace-link report from the prototype traceability tool

according to the guidelines for application of prediction models, specified in Section IV. We present the experiences obtained, while the process of documentation of the trace-links is beyond the scope of this paper.

The prediction models involved are the ones related to "Split signature verification component into two redundant components, with load balancing", corresponding to Change 1 in Omerovic et al. [5]. Three Design Model diagrams were affected, and one, two and one model element on each, respectively. We have tried out the prototype traceability tool on the Design Model diagrams involved, as well as Availability (which was one of the three quality characteristics analyzed) related Quality Model diagrams and DV. Documentation of the trace-links involved within the Availability quality characteristic (as defined by the Quality Model) scope, took approximately three hours. Most of the time was spent on actually typing the names of the traceable elements and the trace-links.

18 instances of traceable elements were registered in the database during the trial: seven Quality Model elements, four DV elements, four Design Model elements and three elements of type "Rationale and Assumptions". 12 trace-links were recorded: three trace-links of type "Design Model Element to Design Model Element", three trace-links of type "Design Model Element to DV Element", one trace-link of type "Design Model Element to Rationale and Assumptions", three trace-links of type "DV Element to Quality Model Element", and two trace-links of type "Structure, Parameter or Semantics of DV Element Documented through Rationale and Assumptions", were documented.

An extract of a screen shot of a trace-link report (obtained from the prototype traceability tool) is shown by Figure 10. The report included: three out of three needed (i.e., actually existing, regardless if they are recorded in the trace-link database) "Design Model Element to Design Model Element" links, three out of four needed "Design Model Element to DV Element" links, one out of one needed "Design Model Element to Rationale and Assumptions" link, three out of six needed "DV Element to Quality Model Element" links and one out of one needed "Structure, Parameter or Semantics of DV Element Documented through Rationale and Assumptions" link.

Best effort was made to document the appropriate trace-links without taking into consideration any knowledge of exactly which of them would be used when applying the change. The use of the trace-links along with the application of change on the prediction models took totally 20 minutes and resulted in the same predictions (change propagation paths and values of QCF estimates on the Availability DV), as in the original case study [5]. Without the guidelines and the trace-link report, the change application would have taken approximately double that time for the same user.

All documented trace-links were relevant and used during the application of the change, and about 73% of the relevant trace-links could be retrieved from the prototype traceability tool. Considering however the importance and the role of the retrievable trace-links, the percentage should increase considerably.

Although hyperlinks are included as meta-data in the user interface for element registration, an improved solution should include interfaces for automatic import of the element names from the prediction models, as well as user interfaces for easy (graphical) trace-link generations between the existing elements. This would also aid verification of the element names.

IX. Why our solution is a good one

This section argues that the approach presented above fulfills the success criteria specified in Section V.

A. Success Criterion 1

The traceability scheme and the prototype traceability tool capture the kinds of trace-links and traceable elements, specified in the Success Criterion 1. The types of trace-links and traceable elements as well as their properties, are specified in dedicated tables in the database of the prototype traceability tool. This allows constraining the types of the trace-links and the types of the traceable elements to only the ones defined, or extending their number or definitions, if needed. The trace-links in the prototype traceability tool are binary and unidirectional, as required by the traceability scheme. Macros and constraints can be added in the tool, to implement any additional logic regarding trace-links, traceable elements, or their respective type definitions and relations. The data properties (e.g., date, hyperlink, or creator) required by the user interface, allow full traceability of the data registered in the database of the prototype traceability tool.

B. Success Criterion 2

Searching based on user input, selectable values from a list of pre-defined parameters, or comparison of one or more database fields, are relatively simple and fully supported

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

based on queries in MS Access. Customized reports can be produced with results of any query and show any information registered in the database. The report, an extract of which is presented in Section VIII, is based on a query of all documented trace-links and the related elements.

C. Success Criterion 3

The text-based fields for documenting the concrete instances of the traceable elements and the trace-links, allow level of detail selectable by the user. Only a subset of fields is mandatory for providing the necessary trace-link data. The optional fields in the tables can be used for providing additional information such as for example rationale, comments, links to external information sources, attachments, strength or dependency. There are no restrictions as to what can be considered as a traceable element, as long at it belongs to one of the element types defined by Figure 4. Similarly, there are no restrictions as to what can be considered as a trace-link, as long at it belongs to one of the trace-link types defined by Figure 4. The amount of information provided regarding the naming and the meta-data, are selectable by the user.

D. Success Criterion 4

As argued, the models and the change specification originate from a real-life industrial case study in which PREDIQT was entirely applied on a comprehensive system for managing validation of electronic certificates and signatures worldwide (a so-called "Validation Authority"). Several essential aspects characterize the application of the approach presented in Section VIII:

- the realism of the prediction models involved in the example
- the size and complexity of the target system addressed by the prediction models
- the representativeness of the change applied to the prediction models
- the simplicity of the prototype traceability tool with respect to both the user interfaces and the notions involved
- the time spent on documenting and using the trace-links

Overall, these aspects indicate the applicability of our solution on real-life applications of PREDIQT, with limited resources and by an average user (in the role of the analyst).

The predictions (change propagation paths and values of QCF estimates) we obtained during the application of our solution on the example were the same as the ones from the original case study [5] (performed in year 2008), which the models stem from. Although the same analyst has been involved in both, the results (i.e., the fact that the same predictions were obtained in both trials in spite of a rather long time span between them) suggest that other users should, by following PREDIQT guidelines and applying the prototype traceability tool, obtain similar results. The process of application of the models has been documented in a structured form, so that the outcome of the use of the prediction models is as little as possible dependent on the analyst performing the actions. Hence, provided the fact that the guidelines are followed, the outcome should be comparable if re-applying the overall changes from the original case study.

The time spent is to some degree individual and depends on the understanding of the target system, the models and the PREDIQT method. It is unknown if the predictions would have been the same (as in the original case study) for another user. We do however consider the models and the change applied during the application of the solution, to be representative due to their origins from a major real-life system. Still, practical applicability of our solution will be subject to future empirical evaluations.

X. Why other approaches are not better in this context

This section evaluates the feasibility of other traceability approaches in the PREDIQT context. Based on our review of the approach-specific publications and the results of the evaluation by Galvao and Goknil [12] of a subset of the below mentioned approaches, we argue why the alternative traceability approaches do not perform sufficiently on one or more of the success criteria specified in Section V. The evaluation by Galvao and Goknil is conducted with respect to five criteria: 1) structures used for representing the traceability information; 2) mapping of model elements at different abstraction levels; 3) scalability for large projects in terms of process, visualization of trace information, and application to a large amount of model elements; 4) change impact analysis on the entire system and across the software development life cycle; and 5) tool support for visualization and management of traces, as well as for reasoning on the trace-link information.

Almeida et al. [25] propose an approach aimed at simplifying the management of relationships between requirements and various design artifacts. A framework which serves as a basis for tracing requirements, assessing the quality of model transformation specifications, meta-models, models and realizations, is proposed. They use traceability cross-tables for representing relationships between application requirements and models. Cross-tables are also applied for considering different model granularities and identification of conforming transformation specifications. The approach does not provide sufficient support for intra-model mapping, thus failing on our Success Criterion 1. Moreover, possibility of representing the various types of trace-links and traceable elements is unclear, although different visualizations on a cross-table are suggested. Tool support is not available, which limits applicability of the approach in a practical setting. Searching and reporting facilities are not available. Thus, it fails on our Success Criteria 1, 2, and 4.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Event-based Traceability (EBT) is another requirements-driven traceability approach aimed at automating trace-link generation and maintenance. Cleland-Huang, Chang and Christensen [26] present a study which uses EBT for managing evolutionary change. They link requirements and other traceable elements, such as design models, through publish-subscribe relationships. As outlined by Galvao and Goknil [12], "Instead of establishing direct and tight coupled links between requirements and dependent entities, links are established through an event service. First, all artefacts are registered to the event server by their subscriber manager. The requirements manager uses its event recognition algorithm to handle the updates in the requirements document and to publish these changes as event to the event server. The event server manages some links between the requirement and its dependent artefacts by using some information retrieval algorithms." The notification of events carries structural and semantic information concerning a change context. Scalability in a practical setting is the main issue, due to performance limitation of the EBT server [12]. Moreover, the approach does not provide sufficient support for intra-model mapping. Thus, it fails on our Success Criteria 1 and 4.

Cleland-Huang et al. [27] propose the Goal Centric Traceability (GCT) approach for managing the impact of change upon the non-functional requirements of a software system. A Softgoal Interdependency Graph (SIG) is used to model non-functional requirements and their dependencies. Additionally, a traceability matrix is constructed to relate SIG elements to classes. The main weakness of the approach is the limited tool support, which requires manual work. This limits both scalability in a practical setting and searching support (thus failing on our Success Criteria 4 and 2, respectively). It is unclear to what degree the granularity of the approach would meet the needs of PREDIQT.

Cleland-Huang and Schmelzer [28] propose another requirements-driven traceability approach that builds on EBT. The approach involves a different process for dynamically tracing non-functional requirements to design patterns. Although more fine grained than EBT, there is no evidence that the method can be applied with success in a practical real-life setting (required through our Success Criterion 4). Searching and reporting facilities (as required through our Success Criterion 2) are not provided.

Many traceability approaches address trace maintenance. Cleland-Huang, Chang, and Ge [29] identify the various change events that occur during requirements evolution and describe an algorithm to support their automated recognition through the monitoring of more primitive actions made by a user upon a requirements set. Mäder and Gotel [30] propose an approach to recognize changes to structural UML models that impact existing traceability relations and, based on that knowledge, provide a mix of automated and semi-automated strategies to update the relations. Both approaches focus on trace maintenance, which is as argued in Section V, not among the traceability needs in PREDIQT.

Ramesh and Jarke [16] propose another requirements-driven traceability approach where reference models are used to represent different levels of traceability information and links. The granularity of the representation of traces depends on the expectations of the stakeholders [12]. The reference models can be implemented in distinct ways when managing the traceability information. As reported by Galvao and Goknil [12], "The reference models may be scalable due to their possible use for traceability activities in different complexity levels. Therefore, it is unclear whether this approach lacks scalability with respect to tool support for large-scale projects or not. The efficiency of the tools which have implemented these meta-models was not evaluated and the tools are not the focus of the approach." In PREDIQT context, the reference models are too broad, their focus is on requirements traceability, and tool support is not sufficient with respect to searching and reporting (our Success Criterion 2).

We could however have tried to use parts of the reference models by Ramesh and Jarke [16] and provide tool support based on them. This is done by Mohan and Ramesh [31] in the context of product and service families. The authors discuss a knowledge management system, which is based on the traceability framework by Ramesh and Jarke [16]. The system captures the various design decisions associated with service family development. The system also traces commonality and variability in customer requirements to their corresponding design artifacts. The tool support has graphical interfaces for documenting decisions. The trace and design decision capture is illustrated using sample scenarios from a case study. We have however not been able to obtain the tool, in order to try it out in our context.

A modeling approach by Egyed [32] represents traceability information in a graph structure called a footprint graph. Generated traces can relate model elements with other models, test scenarios or classes [12]. Galvao and Goknil [12] report on promising scalability of the approach. It is however unclear to what degree the tool support fulfills our success criterion regarding searching and reporting, since semantic information on trace-links and traceable elements is limited.

Aizenbud-Reshef et al. [33] outline an operational semantics of traceability relationships that capture and represent traceability information by using a set of semantic properties, composed of events, conditions and actions [12]. Galvao and Goknil [12] state: the approach does not provide sufficient support for intra-model mapping; a practical application of the approach is not presented; tool support is not provided; however, it may be scalable since it is associated with the UML. Hence, it fails on our Success Criteria 1 and 2.

Limon and Garbajosa [34] analyze several traceability

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

schemes and propose an initial approach to Traceability Scheme (TS) specification. The TS is composed of a traceability link dataset, a traceability link type set, a minimal set of traceability links, and a metrics set for the minimal set of traceability links [12]. Galvao and Goknil [12] argue that "The TS is not scalable in its current form. Therefore, the authors outline a strategy that may contribute to its scalability: to include in the traceability schema a set of metrics that can be applied for monitoring and verifying the correctness of traces and their management." Hence, it fails with respect to scalability in a practical setting, that is, our criterion 4. Moreover, there is no tool support for the employment of the approach, which fails on our success criterion regarding searching and reporting.

Some approaches [35] [36] [37] that use model transformations can be considered as a mechanism to generate trace-links. Tool support with transformation functionalities is in focus, while empirical evidence of applicability and particularly comprehensibility of the approaches in a practical setting, is missing. The publications we have retrieved do not report sufficiently on whether these approaches would offer the searching facilities, the granularity of trace information, and the scalability needed for use in PREDIQT context (that is, in a practical setting by an end-user (analyst) who is not an expert in the tools provided).

XI. Conclusion and future work

Our earlier research indicates the feasibility of the PREDIQT method for model-based prediction of impacts of architectural design changes on system quality. The PREDIQT method produces and applies a multi-layer model structure, called prediction models, which represent system design, system quality and the interrelationship between the two.

Based on the success criteria for a traceability approach in the PREDIQT context, we put forward a traceability scheme. Based on this, a solution supported by a prototype traceability tool is developed. The prototype tool can be used to define, document, search for and represent the trace-links needed. We have argued that our solution offers a useful and practically applicable support for traceability handling in the PREDIQT context. The model application guidelines provided in Section IV complement the prototype traceability tool and aim to jointly provide the facilities needed for a schematic application of prediction models.

Performing an analysis of factors such as cost, risk, and benefit of the trace-links themselves and following the paradigm of value-based software engineering, would be relevant in order to stress the effort on the important trace-links. As argued by Winkler and von Pilgrim [11], if the value-based paradigm is applied to traceability, cost, benefit, and risk will have to be determined separately for each trace according to if, when, and to what level of detail it will be needed later. This leads to more important artifacts having higher-quality traceability. There is a trade-off between the semantically accurate techniques on the one hand and cost-efficient but less detailed approaches on the other hand. Finding an optimal compromise is still a research challenge. Our solution proposes a feasible approach, while finding the optimal one is subject to further research.

PREDIQT has only architectural design as the independent variable – the Quality Model itself is, once developed, assumed to remain unchanged. This is of course a simplification, since quality characteristic definitions may vary in practice. It would be interesting to support variation of the Quality Model as well, in PREDIQT.

Development of an experience factory, that is, a repository of the non-confidential and generalizable experiences and models from earlier analyses, is another direction for future work. An experience factory from similar domains and contexts would allow reuse of parts of the prediction models and potentially increase model quality as well as reduce the resources needed for a PREDIQT-based analysis.

Further empirical evaluation of our solution is also necessary to test its feasibility on different analysts as well as its practical applicability in the various domains which PREDIQT is applied on. Future work should also include integration of the PREDIQT tool with the traceability tool. Particularly important is development of standard interfaces and procedures for updating the traceable elements from the prediction models into our prototype traceability tool.

As model application phase of PREDIQT dictates which trace-link information is needed and how it should be used, the current PREDIQT guidelines focus on the application of the prediction models. However, since the group of recorders and the group of users of traces may be distinct, structured guidelines for recording the traces during the model development should also be developed as a part of the future work.

Acknowledgment

This work has been conducted as a part of the DIGIT (180052/S10) project funded by the Research Council of Norway, as well as a part of the NESSoS network of excellence funded by the European Commission within the 7th Framework Programme.

References

[1] A. Omerovic and K. Stølen, "Traceability Handling in Model-based Prediction of System Quality," in *Proceedings of Third International Conference on Advances in System Simulation, SIMUL 2011*. IARIA, 2011, pp. 71–80.

[2] A. Omerovic, A. Andresen, H. Grindheim, P. Myrseth, A. Refsdal, K. Stølen, and J. Ølnes, "A Feasibility Study in Model Based Prediction of Impact of Changes on System Quality," in *International Symposium on Engineering Secure Software and Systems*, vol. LNCS 5965. Springer, 2010, pp. 231–240.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[3] A. Omerovic, B. Solhaug, and K. Stølen, "Evaluation of Experiences from Applying the PREDIQT Method in an Industrial Case Study," in *Fifth IEEE International Conference on Secure Software Integration and Reliability Improvement*. IEEE, 2011, pp. 137–146.

[4] A. Omerovic, *PREDIQT: A Method for Model-based Prediction of Impacts of Architectural Design Changes on System Quality*. PhD thesis, Faculty of Mathematics and Natural Sciences, University of Oslo, 2012.

[5] A. Omerovic, A. Andresen, H. Grindheim, P. Myrseth, A. Refsdal, K. Stølen, and J. Ølnes, "A Feasibility Study in Model Based Prediction of Impact of Changes on System Quality," SINTEF, Tech. Rep. A13339, 2010.

[6] A. Omerovic and K. Stølen, "Traceability Handling in Model-based Prediction of System Quality," SINTEF, Tech. Rep. A19348, 2011.

[7] A. Knethen and B. Paech, "A Survey on Tracing Approaches in Practice and Research," Frauenhofer IESE, Tech. Rep. 095.01/E, 2002.

[8] "Standard Glossary of Software Engineering Terminology: IEEE Std.610. 12-1990," 1990.

[9] N. Aizenbud-Reshef, B. T. Nolan, J. Rubin, and Y. Shaham-Gafni, "Model Traceability," *IBM Syst. J.*, vol. 45, no. 3, pp. 515–526, 2006.

[10] J. Simpson and E. Weiner, *Oxford English Dictionary*. Clarendon Press, 1989, vol. 18, 2nd edn.

[11] S. Winkler and J. von Pilgrim, "A survey of Traceability in Requirements Engineering and Model-driven Development," *Software and Systems Modeling*, vol. 9, no. 4, pp. 529–565, 2010.

[12] I. Galvao and A. Goknil, "Survey of Traceability Approaches in Model-Driven Engineering," in *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference*, 2007.

[13] G. Spanoudakis and A. Zisman, "Software Traceability: A Roadmap," in *Handbook of Software Engineering and Knowledge Engineering*. World Scientific Publishing, 2004, pp. 395–428.

[14] R. J. Wieringa, "An Introduction to Requirements Traceability," Faculty of Mathematics and Computer Science, Vrije Universiteit, Tech. Rep. IR-389, 1995.

[15] N. Anquetil, U. Kulesza, R. Mitschke, A. Moreira, J.-C. Royer, A. Rummler, and A. Sousa, "A Model-driven Traceability Framework for Software Product Lines," *Software and Systems Modeling*, 2009.

[16] B. Ramesh and M. Jarke, "Toward Reference Models for Requirements Traceability," *IEEE Transactions on Software Engineering*, vol. 27, no. 1, pp. 58–93, 2001.

[17] S. Bohner and R. Arnold, *Software Change Impact Analysis*. IEEE Computer Society Press, 1996.

[18] "International Organisation for Standardisation: ISO/IEC 9126 - Software Engineering – Product Quality," 2004.

[19] I. Refsdal, *Comparison of GMF and Graphiti Based on Experiences from the Development of the PREDIQT Tool*. University of Oslo, 2011.

[20] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified Modeling Language Reference Manual*. Pearson Higher Education, 2004.

[21] A. Omerovic and K. Stølen, "A Practical Approach to Uncertainty Handling and Estimate Acquisition in Model-based Prediction of System Quality," *International Journal on Advances in Systems and Measurements*, vol. 4, no. 1-2, pp. 55–70, 2011.

[22] A. Omerovic and K. Solhaug, B. Stølen, "Assessing Practical Usefulness and Performance of the PREDIQT Method: An Industrial Case Study," *Information and Software Technology*, vol. 54, pp. 1377–1395, 2012.

[23] A. Omerovic and K. Stølen, "Interval-Based Uncertainty Handling in Model-Based Prediction of System Quality," in *Proceedings of Second International Conference on Advances in System Simulation, SIMUL 2010*, August 2010, pp. 99–108.

[24] "Access Help and How-to," accessed: May 19, 2011. [Online]. Available: http://office.microsoft.com/en-us/access-help/

[25] J. P. Almeida, P. v. Eck, and M.-E. Iacob, "Requirements Traceability and Transformation Conformance in Model-Driven Development," in *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference*, 2006, pp. 355–366.

[26] J. Cleland-Huang, C. K. Chang, and M. Christensen, "Event-Based Traceability for Managing Evolutionary Change," *IEEE Trans. Softw. Eng.*, vol. 29, pp. 796–810, 2003.

[27] J. Cleland-Huang, R. Settimi, O. BenKhadra, E. Berezhanskaya, and S. Christina, "Goal-centric Traceability for Managing Non-functional Requirements," in *Proceedings of the 27th International Conference on Software Engineering*. ACM, 2005, pp. 362–371.

[28] J. Cleland-Huang and D. Schmelzer, "Dynamically Tracing Non-Functional Requirements through Design Pattern Invariants," in *Proceedings of the 2nd International Workshop on Traceability in Emerging Forms of Software Engineering*. ACM, 2003.

[29] J. Cleland-Huang, C. K. Chang, and Y. Ge, "Supporting Event Based Traceability through High-Level Recognition of Change Events," in *26th Annual International Computer Software and Applications Conference*. IEEE Computer Society, 2002, pp. 595–600.

[30] P. Mäder, O. Gotel, and I. Philippow, "Enabling Automated Traceability Maintenance through the Upkeep of Traceability Relations," in *Proceedings of the 5th European Conference on Model Driven Architecture - Foundations and Applications*. Springer-Verlag, 2009, pp. 174–189.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[31] K. Mohan and B. Ramesh, "Managing Variability with Traceability in Product and Service Families." IEEE Computer Society, 2002, pp. 1309–1317.

[32] A. Egyed, "A Scenario-Driven Approach to Trace Dependency Analysis," *IEEE Transactions on Software Engineering*, vol. 29, no. 2, pp. 116–132, 2003.

[33] N. Aizenbud-Reshef, R. F. Paige, J. Rubin, Y. Shaham-Gafni, and D. S. Kolovos, "Operational Semantics for Traceability," in *Proceedings of the ECMDA Traceability Workshop, at European Conference on Model Driven Architecture*, 2005, pp. 7–14.

[34] A. E. Limon and J. Garbajosa, "The Need for a Unifying Traceability Scheme," in *2nd ECMDA-Traceability Workshop*, 2005, pp. 47–55.

[35] F. Jouault, "Loosely Coupled Traceability for ATL," in *In Proceedings of the European Conference on Model Driven Architecture (ECMDA) workshop on traceability*, 2005, pp. 29–37.

[36] D. S. Kolovos, R. F. Paige, and F. Polack, "Merging Models with the Epsilon Merging Language (EML)," in *MoDELS'06*, 2006, pp. 215–229.

[37] J. Falleri, M. Huchard, and C. Nebut, "Towards a Traceability Framework for Model Transformations in Kermeta," in *Proceedings of the ECMDA Traceability Workshop, at European Conference on Model Driven Architecture*, 2006, pp. 31–40.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Augmented Reality Visualization of Numerical Simulations in Urban Environments

Sebastian Ritterbusch, Staffan Ronnås, Irina Waltschläger, Philipp Gerstner, and Vincent Heuveline
Engineering Mathematics and Computing Lab (EMCL)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
{sebastian.ritterbusch, staffan.ronnas, vincent.heuveline}@kit.edu, {irina.waltschlaeger, philipp.gerstner}@student.kit.edu

***Abstract*—Visualizations of large simulations are not only computationally intensive but also difficult for the viewer to interpret, due to the huge amount of data to be processed. In this work, we present a novel Augmented Reality visualization method, which enables simulations based on current city model data to be presented with localized real-world images. Test scenarios of urban wind flow and fine dust simulations illustrate the benefits of mobile Augmented Reality visualizations, both in terms of selection of data relevant to the user and facilitation of comprehensible access to simulation results.**

Keywords-Scientific Visualization, Augmented Reality, Numerical Simulation, Urban Airflow, Geographical Information Systems.

I. Introduction

Numerical simulation and interactive 3D visualization has today become an essential tool in many applications, including industrial design, studies of the environment and meteorology, and medical engineering. The increasing performance of computers has played an important role for the applicability of numerical simulation but has also led to a rapid growth in the amount of data to be processed. At present, the use of simulation software and the interpretation of visualization results usually require dedicated expertise. The large amount of data available leads to two problems for the end-user, which are discussed in this paper extending [1]. On the one hand, handling and selection of the appropriate data requires a suitable user interface. On the other hand, the amount of perceptible information is limited, and thus visualizations of large data sets need very intuitive methods to be understandable.

The use of Augmented Reality (AR) is aiming at the extension of human senses for delivering contextual information in an optimized way [2], [3]. For the visual sense, a difference of traditional imaging of virtual information to augmented imaging is the direct correspondence of virtual objects to reality. By exploitation of this additional and seamless information channel, the quality of information representation is strongly enhanced. This generally improves the analysis and comprehension of virtual data, but also opens new aspects for validation. This is especially true for AR visualizations of numerical simulations in living environments, where a manual comparison of results in the form of a visualization in a virtual world with reality may be tedious, and even misleading for an uninformed viewer. For instance, we make use of higher-order elements or artificial boundary conditions to better represent reality [4], [5], but for which highly specialized visualization methods would be needed to represent the data in its full fidelity [6]. When representing the results in the context of reality, the evaluation of the chosen model is simplified, and the results are represented more appropriately in the actual surroundings instead of an arbitrarily complex model thereof.

Numerical simulations in many domains can benefit from AR visualizations. Besides analysis of urban airflow and a forecast of fine dust distribution as presented in this paper, examples include noise propagation [7], urban climate simulation [8], and human crowd simulation [9]. The general feasibility of simulations in living environments and AR visualization was strongly promoted by the introduction and increasing role of Geographical Information Systems (GIS) for urban planning [10]. Their improved accuracy joined with the increasing performance of computing systems are making accurate large scale urban simulations feasible. We present the results of the joint work with the city council of Karlsruhe for simulations in an urban environment as an illustrative example setting, with focus on the advantages of mobile AR visualization of large numerical simulations. The proposed visualization method, whose development started with the *Science to Go* project, serves as a technology for solving problems of large scale data visualizations. Additionally, it also opens the path to making results of numerical simulations accessible to decision makers and to the citizen at large, both from the technical and the comprehension perspective. The general availability of smartphones and tablets equipped with GPS, cameras and graphical capabilities fulfills the technical requirements on the client side for implementing the presented visualization method. This allows for an intuitive exploration of large scale simulations. The ongoing standardization process of GIS for city modeling in the CityGML consortium [11] enables standardized simulation and visualization services for world-wide use

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

based on the presented method in the future.

This work is an extension of [1] with a more in-depth description and discussion of the method, the application to a new scenario and simulations, as well as a description of further research into solutions for accurate visual alignment of AR visualizations using active markers.

In this paper, we first present previous papers and projects, which relate to the proposed concept. This is followed by a description of the visualization method, with details on the needed steps of pre-processing, simulation, AR visualization, interaction, and the client-server framework. The text ends with the conclusion and acknowledgments of partners and funding for the project.

II. Related Work

The Touring machine [12] was one of the first mobile solutions for AR illustrating the potential of enhancing real life images in real-time for exploration of the urban environment. The approach was to display information overlays on the camera image, which is still popular in AR applications of today [13], [14]. This concept is well suited to presenting textual or illustrative information, such as designation of points of interest, or augmented objects on top of printed markers. But this does not directly apply to immersive AR visualization of simulation results in the living environment around the viewer as presented in this paper.

The availability of dedicated graphical processing units on mobile devices has led to AR visualizations of pre-defined 3D objects [15], which have been found beneficial in laboratory setups [16]. This is the basis for visualization of 3D structures representing the results of simulations. The use of AR visualization for environmental data is presented in the HYDROSYS framework [17], which provides a method to combine measurements and simulation data with geographic information. Similar to the work presented in this paper, that framework emphasizes the need for simulation information on-site. The conceptional need for combining simulation results with data from geographic information systems is also a driving force for the CityGML project [10], which has applications to natural disaster management.

AR visualization of urban air flow phenomena in an indoor virtual reality laboratory setting based on physical mock-up building blocks is presented in [18]. The general aim of that work is similar to the one presented here, but it is focused on the interaction with objects in the visualization, and does not treat the aspect of remote visualization on mobile devices.

A related domain is that of map generation through interpolation of geographically localized, sparse data. A sophisticated algorithm for this type of problem is proposed in [19], which could conceivably also be used as a source of data for the visualization method presented in this work. In the applications presented here, the focus is on the use of data obtained through numerical simulation.

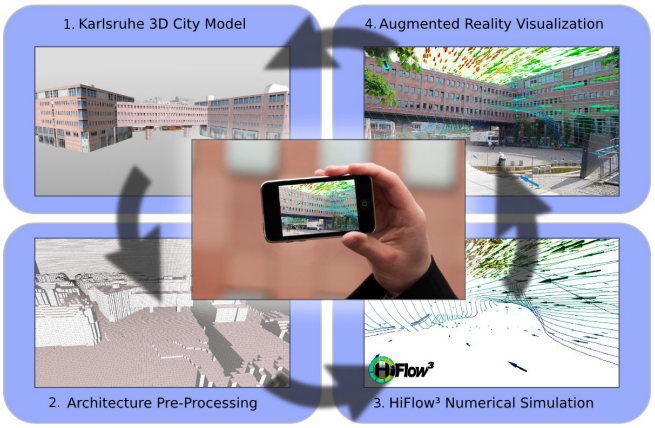


Figure 1. Augmented Reality simulation and visualization workflow.

The simulations that are presented in this work concern wind flow and particle distribution in urban environments. This setting has previously been investigated in several works, including [20] and [21]. In contrast to those papers, we employ a simplified model, which does not include the effects of wind turbulence. This reduces the computational costs, while still delivering results that serve to illustrate the potential of the AR visualization method. It is also advantageous in cases where the outcome of numerical simulations has to be related to the real surroundings, such as for the placement of mini wind turbines in urban spaces, which does not only depend on the optimal wind conditions as discussed in [22] and [23], but also their fit into the city scape.

III. Visualization Method

The problem of creating AR visualizations of scientific data is demanding in several aspects, and its solution must necessarily combine a range of techniques from different fields, including geometric modeling, numerical simulation, computer graphics and network programming, as illustrated in Figure 1. In this section, we describe the method that we have developed to achieve this goal. First, we outline the problems that were identified in the early phases of development. Next, we describe two scenarios, which are used to illustrate the use of the method. In the remainder of the section, we provide details on various aspects of the techniques that were used, including the construction and discretization of a virtual geometry, modeling and numerical simulation, AR visualization, interface for user interaction, and a framework for distribution of the compute load.

A. Identification of Problems

To obtain a clear understanding of the steps required to create AR visualizations of scientific data, we have identified and analyzed the main problems associated with this task. As with any AR implementation, the first challenge is to construct a virtual geometry. In this work, we have focused

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

on use cases in an exterior urban setting, but the proposed concept could also be applied in large open areas as well as inside buildings.

The next challenge is to create datasets that are suitable to visualize in the AR rendering. In this work, we are interested in displaying solutions of numerical simulations of physical phenomena, such as wind flow, noise, temperature or particle concentrations. The process to compute these solutions is largely manual: one has to determine a suitable mathematical model, formulate a precise and well-posed problem, choose an appropriate numerical method, and perform discretizations of the equations as well as the geometry. Furthermore, one must acquire the necessary input data such as material properties, boundary values and initial conditions. Ideally, all these steps would be automated, but at the present state of research, at least the steps up to and including the discretization require some human intervention.

Once a dataset has been computed for the virtual geometry, one has to combine it with the real-world geometry, based on the position and orientation of the user. The major difficulties in this context are the alignment of the virtual and real geometries, and the combination of the computed dataset and the current camera view.

AR visualization is by nature interactive, and should permit the user to control the displayed data in various ways, not only by moving the camera. Furthermore, it is not always evident how visualizations of scientific data, and its associated uncertainties, should be interpreted. An important challenge is how to present data in such a way that it can be correctly understood by non-experts.

The final problem that we identified is the need for substantial compute power, both for the numerical simulation and for visualization of the results. Although the capabilities of handheld devices is steadily increasing, the processor within a single mobile phone is not able to solve three-dimensional fluid flow problems with reasonable accuracy within acceptable time limits. Hence, a distributed architecture is needed, which allows remote access to numerical simulations on powerful hardware.

The method proposed in this work is an attempt to address all these problems. We discuss the extent to which we consider our solution successful, as well as the open problems that remain, in Sections IV and V.

B. Scenarios

In order to demonstrate the capabilities of our visualization method, we define two test scenarios, each consisting of a specific numerical simulation in a specified place. Figure 2 shows the location of these sites on a map of the city of Karlsruhe. These scenarios are primarily meant to illustrate how AR visualizations of scientific data are useful, and to provide datasets upon which the various data processing and visualization techniques can be tested. The accurate simulation of the physical processes that we have chosen is generally a difficult and time-consuming problem, which is not the main focus of this work. For this reason, the models have been simplified, and the input parameters has been chosen in such a way as to make it possible to obtain the data in a short amount of time, at the expense of accuracy and physical realism of the results. The computations performed and the simplifications that were made, are described in detail later in this section.

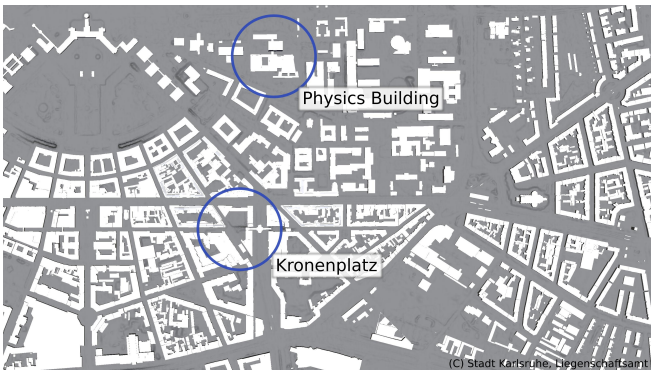


Figure 2. Map of Karlsruhe with places corresponding to scenarios.

The first scenario that we consider is wind simulation around the building that hosts the Department of Mathematics of the Karlsruhe Institute of Technology (KIT). It is located at the Kronenplatz square in Karlsruhe. We use synthetic data to determine a plausible wind velocity flow on the boundary of the domain, and solve the incompressible Navier-Stokes equations to obtain the solution in the entire domain.

The second scenario concerns the spread of fine dust particles in the vicinity of the Physics building on the campus of KIT. In a first step, we again compute a velocity field around the buildings as in the first scenario; and then solve a model for the transport of microscopic particles suspended in the air based on this velocity field.

C. Virtual Geometry

A numerical simulation can be viewed as the combination of a mathematical description of the physical phenomenon to be simulated, a numerical method to solve the problem, and a computational domain describing the space in which the simulation is performed. While the first two aspects are discussed in literature, and actively researched in computational sciences, the third aspect traditionally receives less attention for living environments. Understandably, this is due to the fact, that the effort of performing measurements of buildings is too large compared to the value of individual numerical simulations. Furthermore, the alignment with real world coordinates as needed for AR applications is an additional requirement. A solution to this problem is to derive the computational domain from other data sources, performing additional steps to convert the geometrical description to a

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Figure 3. Photo-realistic building in the Karlsruhe 3D city model.

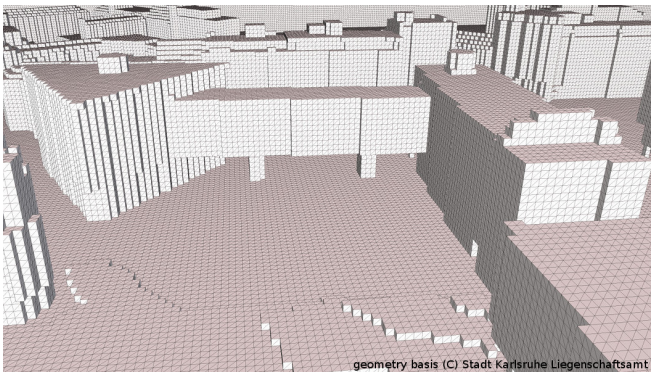


Figure 4. Computational geometry based on the Karlsruhe 3D City Model.

suitable computational domain. This approach is followed and explained in this text, based on a GIS urban model.

The project "3D-Stadtmodell Karlsruhe" [24] was started in 2002 as an improved database of geographic information to meet the demands of the local administration. It consists of several data sets of varying purpose, coverage, accuracy and detail, starting with a terrain model without buildings, and including large brick models for the cityscape, up to a photo-realistic model, as seen in Figure 3. All data sets are expressed in a global Cartesian coordinate system, such as Gauß-Krüger or Universal Transverse Mercator (UTM) coordinates, for alignment with the real world. The city model is currently progressing towards an integration into a CityGML [10] based representation.

Since none of the models were created for use by numerical simulation software, extensive pre-processing steps were necessary. In general, two or three models have to be combined to create a suitable computational domain, as seen in Figure 4. Special care was necessary to deal with model enhancements that had been made mainly for visual effects. For instance, there were closed window panes in garages facing the outside world on both sides with zero width, which are very significant for wind flow simulations around buildings. Although such irregularities could be avoided by imposing strict conditions on the city models, in general we cannot expect available city models to conform to these conditions, since they were originally created for visual planning. To avoid problems arising from these kinds of artifacts, an emphasis was put on the use of robust and efficient region growing methods that are well known from medical applications such as the realistic computational fluid dynamics simulations of the nose and lungs (see, e.g., [25], [26]).

The chosen approach approximates the geometry by discretization into voxels of pre-defined size. On the one hand, this avoids problems around very small details, that would require a high level of detail in the computational domain. This would lead to an increase of the computational effort a lot and a decrease the numerical stability, without

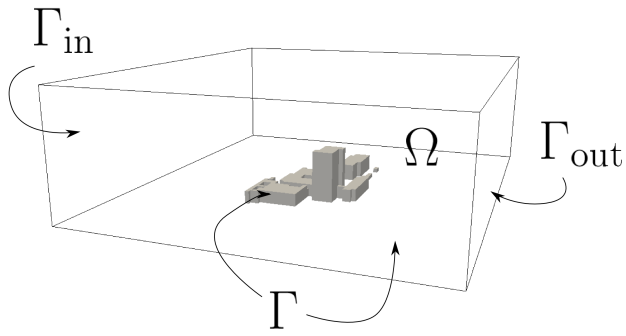


Figure 5. Schematic description of computational domain and boundary conditions for wind flow model.

necessarily yielding large gains in accuracy. On the other hand, the actual discrepancy between a given model and its approximation is easily controllable by the size of voxels, offering the choice between accuracy and computing time in advance.

Another challenge for enabling widespread use of numerical simulations in urban environments is the scarcity of highly accurate city models. This condition can be weakened to the availability of high resolution models in the main areas of interest, since widely available low accuracy models are sufficient for the necessary peripheral simulation in the surrounding area. In spite of the varying detail of the models, the very accurate geographic alignment offers the opportunity for an automated data source selection and pre-processing workflow.

D. Wind Flow Simulation

In both scenarios, we want to compute the flow of the wind around isolated buildings in the city. For this, we employ a simulation that solves a standard model based on the instationary version of the incompressible Navier-Stokes equations (see, e.g., [27]) in a sufficiently large computational domain Ω surrounding the area of interest. We apply suitable artificial boundary conditions for the assumed wind flow conditions, thereby neglecting the impact

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

of surrounding buildings outside the domain. Since air can be considered incompressible for speeds much lower than the speed of sound, these equations provide an accurate description of the behavior of the air flow.

The model is formulated as an initial boundary value problem for a set of partial differential equations, which describe the time evolution of the velocity $\vec{u}(\vec{x},t)$ and the pressure $p(\vec{x},t)$, both of which are functions of position $\vec{x} \in \Omega$ and time t in an interval $[0,T)$. The problem is stated in (1), where the first equation is derived from the principle of conservation of momentum, and the second from that of conservation of mass. The derivation of these equations make use of the fact that air can be considered to be a Newtonian fluid.

$$\begin{aligned}
\partial_t \vec{u} + (\vec{u}\cdot\nabla)\,\vec{u} &= -\frac{1}{\rho_F}\nabla p + \nu\Delta\vec{u}, && \text{in } \Omega\times(0,T),\\
\nabla\cdot\vec{u} &= 0, && \text{in } \Omega\times(0,T),\\
\vec{u} &= \vec{u}^{in}, && \text{in } \Gamma_{\text{in}}\times(0,T),\\
(-\mathcal{I}p + \nu\nabla\vec{u})\cdot\vec{n} &= 0, && \text{in } \Gamma_{\text{out}}\times(0,T),\\
\vec{u} &= 0, && \text{in } \Gamma\times(0,T),\\
\vec{u}(\vec{x},0) &= \vec{u}_0(\vec{x}), && \text{in } \Omega\ .
\end{aligned} \tag{1}$$

Here, the parameters ρ_F and ν correspond to the density and kinematic viscosity of air, which are both assumed to be constant. Since we solve the equations on a truncated domain, the solution has to be prescribed on the boundary. Figure 5 shows a schematic overview of the boundary conditions. At the walls of buildings as well as on the ground, the velocity is set to zero, which corresponds to so-called *no-slip* boundary conditions. This part of the boundary is denoted Γ in (1). On one side of the domain, Γ_{in}, we prescribe a fixed velocity $\vec{u}^{in}$. Since this velocity is not known exactly for a given situation, we need to make an assumption about it. A common model for the general behavior of the lowest layer of the atmosphere (also called the *Prandtl layer*) is to assume that the speed grows logarithmically with the height z above ground [28], [29]. This corresponds to the following expression:

$$\vec{u}^{in}(z) = -\frac{U}{\kappa}\left(\ln\left(\frac{z}{z_0}\right)\right)\vec{n}_{\text{in}}, \tag{2}$$

where U is an estimated average wind speed, $\kappa \approx 0.4$ is the von-Kármán constant, and z_0 is a measure of the roughness, and corresponds to the height above the ground where the velocity becomes zero. The vector $\vec{n}_{\text{in}}$ is the outward unit normal on Γ_{in}. In the lack of wind profile measurements, also a simplified model with a linear profile can be considered:

$$\vec{u}^{in}(z) = -U\left(\frac{z}{z_1}\right)\vec{n}_{\text{in}}, \tag{3}$$

where U in an estimated average wind speed at height z_1. In the simulations, the second approach was adopted, and the parameters were chosen to be arbitrary, but reasonable values, which are shown in Table I. In future work, one could imagine to base the boundary values on current solutions of the lowest layers in weather forecasting models, such as the global model GME [30] or the regional model COSMO [31].

Table I
VALUES OF THE PARAMETERS USED IN THE WIND FLOW SIMULATIONS.

Parameter	Assumed value
Kinematic viscosity ν	0.001 m^2/s
Density ρ_F	1.2041 kg/m^3
Max. inflow speed U	10 m/s
Height z_1	150 m

Here, we have chosen the approach of using fixed values of the velocity on the sides (Dirichlet boundary conditions), for example to set the known wind profile [32]. This offers the chance of using an exterior flow condition on the top plane [33], which can be used to significantly reduce the required size of the computational domain. Another approach for choosing suitable conditions would be to consider a city with regularly aligned blocks and using a lid driven simulation with cyclic boundary conditions on the sides with sufficient height, as in [34].

On the remaining part of the boundary, denoted by Γ_{out}, a relation between pressure and velocity is imposed, which corresponds to an outflow. This so-called *do-nothing* condition appears naturally in the weak formulation that is used for the finite element discretization, and is easy to work with since it does not require any special treatment in the discretization.

The kinematic viscosity ν in (1) describes roughly the thickness of the fluid. It plays an important role via the *Reynolds number*, a dimensionless quantity that characterizes the behavior of the flow with respect to turbulence. It is defined as $\text{Re} = \nu^{-1}|\vec{u}|L$, where L is the characteristic length scale of the problem. When Re is large, the flow has a turbulent character, which requires highly sophisticated methods for its solution. With realistic values of $\nu \approx 10^{-5}$ m^2/s for the type of geometries and flow speeds that we are considering, Re would certainly lie in this regime. Investigations such as those described in [20] and [21] show that this type of turbulence computation is within the possibility of present simulation technology. However, to avoid the additional expense of performing such computations for this scenario, we have chosen to use a larger value of ν. The value for this and the other parameters that were used in the simulations are listed in Table I.

We discretize this mathematical model using a finite element method based on a standard weak formulation of (1). We follow the discretization approach used in [35], with Q_2/Q_1 finite elements, which yields second order accuracy for the velocity field, and first order for the pressure. The solution of the nonlinear system of equations uses the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Newton method with a GMRES linear solver to compute the corrections. The GMRES method uses preconditioning by multilevel incomplete LU factorization through the ILU++ software package described in [36]. The implementation of the simulation is based on the finite element library HiFlow^3 [37].

E. Fine Dust Simulation

For the second scenario, we simulate the spread of fine dust particles in the air. This type of computation has several important applications, which include predicting the effect of pollution (heavy metals, smog, smoke), as well as estimating the transport of naturally occurring dust and pollen, both of which can be useful for instance in city planning. At low altitudes in urban areas, the occurrence of buildings strongly limits the transport of particles, and the question of deposition of particles becomes important. In the following, we describe a mathematical model for particle transport, which is derived from the work presented in [38].

We assume a set of non-interacting, spherical particles P_i, $i = 1, ..., N$, with radii r^i and masses m^i. In the following, a superscript i denotes that a quantity is related to particle P_i. Its position $\vec{x}^i(t)$ will evolve according to its velocity $\vec{u}^i(t)$ via the ordinary differential equation (ODE):

$$\frac{\mathrm{d}\vec{x}^i}{\mathrm{d}t}(t) = \vec{u}^i(t). \tag{4}$$

The velocity of a particle P_i is the sum of the velocity $\vec{u}_F$ of the air at x^i, and a velocity $\vec{u}^i_P$ that arises due to the total external force $\vec{F}^i(t)$ acting on the particle:

$$\vec{u}^i(t) = \vec{u}_F(\vec{x}^i(t)) + \vec{u}^i_P(t). \tag{5}$$

Figure 6 shows the two contributions to the particle velocity together with the forces that are accounted for in the model. The air velocity field $\vec{u}_F$ is obtained from a computation of the wind flow, as described in III-D. In general, this wind field varies in time, but for simplification, we have assumed that it is stationary in our model. One can think of this as an average over time of the possible wind fields; although in the computations, we have simply used the instantaneous solution at an arbitrary point in time.

The second part of the velocity $\vec{u}^i_P(t)$ is computed according to Newton's second law, which can be expressed as follows:

$$m^i \frac{\mathrm{d}\vec{u}^i_P}{\mathrm{d}t}(t) = \vec{F}^i(t). \tag{6}$$

The force acting on a particle is assumed to consist of three effects:

$$\vec{F}^i(t) = \vec{F}^i_{\text{grav}} + \vec{F}^i_{\text{pres}} + \vec{F}^i_{\text{drag}}. \tag{7}$$

Here, $\vec{F}^i_{\text{grav}} = -m^i g \vec{e}_z$ is the gravitational force, with $g \approx 9.81\,\text{m}\cdot\text{s}^{-2}$ the gravity of earth, and $\vec{e}_z$ the upward vertical direction vector. $\vec{F}^i_{\text{pres}} = -\frac{4\pi}{3}(r^i)^3 \nabla p_F$ is the force that the air pressure p_F exerts on the spherical particle. Finally, $\vec{F}^i_{\text{drag}}$ corresponds to the friction force, which acts on the particle as it moves in the fluid. It is given by

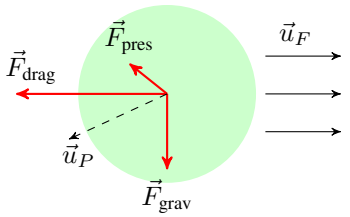


Figure 6. Schematic image of forces acting on a particle in the fine dust model.

$$\vec{F}^i_{\text{drag}} = -0.5 c^i(\text{Re}^i)\rho_F A^i |\vec{u}^i_P| \vec{u}^i_P, \tag{8}$$

where c^i is the drag coefficient associated with, ρ_F the density of the fluid, and $A^i = \pi(r^i)^2$ the cross-sectional area of the particle perpendicular to the direction of motion.

The drag coefficient is determined in terms of the particle Reynolds number, which is defined as $\text{Re}^i = \frac{|\vec{u}^i_P| r^i}{\nu_F}$, where nu_F is the kinematic viscosity of the fluid. An empiric law for the drag coefficient, which is known [39] to be valid for low values of Re^i is

$$c^i = \begin{cases} \frac{24}{\text{Re}^i}, & \text{if } 0.0 < \text{Re}_P \leq 1.0, \\ \frac{24}{(\text{Re}^i)^{0.646}}, & \text{if } 1.0 < \text{Re}_P \leq 400\,. \end{cases} \tag{9}$$

For the computation of Re^i, the kinematic viscosity and density of the fluids were chosen as $\nu_F = 1.71 \cdot 10^{-5}\text{m}^2/\text{s}$ and $\rho_F = 1.20\,\text{kg}/\text{m}^3$, which corresponds to air at standard outside temperatures. We have further assumed for simplicity that all the particles have the same radius $r^i = 1.9 \cdot 10^{-5}\,\text{m}$ and mass $m^i = 1.15 \cdot 10^{-10}\,\text{kg}$: a more sophisticated method would be to assign this at random from a given distribution.

Altogether, the evolution of the particles is described by the $2N$ ODE (4) and (6), supplemented by initial conditions for $\vec{x}^i$ and $\vec{u}^i$ at $t = 0$. These conditions are typically chosen at random based on a distribution that corresponds to the specific situation at hand. For the velocity, another possible choice is to first determine the initial positions, and then to start the particles with the same velocity as the underlying fluid: $\vec{u}^i(0) = \vec{u}^i_F(x^i(0))$.

Many different methods exist for solving systems of ODE. In accordance with the wish to keep our procedure as simple as possible, we have chosen to use quite basic methods. The total time-interval $[0, T)$ is split into time steps of size Δt, and the solution is computed at the discrete times $t_n = n\Delta t$. For solving (4), the implicit Euler method is applied, which yields the following iterative method, for $n = 0, 1, \ldots, T/\Delta t$.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$$\vec{x}^i(t_{n+1}) = \vec{x}^i(t_n) + \Delta t \left(\vec{u}^i_P(t_{n+1}) + \vec{u}^i_F\left(\vec{x}^i(t_{n+1})\right)\right). \quad (10)$$

To avoid having to solve a nonlinear problem in this case, we assume that the fluid velocity varies slowly in space, and make the approximation $\vec{u}^i_F\left(\vec{x}^i(t_{n+1})\right) \approx \vec{u}^i_F\left(\vec{x}^i(t_n)\right)$, which yields the modified iteration step:

$$\vec{x}^i(t_{n+1}) = \vec{x}^i(t_n) + \Delta t \left(\vec{u}^i_P(t_{n+1}) + \vec{u}^i_F\left(\vec{x}^i(t_n)\right)\right). \quad (11)$$

This can be computed explicitly, once $u^i_P(t_{n+1})$ has been determined from the discretization of (6). Again, the implicit Euler method is used, giving the basic iteration:

$$\vec{u}^i_P(t_{n+1}) = \vec{u}^i_P(t_n) + \frac{\Delta t}{m^i}\Big(\vec{F}^i_{\text{drag}}(\vec{u}^i_P(t_{n+1})) + \vec{F}^i_{\text{pres}}(\vec{x}^i(t_{n+1})) + \vec{F}^i_{\text{grav}}\Big). \quad (12)$$

Similarly to above, it is assumed that the gradient of the fluid pressure varies slowly in space, so that the approximation $\vec{F}_{\text{pres}}(\vec{x}^i(t_{n+1})) \approx \vec{F}_{\text{pres}}(\vec{x}^i(t_n))$ can be made. The gravitational force is constant in both time and space. To treat the drag force in an accurate way, we keep the form as it is, and use a fixed point iteration to solve the resulting non-linear equation.

As was the case for the wind flow simulation, the model that we have used here has been simplified to make the computations easier, and to be able to arrive at a result with a limited effort. In particular, a more complete model would also take into account the effects of turbulence, and the resulting random variations in the particle force.

F. Augmented Reality Visualization

The problem of combining virtual objects with an image of reality is discussed in two steps. First of all, the general composition of virtual data with a photographic image is introduced, followed by approaches for the actual alignment of the virtual world with reality in the next subsection.

The visualization method is based on the accurate alignment of the viewer's position and the orientation of his camera view with the three-dimensional city model and the numerical simulation. In the setup considered here, only the graphics representing the flow field are to be embedded in the real-life image as seen in Figure 1, and therefore, the virtual city model and the computational mesh should not be visible. However, the simulation results that are covered by buildings in the city model must also be removed from the image. The approach we followed is to paint the background and city models completely in black. Therefore, the occluded simulation results are masked by the city model, which itself remains invisible, leading to a masked visualization as displayed in Figure 7. All black areas will then be treated as being transparent. Such a color-key method can be improved by rendering using an alpha-channel, but this generally requires more adaptions in the visualization software, and was not deemed necessary for these examples.

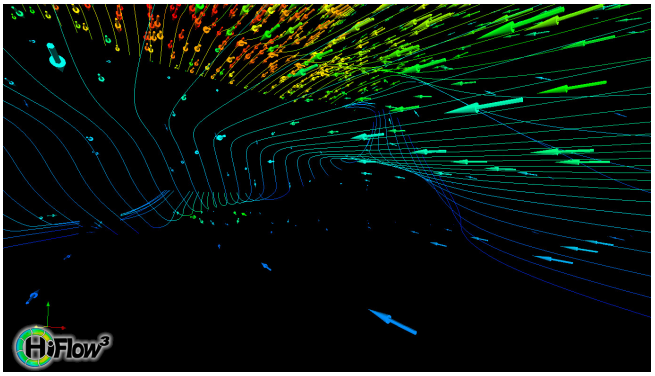


Figure 7. Masked numerical simulation visualization.

The masked visualization can then be composed onto the camera view leading to the augmented numerical simulation visualization in Figure 16, which was extended with the computational domain for illustration. The resulting image is very informative and gives insight into the simulation results. Since the displayed part of the simulation coincides with the viewer's position, the data selection is most intuitive and the full simulation can be explored by simply wandering around in the computational domain. A corresponding AR visualization for an isolated multi-component building in the Physics scenario is shown in Figure 17.

G. Interaction and User Interface

The AR visualization needs accurate positioning and orientation information. This is strongly linked to the user interface, in which the position in space and the view orientation defines the information the viewer wants to analyze. We will discuss the use of sensors of hand-held devices as a man-machine interface, its use for AR positioning and orientation, and an extension for accurate positioning and orientation using active markers.

The interaction and the user interface is crucial for usability and comprehension. The proposed model is to present the mobile device as a window to the AR and the results of the numerical simulation. This leads to challenges as outlined in [13] that can be addressed using sophisticated mathematical methods such as filtering, simulation, and parameter identification. Only the increasing computing power available in modern mobile devices such as smartphones and tablets enable the use of such costly algorithms in real-time the are necessary for responsive haptic user interfaces.

The camera view in space is defined by six parameters, the three-dimensional position and the three viewing angles. Therefore, at least six dimensions of sensor data are needed to control the user interface. Besides GPS, mobile devices of the latest generation contain spatial accelerometers as well as

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

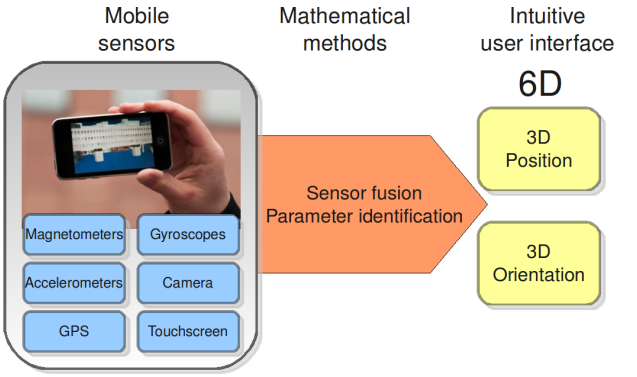


Figure 8. Mathematical methods enable intuitive user interfaces.

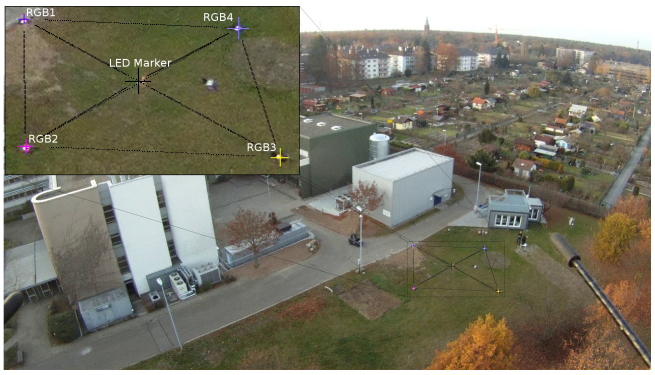


Figure 9. Marker detection from UAV camera view.

spatial magnetometers as a minimum. Taken together, they provide the necessary six degrees of freedom in the sensor data, enabling a new approach to an intuitive interface, which can be improved by any other additional sensors such as gyroscopes or camera based marker detection. Figure 8 illustrates that this step covers the real-time fusion of various sensor readings to gain the position and orientation information that is the basis for the AR visualization.

As evaluated in [40], the effective orientational accuracy of current mobile devices is about two to three degrees in heading, pitch and roll, and an absolute GPS position is accurate to at most 10 m. A typical horizontal field of view of a smartphone camera is 55 degrees, which means that the orientational error results in about 5 % on-screen distance error. The visual error induced by the positioning error depends on the viewing distance to the building. For 50 m distance, the angular error can add up to 16 degrees, for 100 m distance up to 8 degrees, yielding $15 - 30$ % on-screen distance errors. Therefore, user interaction is necessary to align the AR visualization with reality. Although the positioning errors seem to be dominant, they are less problematic once an alignment was successful, as relative GPS measurements are far more accurate.

An alternative approach is to take advantage of markers for augmented reality such as introduced by [41]. While this approach is well suited for small objects, it does not scale up to buildings. Therefore, it was proposed in [42] to introduce active markers for AR visualizations of buildings and simulations. Such markers are not only suited for ground-based AR visualizations, but also for visualizations from radio controlled multicopter aircrafts.

In Figure 9, we show the test setup from an unmanned areal vehicle (UAV) and the accurate detection of the active markers from the movie stream. This resulted in the AR visualization of a building model in Figure 10.

Interaction with a numerical simulation consists not only of moving around and changing the view; it is highly desirable to also offer access to visualization parameters, such as what quantities are displayed, the method used, and

Figure 10. Augmented reality building visualization.

potentially to enable changing some simulation parameters. From the view of the user interface, the touchscreen interfaces of modern mobile devices offer endless possibilities for manipulation of visualization and simulation parameters. Another crucial issue is the interactivity that is offered to the user: the presented visualization needs to be updated frequently, but is limited by the available network bandwidth.

H. Client-Server Framework

In general, large scale numerical simulations and scientific visualization are resource-intensive, and require dedicated high-performance hardware. Although mobile devices are becoming increasingly powerful, there is still a large gap in performance between these devices and the clusters of thousands of servers that are typically used in scientific computing.

In order to enable interactive AR visualizations on mobile devices, we propose a client-server approach where the display and data selection is performed with a user interface on a mobile device, but the actual simulation results and visualization remains on a high-performance server infrastructure. As illustrated in Figure 11, the clients are connected to the visualization service on the servers by wireless or cellular networks, which are limited by the available bandwidth. In a direct image transport, a refresh

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

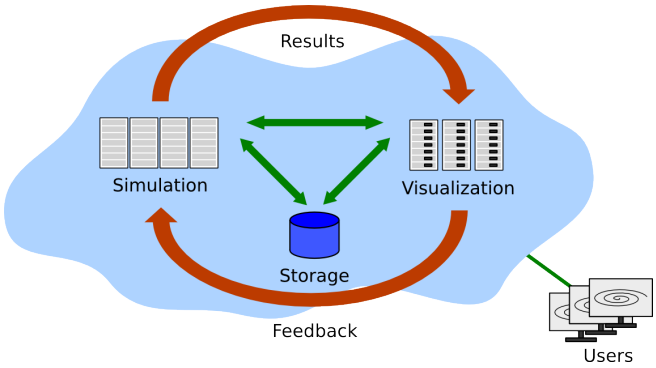


Figure 11. Interaction model.

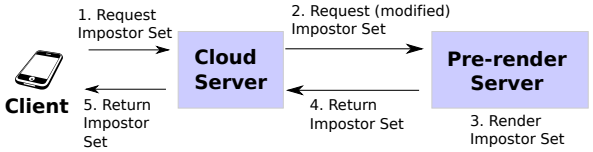


Figure 12. Schematic overview of the client-server framework.

rate of several frames per second is feasible on UMTS networks. But the interactivity is bound to latencies ranging from 100 ms to several seconds.

In computer gaming, there are similar requirements for interactivity as in scientific visualization. In [43], a platform is introduced which aims at providing 3D games even on handheld devices. They either transmit the OpenGL or DirectX commands directly to the client, or use a low-latency version of the H.264 encoder to transmit the visual information to the client. While the system aims at WLAN networks, the concept seems applicable to Long Term Evolution (LTE) mobile networks, that can provide peak bandwidths exceeding 100 Mbps in the downlink direction [44].

In AR applications, orientation and position changes are most common, and theoretically, the optimal approach would in this case be to transmit the full 3D model to the mobile client, to enable realtime interaction. But for large datasets, the mobile devices generally cannot meet the memory demands and GPU performance needed.

Therefore, the approach that we have adopted in the European Project *MobileViz* is to compute visually indistinguishable but reduced 3D models, which enable high refresh rates and low interaction latencies even when they are rendered on a mobile device. The reduced models consist of a set of impostors in the form of simple images, which are generated on the server for the current viewpoint, and then transmitted to the client, where they can be rendered at low cost. The details of this method are described in [45] and [46].

Figure 12 shows schematically how this type of rendering is embedded in our client-server framework. The server component of this framework is split into two parts. The *pre-rendering* service accepts incoming requests for visualizations of particular datasets, and generates the corresponding impostor images, possibly by using hardware dedicated to scientific visualization. The *cloud server* provides a web service, which accepts multiple concurrent incoming requests, and determines which impostors should be generated to fulfill these requests. The requests are then forwarded to the pre-rendering service. In order to keep the load on the pre-rendering server small, the cloud service caches already computed results, and determines the optimal parameters for the impostor rendering. To reduce the amount of computation, it can choose to return a slightly different view than what was requested, in order to make use of already existing data.

In order to give the user of the mobile device the possibility to interact with the visualization, and by extension also the numerical simulation, the *cloud server* will also interact with those components, to forward user requests to them via a specialized interface. Whereas a prototype implementation of the impostor-based rendering is already in place, the development of the aspects dealing with the interactivity is still on-going.

The architecture presented here can be understood in the context of *Mobile Cloud Computing*, where part of an application running on a mobile device is offloaded to a server infrastructure. This model of computing is undergoing rapid growth and offers several advantages, as described in for instance [47] and [48]. In the current work, we have partitioned the application statically between the mobile client and the cloud server. The interaction with the reduced visualization in the form of the impostors takes place on the client, and the actual compute-intensive rendering, on the server.

An alternative approach would be to employ a dynamic partitioning of the execution between server and client as suggested in [47], [48]. The decision of what part is executed where would then be determined by the capacity of the device and the quality of the network connection. A limitation to this approach is that the amount of data being visualized is often very large, and might therefore have to be kept on the cloud server.

IV. Results

In this section, we discuss the results of our tests with the presented methods.

A. Virtual Geometry

Based on 3D city models, our voxelization method is able to derive a computational domains for simulation in a robust way. We joined several data sets of various levels of detail to achieve the most accurate data basis, which was then mapped into voxels of given size. By this, the method can adapt the resulting model to the demanded accuracy, and at

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

the same time filters out small artefacts or errors that would otherwise influence the simulation. The results are presented for two building complexes in Figures 4 and 5.

If a 3D city model is available, the presented method is automated, and delivers computational domains in a robust way. This could be improved by using a more general representation model than a voxel-based approach, but the aspects of robustness, resulting level of detail, and additional computational costs would need to be weighed against the potential benefits. A straight-forward compromise with small additional computational costs in this step, could be to use a hierarchy of voxels, such that the level of detail remains fixed, but larger areas can be covered by larger voxels, as long the numerical method and simulation software permits such selectively coarsened representations of the computational domain. This type of approach could significantly speed up the simulation.

Naturally, the method based on data from a GIS urban model will require additional consideration, as important aspects for simulation were not taken into account in the generation of the models. An example is given by thin glass panes, where both sides face the outside. This needed special treatment to prevent an air passage through this flat object where in reality, the air is blocked. Also additional information about the surface materials should be extracted from databases, to provide hints for which mathematical model should be employed on very smooth surfaces compared to rough planes.

Already in its simple form, however, our method was capable of providing usable computing domains for simulation, while leaving the world coordinate reference system intact, for later virtual reality visualization.

B. Wind Flow Simulation

We used our implementation of the simplified wind flow model described in Section III-D to generate data for the Kronenplatz and Physics scenarios. The same setup was used to treat both the case of the single isolated building in the former scenario, and the group of buildings in the latter. Visualizations with streamlines created using Paraview [49] are shown for the two scenarios in Figures 13 and 14.

For both scenarios, the results obtained are plausible, given the simplifying assumptions made for the model. The way the velocity fields are affected by the presence of buildings is qualitatively correct, which is sufficient to illustrate the functioning and utility of the AR visualization method.

In order to be appropriate for a real use case, the simulation would of course have to deliver data that reflects reality in a more accurate way. The corresponding model would have to use values of the material parameters deduced from measurements, and be modified to deal with the turbulence effects that would arise. Furthermore, the data for the boundary conditions would have to be chosen in a meaningful way. This could be done in several ways: through user input, local measurements, or, as mentioned in Section III-D, interpolation of meteorological data that is available at larger scales.

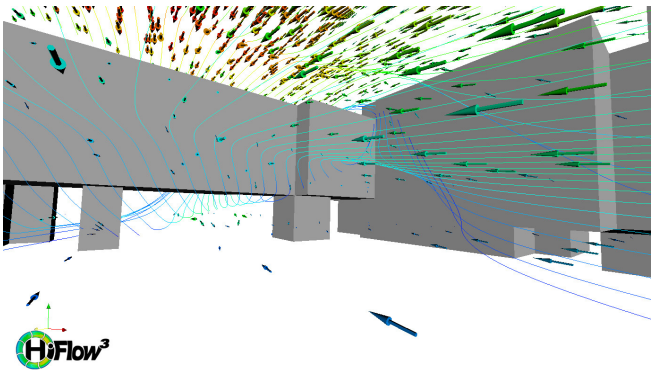


Figure 13. Visualization of computed wind flow field for the Kronenplatz scenario.

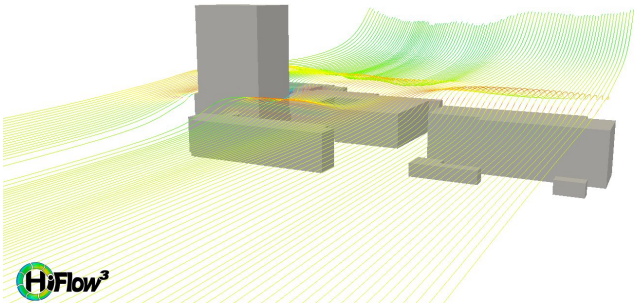


Figure 14. Visualization of computed wind flow field for the Physics buildings.

Naturally, so long as one can only obtain sparse and imprecise information about the current state of the wind, the accuracy of the simulation results will be limited. Therefore, it is important to be clear about the suitability of the simulation results in the context of specific use cases. We would expect that this type of simulation, together with the AR visualization method that we propose, find use for instance when assessing decisions in urban planning, or when evaluating risks associated with airborne pollution. In these cases, one can base the computations on sets of measurements taken over a long time, or averages thereof. Of course, the simulation cannot be expected to exceed the accuracy of the data describing the meteorological situation and the computational domain. Communicating the restrictions in accuracy to the user of the visualization remains an important open problem.

C. Particle Simulation

For the second scenario with the Physics building, we also implemented a numerical simulation for the spread of

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

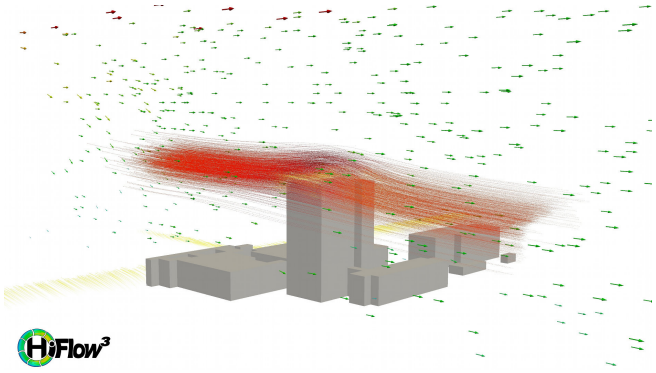


Figure 15. Visualization of fine dust particles distributed around the Physics buildings.

Figure 16. Enhanced Augmented Reality visualization of air flow.

Figure 17. Augmented Reality visualization of air flow around isolated building from the Physics scenario.

fine dust based on the mathematical model described in Section III-E. This simulation used the computed velocity field of the wind described in the previous section. We considered a setup with particles originating from a hypothetical chimney high up in the air, as well as along a hypothetical street passing by parallel to the buildings. The result of the simulation is shown in Figure 15, which displays the positions of the particles throughout the simulated time interval, in order to capture the entire simulation in one picture.

As was the case for the wind simulation, the results are of sufficient quality to illustrate the potential of particle simulations in conjunction with AR visualization, but cannot be considered an accurate representation of how particles would really behave in the atmosphere. The errors in the simulation are due both to the inaccuracies in the model for the wind flow and the simplifications that were made in the particle model. Additionally, the initial particle distribution is synthetically generated in this case, whereas a realistically relevant simulation would require measurements of this data.

We consider this type of simulation coupled with AR visualization to be applicable to for instance urban planning, evaluations the impact of pollution on the environment, and disaster planning.

D. AR Visualization of Wind Field and Particles

We combined the simulation data from the wind flow and particle computations for the Physics scenario into one image using the masking technique described in Section III-F. Figure 18 shows one such image, where the underlying photo was taken using a standard camera. This example illustrates how numerical results from several computations can be combined into one image, providing several pieces of information at once. The viewer gets an idea both about how the wind flows around the building, and how small particles might behave in this flow.

This image was created manually by aligning the computational geometry with the corresponding objects in the photo. The alignment is critical for the visualization, and not an easy task due to potential inaccuracies of the position and orientation information in the computational geometry, as well as additional camera parameters, such as field of view or distortions.

On a mobile device on-site, this information is available, at least approximately, and one can hope to obtain a rea-

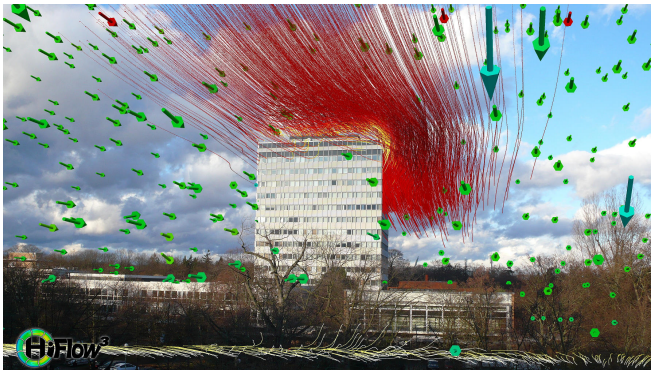


Figure 18. Augmented Reality visualization of velocity field and distributed fine dust particles around the Physics buildings, created using the masking technique.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

sonably good fit between the simulated data and the camera image. Where high accuracy is required, one can compensate for errors in the position and orientation using a marker-based approach, as it was presented before.

The feasibility of such a solution was presented in Figures 9 and 10 for use with UAVs, which hints to a very promising application field of the presented visualization method in combination with flying cameras. This way, the simulation can be analyzed using the AR visualization also from above.

V. Conclusion

In this paper, we have presented a novel visualization method for large-scale scientific computing, illustrated by the examples of simulating urban air flow and fine dust distribution. The use of mobile devices opens the path to intuitive access to, and interaction with, numerical simulations that are highly comprehensible due the embedding in to the real-life camera view as AR visualizations. This is an answer to how sophisticated simulations can be made usable for non-scientists, as it is replacing the artificial and complex virtual representation of reality with a direct view of reality itself. However, this is not a complete solution, since the actual representation of simulation results needs to be understood correctly. This AR presentation aids greatly through the direct correspondence with reality, but there are other areas that require further investigation to find suitable imaging methods. For instance all numerical simulations are approximations with associated errors, both introduced by the computation itself, and by the limited accuracy of the measurements. Such uncertainties should be made obvious also to an uninformed viewer. Suitable visualization concepts for this is an open area of research.

An advantage of the method is the simplicity of selecting the data of interest and view orientation by just walking through the immersive simulation in reality and pointing the mobile device. Of course, this is limiting us to views from places, that the viewer can walk to. The general availability of UAV, combined with their ease of operation, is overcoming this issue to some extent.

We depend on the availability of a 3D city model of sufficient accuracy, in order to derive a computational domain in a robust way. All additional information that is included in the model, such as surface properties, can aid to improve the simulation quality. The introduction and adoption of a general standard such as the CityGML standard is of great help, but also offers the chance to integrate simulations into GIS databases. The work presented here, could improve the way in which such information is evaluated through AR visualizations.

The technical problem of exact alignment of real-world images with the virtual objects cannot yet be solved solely based on sensor measurements of mobile devices, but active markers can help solving this issue. This is a topic of ongoing research and development.

Another technical problem is to derive accurate information on the current conditions around the computing domain, such as the current weather conditions. Such information is available in databases from weather forecast agencies, but the resolution provided is on the order of kilometers, compared to the level of detail suitable for this visualization method that can go down the order of meters. We can expect the availability of higher resolution weather models in future, but suitable mathematical modeling for interpolation of surrounding weather conditions is a topic current research.

The proposed remote visualization method detailed in [45] and [46] is perfectly suited for displaying large stationary numerical simulations on mobile devices using the presented AR visualization method, due to its support for AR applications and its economic resource usage. By exploiting the increasing graphical performance of mobile devices, the scarce network bandwidth is utilized very efficiently. It is desirable to extend this method to instationary simulations as well, but the increased amount of data to be transmitted is limited by the traditionally small bandwidth available to mobile devices. There are promising approaches for periodic cases, but until there are new concepts for remote visualizations, increased bandwidth through new transmission standards look promising to solve this issue.

The development of the client-server framework for remote visualization enables the access to, and interaction with, large scientific datasets on mobile devices. Although still not completed, the design and prototype implementation of this framework is an important step towards realizing the goal of providing distributed visualization and simulation services over the Internet.

The presented AR visualization method is very general in its scope in the sense that it is usable for many application areas. It is expected to facilitate the use of numerical simulations by scientists as well as citizens and decision-makers. Furthermore, we are convinced that it can increase the impact and improve the communication of scientific results in interdisciplinary collaborations and to the general public.

VI. Acknowledgments

The *Karlsruhe Geometry* project is a collaboration of the Liegenschaftsamt of the city council of Karlsruhe with the Engineering Mathematics and Computing Lab (EMCL) and was supported by the KIT Competence Area for Information, Communication and Organization. The authors thank the Fraunhofer IOSB Karlsruhe and Building Lifecycle Management (BLM) at the KIT for execution of the UAV flights. The development of intuitive user interfaces for scientific applications on mobile devices was part of the *Science to Go* project, which received funding from the Apple Research & Technology Support (ARTS) programme. The authors

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

appreciate the support of the Federal Ministry of Education and Research and Eurostars within the Project E! 5643 MobileViz. The Eurostars Project is funded by the European Union.

REFERENCES

[1] V. Heuveline, S. Ritterbusch, and S. Ronnas, "Augmented reality for urban simulation visualization," in *Proceedings of The First International Conference on Advanced Communications and Computation INFOCOMP 2011*. Barcelona, Spain: IARIA, 2011, pp. 115–119.

[2] U. Neumann and A. Majoros, "Cognitive, performance, and systems issues for augmented reality applications in manufacturing and maintenance," in *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE 1998*. IEEE, 1998, pp. 4–11.

[3] R. T. Azuma *et al.*, "A survey of augmented reality," *Presence-Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.

[4] K. Gerdes, "A summary of infinite element formulations for exterior helmholtz problems," *Computer methods in applied mechanics and engineering*, vol. 164, no. 1, pp. 95–105, 1998.

[5] J. P. Wolf and C. Song, *Finite-element modelling of unbounded media*. Wiley Chichester, England, 1996.

[6] W. J. Schroeder, F. Bertel, M. Malaterre, D. Thompson, P. P. Pebay, R. O'Bara, and S. Tendulkar, "Methods and framework for visualizing higher-order finite elements," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 446–460, 2006.

[7] J. Kang, "Numerical modelling of the sound fields in urban streets with diffusely reflecting boundaries," *Journal of sound and vibration*, vol. 258, no. 5, pp. 793–813, 2002.

[8] A. J. Arnfield, "Two decades of urban climate research: a review of turbulence, exchanges of energy and water, and the urban heat island," *International Journal of Climatology*, vol. 23, no. 1, pp. 1–26, 2003.

[9] S. R. Musse and D. Thalmann, "Hierarchical model for real time simulation of virtual human crowds," *IEEE Transactions on Visualization and Computer Graphics*, vol. 7, no. 2, pp. 152–164, 2001.

[10] T. Kolbe, G. Gröger, and L. Plümer, "CityGML: Interoperable access to 3d city models," in *Geo-information for Disaster Management*, P. Oosterom, S. Zlatanova, and E. Fendel, Eds. Springer Berlin Heidelberg, 2005, pp. 883–899.

[11] T. H. Kolbe, "Representing and exchanging 3d city models with CityGML," in *Proceedings of the 3rd International Workshop on 3D Geo-Information, Lecture Notes in Geoinformation & Cartography*, J. Lee and S. Zlatanova, Eds. Seoul, Korea: Springer Verlag, 2009, p. 20.

[12] S. Feiner, B. MacIntyre, T. Hollerer, and A. Webster, "A Touring machine: prototyping 3d mobile augmented reality systems for exploring the urban environment," in *Wearable Computers, 1997. Digest of Papers., First International Symposium on*, oct 1997, pp. 74 –81.

[13] J. B. Gotow, K. Zienkiewicz, J. White, and D. C. Schmidt, "Addressing challenges with augmented reality applications on smartphones," in *MOBILWARE*, 2010, pp. 129–143.

[14] D. Schmalstieg, T. Langlotz, and M. Billinghurst, "Augmented reality 2.0," in *Virtual Realities*, G. Brunnett, S. Coquillart, and G. Welch, Eds. Springer Vienna, 2011, pp. 13–37.

[15] D. Wagner, T. Pintaric, F. Ledermann, and D. Schmalstieg, "Towards massively multi-user augmented reality on handheld devices," in *In Third International Conference on Pervasive Computing*, 2005.

[16] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, "Recent advances in augmented reality," *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, 2001.

[17] A. Nurminen, E. Kruijff, and E. E. Veas, "Hydrosys - a mixed reality platform for on-site visualization of environmental data," in *W2GIS*, 2010, pp. 159–175.

[18] H. Graf, P. Santos, and A. Stork, "Augmented reality framework supporting conceptual urban planning and enhancing the awareness for environmental impact," in *Proceedings of the 2010 Spring Simulation Multiconference*. ACM, 2010, pp. 181:1–181:8.

[19] M. Hammoudeh, R. Newman, C. Dennett, and S. Mount, "Interpolation techniques for building a continuous map from discrete wireless sensor network data," *Wireless Communications and Mobile Computing*, 2011. [Online]. Available: http://dx.doi.org/10.1002/wcm.1139

[20] S. R. Hanna, M. J. Brown, F. E. Camelli, S. T. Chan, W. J. Coirier, S. Kim, O. R. Hansen, A. H. Huber, and R. M. Reynolds, "Detailed simulations of atmospheric flow and dispersion in downtown Manhattan: An application of five computational fluid dynamics models," *Bulletin of the American Meteorological Society*, vol. 87, no. 12, pp. 1713–1726, Dec 2006. [Online]. Available: http://dx.doi.org/10.1175/BAMS-87-12-1713

[21] P. Gousseau, B. Blocken, T. Stathopoulos, and G. van Heijst, "CFD simulation of near-field pollutant dispersion on a high-resolution grid: A case study by les and rans for a building group in downtown montreal," *Atmospheric Environment*, vol. 45, no. 2, pp. 428 – 438, 2011.

[22] J. S.-D. Muro, E. J. Macías, J. B. Barrero, and M. P. de la Parte, "Two-dimensional model of wind flow on buildings to optimize the implementation of mini wind turbines in urban spaces," in *International Conference on Renewable Energies and Power Quality*, 2010.

[23] F. Balduzzi, A. Bianchini, and L. Ferrari, "Microeolic turbines in the built environment: Influence of the installation site on the potential energy yield," *Renewable Energy*, vol. 45, pp. 163 – 174, 2012.

[24] T. Hauenstein, "Das 3D-Stadtmodell Karlsruhe," in *INTERGEO*, 2009. [Online]. Available: http://www.intergeo.de/archiv/2009/Hauenstein.pdf 29.7.2011

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[25] M. J. Krause, "Fluid flow simulation and optimisation with lattice boltzmann methods on high performance computers: Application to the human respiratory system," Ph.D. dissertation, Karlsruhe Institute of Technology (KIT), 2010.

[26] K. Inthavong, J. Wen, J. Tu, and Z. Tian, "From CT scans to CFD modelling - fluid and heat transfer in a realistic human nasal cavity," *Engineering Applications of Computational Fluid Mechanics*, vol. 3, no. 3, pp. 321–335, 2009.

[27] J. H. Spurk and N. Aksel, *Fluid Mechanics*, 2nd ed. Springer-Verlag Berlin Heidelberg, 2008.

[28] H. Kraus, *Die Atmosphäre der Erde: Eine Einführung in die Meteorologie*. Springer Berlin Heidelberg, 2004.

[29] D. Etling, *Theoretische Meteorologie: Eine Einführung*. Springer Berlin Heidelberg, 2008.

[30] D. Majewski, D. Liermann, P. Prohl, B. Ritter, M. Buchhold, T. Hanisch, G. Paul, W. Wergen, and J. Baumgardner, "The operational global icosahedral-hexagonal gridpoint model GME: Description and high-resolution tests," *Monthly Weather Review*, vol. 130, no. 2, pp. 319–338, 2002.

[31] "Core documentation of the COSMO-model," http://www.cosmo-model.org/content/model/documentation/core/default.htm (Accessed 2013-06-09).

[32] I. Waltschläger, "Randbedingungen zur Windsimulation im Stadtgebiet," Master's thesis, Karlsruhe Institute of Technology (KIT), 2011.

[33] V. Heuveline and P. Wittwer, "Adaptive boundary conditions for exterior stationary flows in three dimensions," *Journal of Mathematical Fluid Mechanics*, vol. 12, no. 4, pp. 554–575, 2009.

[34] P. He, T. Katayama, T. Hayashi, J. Tsutsumi, J. Tanimoto, and I. Hosooka, "Numerical simulation of air flow in an urban area with regularly aligned blocks," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 67-68, pp. 281 – 291, 1997.

[35] V. John, G. Matthies, and J. Rang, "A comparison of time-discretization/linearization approaches for the incompressible Navier-Stokes equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 44/47, pp. 5995 – 6010, 2006.

[36] J. Mayer, "A multilevel Crout ILU preconditioner with pivoting and row permutation," *Numerical Linear Algebra with Applications*, vol. 14, no. 10, pp. 771–789, 2007. [Online]. Available: http://dx.doi.org/10.1002/nla.554

[37] H. Anzt, W. Augustin, M. Baumann, T. Gengenbach, T. Hahn, A. Helfrich-Schkarbanenko, V. Heuveline, E. Ketelaer, D. Lukarski, A. Nestler, S. Ritterbusch, S. Ronnas, M. Schick, M. Schmidtobreick, C. Subramanian, J.-P. Weiss, F. Wilhelm, and M. Wlotzka, "HiFlow3: A hardware-aware parallel finite element package," in *Tools for High Performance Computing 2011*, H. Brunst, M. S. Müller, W. E. Nagel, and M. M. Resch, Eds. Springer Berlin Heidelberg, 2012, pp. 139–151.

[38] T. Gengenbach, "Numerical simulation of particle deposition in the human lung," Ph.D. dissertation, Karlsruhe Institute of Technology, 2012.

[39] J. K. Comer, C. Kleinstreuer, and C. S. Kim, "Flow structures and particle deposition patterns in double-bifurcation airway models. Part 2. Aerosol transport and deposition," *Journal of Fluid Mechanics*, vol. 435, pp. 55–80, 4 2001.

[40] M. K. Kirchhoefer, J. H. Chandler, and R. Wackrow, "Cultural heritage recording utilising low-cost close-range photogrammetry," in *Proceedings of CIPA 23rd International Symposium*, 2011.

[41] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," *2nd IEEE and ACM International Workshop on Augmented Reality*, pp. 85–94, 1999.

[42] V. Koch, S. Ritterbusch, A. Kopmann, M. Mueller, T. Habel, and P. von Both, "Flying augmented reality," in *Proceedings of the 29th eCAADe conference*, Ljubljana, Slovenia, 2011.

[43] A. Jurgelionis, P. Fechteler, P. Eisert, F. Bellotti, H. David, J. P. Laulajainen, R. Carmichael, V. Poulopoulos, A. Laikari, P. Peraelae, A. D. Gloria, and C. Bouras, "Platform for distributed 3d gaming," *International Journal of Computer Games Technology*, vol. 2009, p. 15, 2009.

[44] E. Dahlman, H. Ekström, A. Furuskar, Y. Jading, J. B. Karlsson, M. Lundevall, and S. Parkvall, "The 3G long-term evolution - radio interface concepts and performance evaluation," in *IEEE 63rd Vehicular Technology Conference*, vol. 1, 2006, pp. 137–141.

[45] A. Helfrich-Schkarbanenko, V. Heuveline, R. Reiner, and S. Ritterbusch, "Bandwidth-efficient parallel visualization for mobile devices," in *The Second International Conference on Advanced Communications and Computation*. IARIA, 2012, pp. 106–112.

[46] V. Heuveline, M. Baumann, S. Ritterbusch, and R. Reiner, "Method and system for scene visualization," Feb. 27 2013, WO Patent 2,013,026,719.

[47] D. Kovachev and R. Klamma, "Beyond the client-server architectures: A survey of mobile cloud techniques," in *1st IEEE International Conference on Communications in China Workshops (ICCC)*, 2012, pp. 20–25.

[48] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Networks and Applications*, vol. 18, no. 1, pp. 129–140, 2013.

[49] "ParaView - Open Source Scientific Visualization," http://www.paraview.org/ (Accessed 2013-06-09).

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

An Explorative Study of Module Coupling and Hidden Dependencies based on the Normalized Systems Framework

Dirk van der Linden, Peter De Bruyn, Herwig Mannaert, and Jan Verelst
University of Antwerp
Antwerp, Belgium
dirk.vanderlinden, peter.debruyn, herwig.mannaert, jan.verelst@uantwerpen.be

***Abstract*—Achieving the property of evolvability is considered a major challenge of the current generation of large, compact, powerful, and complex systems. An important facilitator to attain evolvability is the concept of modularity: the decomposition of a system into a set of collaborating subsystems. As such, the implementation details of the functionality in a module is hidden, and reduces complexity from the point of view of the user. However, some information should not be hidden if they hinder the (re)use of the module when the environment changes. More concretely, all collaborating modules must be available for each other. The way how a collaborating module is accessible is also called module coupling. In this paper, we examined a list of classifications of types of module couplings. In addition, we made a study on the implications of the used address space for both data and functional constructs, and the implications of how data is passed between modules in a local or remote address space. Several possibilities are evaluated based on the Normalized Systems Theory. Guidelines are derived to improve reusability.**

Keywords-Reusability, Evolvability, Modularity, Coupling, Address space.

I. INTRODUCTION

Modern technologies provide us the capabilities to build large, compact, powerful, and complex systems. Without any doubt, one of the major key points is the concept of modularity. Systems are built as structured aggregations of lower-level subsystems, each of which have precisely defined interfaces and characteristics. In hardware for instance, a USB memory stick can be considered a module. The user of the memory stick only needs to know its interface, not its internal details, in order to connect it to a computer. In software, balancing between the desire for information hiding and the risk of introducing undesired hidden dependencies is often not straightforward. However, these undesired hidden dependencies should be made explicit [1]. Experience contributes in learning how to deal with this issue. In other words, best practices are rather derived from heuristic knowledge than based on a clear, unambiguous theory.

Normalized Systems Theory has recently been proposed [2] to contribute in translating this heuristic knowledge into explicit design rules for modularity. In this paper, we want to evaluate which information hiding is desired and which is not with regard to the theorems of Normalized Systems. The Normalized Systems theorems are fundamental, but it is not always straightforward to check implementations in different application domains against these theorems. This paper aims at deriving more concrete guidelines for software development in a PLC environment on a conceptual level.

Doug McIlroy already called for *families of routines to be constructed on rational principles so that families fit together as building blocks. In short, [the user] should be able safely to regard components as black boxes* [3]. Decades after the publication of this vision, we have black boxes, but it is still difficult to guarantee that users can use them safely. However, we believe that a lot of necessary knowledge to achieve important parts of this goal are available and we should primarily document all the necessary unambiguous rules to make this (partly tacit) knowledge explicit.

In this paper, we examined a list of classifications of types of module couplings, and evaluated in which terms these types are contributing towards potentially compliance with the Normalized Systems theory. These couplings are studied in an abstract environment [1]. Further, we extended this study by placing the constructs in an address space, and evaluated the consequences. This evaluation is based on some case studies in an IEC 61131-3 programming environment by way of small pieces of code [4]. We investigated on how different data constructs relate to a local or a remote memory address space, and which consequences these relations have to functional modules. Next, we placed the focus on the functional constructs and paradigms, which also reside in a local address space and might have a coupling to a remote address space. We investigated the potential to use them complying the Normalized Systems principles. Finally, we present an set of derived, more concrete principles.

The paper is structured as follows. In Section II, the Normalized Systems theory will be discussed. In Section III, we discuss categories of coupling, seen in an abstract way. In Section IV, we give an overview of how data can be passed between functional modules in a local data memory address space, or coupled with constructs in a remote address spaces. In Section V, we focus on constructs for functionality, and how they can be coupled (locally or remotely). A summary of the evaluations and guidelines is given in Section VI. Finally, Section VII concludes the paper.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

II. Normalized Systems

The current generation of systems faces many challenges, but arguable the most important one is evolvability [5]. The evolvability issue of a system is the result of the existence of Lehman's Law of Increasing Complexity which states: "As an evolving program is continually changed, its complexity, reflecting deteriorating structure, increases unless work is done to maintain or reduce it" ([6] p. 1068). Starting from the concept of systems theoretic stability, the Normalized Systems theory is developed to contribute towards building systems, which are immune against Lehman's Law.

A. Stability

The postulate of Normalized Systems states that *a system needs to be stable with respect to a defined set of anticipated changes*. In systems theory, one of the most fundamental properties of a system is its stability: a bounded input function results in bounded output values, even for $T \rightarrow \infty$ (with T representing time).

Consequently, the impact of a change should only depend on the nature of the change itself. Systems, built following this rule can be called stable systems. In the opposite case, changes causing impacts that are dependent on the size of the system, are called *combinatorial effects*. To attain stability, these combinatorial effects should be removed from the system. Systems that exhibit stability are defined as *Normalized Systems*. Stability can be seen as the requirement of a linear relation between the cumulative changes and the growing size of the system over time. Combinatorial effects or instabilities cause this relation to become exponential (Figure 1). The design theorems of Normalized Systems Theory contribute to the long term goal of keeping this relation linear for an unlimited period of time, and an unlimited amount of anticipated changes to the system.

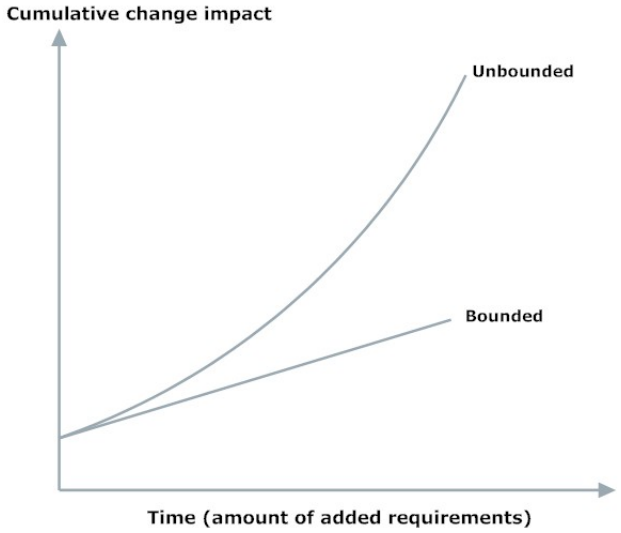


Figure 1. Cumulative impact over time

B. Design Theorems of Normalized Systems

In this section, we give an overview of the design theorems or principles of Normalized Systems theory, i.e., to design systems that are stable with respect to a defined set of anticipated changes:

- A new version of a data entity;
- An additional data entity;
- A new version of an action entity;
- An additional action entity.

Please note that these changes are associated with software primitives in their most elementary form. Hence, real-life changes or changes with regard to 'high-level requirements' should be converted to these elementary anticipated changes [7]. We were able to convert all real-life changes in several case studies to one or more of these abstract anticipated changes [8][9]. However, the systematic transformation of real-life requirements to the elementary anticipated changes is outside the scope of this paper. In order to obtain systems theoretic stability in the design during the implementation of software primitives, Normalized Systems theory prescribes the following four theorems:

1) Separation of concerns:

An action entity can only contain a single task in Normalized Systems.

This theorem focuses on how tasks are structured within processing functions. Each set of functionality, which is expected to evolve or change independently, is defined as a change driver. Change drivers are introducing anticipated changes into the system over time. The identification of a task should be based on these change drivers. A single change driver corresponds to a single *concern* in the application.

2) Data version transparency:

Data entities that are received as input or produced as output by action entities, need to exhibit version transparency in Normalized Systems.

This theorem focuses on how data structures are passed to processing functions. Data structures or *data entities* need to be able to have multiple versions, without affecting the processing functions that use them. In other words, data entities having the property of data version transparency, can evolve without requiring a change of the interface of the action entities, which are consuming or producing them.

3) Action version transparency:

Action entities that are called by other action entities, need to exhibit version transparency in Normalized Systems.

This theorem focuses on how processing functions are called by other processing functions. Action entities need to be able to have multiple versions without affecting any of the other action entities that call them. In other words, action entities having the property of action version transparency, can evolve without requiring a change of one or more action entities, which are connected to them.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

4) Separation of states: The calling of an action entity by another action entity needs to exhibit state keeping in Normalized Systems.

This theorem focuses on how calls between processing functions are handled. Coupling between modules, that is due to errors or exceptions, should be removed from the system to attain stability. This kind of coupling can be removed by exhibiting state keeping. The (error) state should be kept in a separate data entity.

III. Evaluation of types of coupling

Coupling is a measure for the dependencies between modules. Good design is associated with low coupling and high reusability. However, merely lowering the coupling is not sufficient to guarantee reusability. Classifications of types of coupling were proposed in the context of structured design and computer science [10][11]. The key question of this paper is whether a hidden dependency and, therefore, coupling is affecting the reusability of a module? In general, the Normalized Systems theorems identify places in the software architecture where high (technical) coupling is threatening evolvability [12]. More specifically, we will focus in this section on several kinds of coupling and evaluate which of them is lowering or improving reusability. The sequence of the subsections is chosen from the most tight type coupling to the most loose type of coupling.

A. Content coupling

Content coupling occurs when module A refers directly to the content of module B. More specifically, this means that module A changes instructions or data of module B. When module A branches to instructions of module B, this is also considered as content coupling.

It is trivial that direct references between (internal data or program memory of) modules prevent them from being reused separately. In terms of Normalized Systems, content coupling is a violation of the first theorem, separation of concerns. Achieving version transparency is practical not possible. The same can be said about separation of states.

This intent to avoid content coupling is not new, other rules than those of the Normalized Systems already made this clear. For instance, Dijkstra suggested decades ago to abolish the goto statement from all 'higher level' programming languages [13]. The goto statement could indeed be used for making a direct reference to a line of code in another module. Together with restricting access to the memory space of other modules, Dijkstra's suggestion contributed to exile content coupling out of most modern programming languages. Note that in the IEC 61131-3 standard, the Instruction List (IL) language still contains the JMP (jump) instruction. For this and other reasons, IL is considered a low level language, and similar to assembly.

B. Common coupling

Common coupling occurs when modules *communicate using global variables*. A global variable is accessible by all modules in the system, because they have a memory address in the 'global' address space of the system. If a developer wants to reuse a module, analyzing the code of the module to determine which global variables are used is needed. In other words, a white box view is required. Consequently, black box use is not possible. In terms of Normalized Systems, common coupling is a violation of the first theorem, separation of concerns.

We add however, that not the *existence* but the *way of use* of global variables violates the separation of concerns theorem. A global variable is in fact just a variable in the scope of the main program. When these *global variables* are treated like a kind of *local variables in the scope of the main program*, they do not cause combinatorial effects. However, when these variables are passed to the submodular level without using the interface of (sub)modules, which are called by the main program, they can cause combinatorial effects. Since the use of global variables in case of common coupling is not visible through the (sub)module's interface, this way to use these global variables is considered to be a hidden dependency. And since common coupling is a violation of separation of concerns, this is an undesired hidden dependency with respect to the safe use of black boxes.

As a research case, we used global variables in a proof of principle with IEC 61131-3 code, which complies with Normalized Systems [9]. The existence of global variables was needed for other reasons than mutual communication between modules (i.e., connections with process hardware). In this project, the global variables were passed via an interface from one module to the other. In some cases, having a self-explaining interface between collaborating modules is enough to comply with the separation of concerns principle. In other cases, dedicated modules called connection entities are needed to guarantee this separation. In this paper, we investigated in which cases there is a need for a connection entity or not (see following subsections).

C. External coupling

External coupling occurs when two or more modules communicate by using an external (third party?) database, communication protocol, device or hardware interface. The external entity, system or subsystem is accessible by all (internal) modules. Consequently, the support (e.g., fault handling) for the external access has to be included for all modules.

Support for this particular external access is a concern. Every module also includes at least one core functionality, which is also a concern. Having more than one concern in a single module is a violation of the separation of concerns principle. Indeed, when the external entity receives

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

an update, every module, which is calling the external entity, needs an update too. This is an example of a combinatorial effect.

To avoid this kind of combinatorial effect, one should dedicate a special module - a connection entity - to make the link with the external technology. More precisely, one connection entity for every version or alternative external technology. Version tags can be used to select the appropriate connection entity. Each internal module should call the connection entity to map parameters with the external entity.

Such a connection entity is considered to be a supporting task. Separating the core task from supporting task does not have to decrease cohesion. On the contrary they can nicely fit together on the next modular level. In other words, the core task module can be 'hosted' together with one or more supporting task module in a higher-lever module.

D. Control coupling

Control coupling occurs when module A influences the execution of module B by passing data (parameters). Commonly, such parameters are called '*flags*'. Whether a module with such a flag can be used as a black box depends on the fact whether the interface is explaining sufficiently the meaning of this flag for use. If a white box view is necessary to determine how to use the flag, black box use is not possible. The evaluation of control coupling in terms of reusability is twofold. On the one hand, adding a flag can introduce a slightly different functionality and improve the reuse potential. For example, if a control module of a motor is supposed to control pumping until a level switch is reached, a flag can provide the flexibility to use both a positive level switch signal and an inverted one (i.e., positive versus negative logic). On the other hand, extending this approach to highly generic functions, would lead in its ultimate form to a single function *doIt*, that would implement all conceivable functionality, and select the appropriate functionality based on arguments. Obviously, the latter would not hit the spot of reusability.

One of the key questions during the evaluation of control coupling is: how many functionalities should be hosted in one module? In terms of Normalized Systems, the principle 'separation of concerns' should not be violated. The concept of change drivers brings clarity here. A module should contain only one core task, eventually surrounded by supporting tasks. Control coupling can help to realize theorem 2 (data version transparency) and theorem 3 (action version transparency) by way of version selection. The calling action is able to select a version of the called action based on control coupling. We conclude that *control coupling should be used for version selection only*.

Control coupling, as a way of connecting two or more modules, says something about the functional impact of the coupling, not about how the coupling is realized. Consequently, control coupling does not influence the choice whether a connection entity is necessary or not.

E. Data coupling

Data coupling occurs when two modules pass data using simple data types (no data structures), and every parameter is used in both modules.

Realizing theorem 3 (action version transparency) is not straightforward with data coupling, since the introduction of a new parameter affects the interface of the module. This newer version of the interface could not be suitable for previous action versions, and could consequently not be called a version transparent update. Not all programming languages support flexibility in terms of the amount of individual parameters. Changing the datatype, or removing a parameter is even worse.

Note that the disadvantage of data coupling, affecting the module's interface in case of a change, does not apply on reusing modules, which are not evolving. This can be the case when working with system functions, e.g., aggregated in a system function library. However, problems can occur when the library is updated. We will give more details about this issue in the next section.

When working with separated, simple data types as a set of parameters, every change requires a change of the interface of the module. Since we do not consider 'changing the interface' as one of our anticipated changes, this should be avoided. Huang et al. emphasized that it is important to separate the version management of components with their interfaces [14]. As such, the interface can be seen as a concern, and should consequently be separated to comply with the separation of concerns principle.

In other words, in case the development environment does not support a flexible interface for its modules, data coupling can cause combinatorial effects. In case mandatory arguments are removed in a new version, a flexible development cannot guarantee the absence of combinatorial effects.

F. Stamp coupling

Stamp coupling occurs when module A calls module B by passing a data structure as a parameter when module B does not require all the fields in the data structure.

It could be argued that using a data structure limits the reuse to other systems where this data structure exists, whereas only sending the required variables separately (like with data coupling) does not impose this constraint. However, we emphasize that the key point of this paper does not concern *reuse* in general. Rather, it focuses on *safe reuse* specifically. Stamp coupling is an acceptable form of coupling. With regard to the first theorem, separation of concerns, one should keep the parameter set (data entity), the functionality of the module (action entity) and the interface separated. Keeping the interface unaffected, while the data entity and action entity are changing, can be realized with stamp coupling. Note that stamp coupling should be

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

combined with the rule that fields of a data structure can be added, but not modified or deleted. This rule is necessary to enable version transparency.

Note that if the data structure in a stamp coupling scenario increases, it becomes convenient to pass the structure by reference (see Section IV-D). As such, memory use and copying processes can be limited. However, referring to the data structure requires the stamp coupling to be applied between modules which reside in the same address space (see Section V-D).

G. Message coupling

Message coupling occurs when communication between two or more modules is done via message passing. With message passing, a copy of a data entity is sent to a so-called communication endpoint. An underlying network does the transport of (the copy of) the data entity. This underlying network can offer incoming data, which can be read via the communication endpoint. Message passing systems have been called 'shared nothing' systems because the message passing abstraction hides underlying state changes that may be used in the implementation of the transport.

The property 'sharing nothing' makes message coupling a very good incarnation of the separation of concerns principle. Please note that asynchronous message passing is highly preferable above synchronous message passing, which violates the separation of states principle. The system works with copies of the data, and the states of the transport are separated from the application which is producing or consuming the data. This concept complies with the separation of states principle.

In comparison with stamp coupling, stamp coupling can be realized by passing a pointer, which refers to the data structure. To implement this, both modules should share the memory address space, where the pointer is referring to. Since the concept of message coupling does not share anything, also no address space, every data passing works with copies. For this reason, message coupling is considered the most loosely coupled of all categories.

Message coupling implies additional functionality with regard to the modules which need to exchange data. To comply with the separation of concerns principle, this additional functionality should be separated from the core functionality of the collaborating modules. Consequently, while the data structure in a stamp coupling scenario – in a common address space – can be used directly by the collaborating modules, at least two connection entities are required when these modules reside in a different address space (see Section V-D)).

H. Summary of the theoretic evaluation of couplings

The existing categorization of coupling is based and ordered on how tight or how loose the discussed coupling type is. We agree that in general loose coupling is better than tight coupling, but there are more important consequences based on the different types of coupling. It is not too surprising that, following our evaluation, we discourage the use of the two most tight types of coupling, i.e., content coupling and common coupling. However, other conclusions are not based on how tight a type of coupling is. For example, control coupling is a special one, because it is the only discussed type which says something about the functionality of the connected modules. All other types says something about how these modules are coupled. Data coupling and stamp coupling are alternatives for each other, while other types can be used complementary. We highly recommend stamp coupling in stead of data coupling, because data coupling can cause combinatorial effects.

Stamp coupling can be combined with control coupling, message coupling or partly external coupling (depending on the application). Control coupling should be used for version selection only. Stamp coupling can be used as it is in cases where the collaborating modules reside in the same system. In case these collaborating modules reside in different systems, stamp coupling has to be combined with message coupling. In case the collaboration includes external entities, from which we cannot control the evolution, connection entities are necessary, which is a prerequisite to use external coupling without potentially causing combinatorial effects.

IV. Data memory address space and its borders

The discussion about message coupling illustrates that a reference to a variable in a particular address space can be seen as an occurrence of a hidden dependency. In this section, we investigate this more in depth, and discuss several software constructs which have a relation with one or more memory address spaces.

In its most elementary form, programs are nothing but a sequence of instructions, which perform operations on one or more variables. These variables correspond to registers in the data memory of the controller, and the instructions correspond to registers in the program memory. The instructions are executed in sequential order, but instructions for selections and jumping to other instructions are available. In this elementary kind of programs, there is no explicit modularity at all, any instruction can read any variable in the program, and jumping from any instruction to any other instruction is possible. For this purpose, we had in the early ages of software development an instruction, which has become well-known: the goto-statement. Dijkstra called for the removal of the goto-statement in higher level languages [13], and this call is mainly addressed. However, the JMP (jump) instruction is still available in the lower level language Instruction List (IL) of the IEC 61131-3 standard for PLC (Programmable Logic Controller) programming [4]. Also, in surprisingly recent literature, goto elimination is still a research objective [15].

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

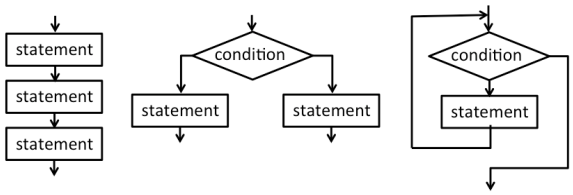


Figure 2. Concatenation, Selection, and Iteration

Alternatively, Dijkstra elaborated on the concepts concatenation, selection and iteration (Figure 2) to bring more structure in a program [16]. However, these concepts do not force modularity. In terms of Normalized Systems theory reasoning, the separation of concerns principle is not addressed. Because of the lack of clearly identifiable modules, the other theorems cannot be evaluated as well.

In this section, we discuss an amount of software constructs and how they relate to the address space, and whether the desired coupling has to cross the borders of this address space. We evaluate some concepts or paradigms based on the Normalized Systems theorems. We start our discussion with the very first attempt to build modular software systems: the 'closed subroutines' of Wilkes et al. (1959). Next, we discuss the concept of data variables, and how their scope can differ corresponding their definition. Further, we discuss variables which can be exchanged between modules. These kind of variables are typically called parameters or arguments. Two main ways how they can be passed is 'by value' or 'by reference', which will be discussed. Finally, the concepts of static and external variables will be discussed.

A. Subroutines

Wilkes et al. introduced the concept of subroutines, which they termed a closed subroutine [17]. The concept of subroutines is the first form of modularity. A subroutine, also termed subprogram, is a part of source code within a larger program that performs a specific task. As the name subprogram suggests, a subroutine can be seen as a piece of functionality, which behaves as one step in a larger program or another subprogram. A subroutine can be called several times and/or from several places during one execution of the program (including from other subroutines), and then return to the next instruction after the call once the subroutine's task is done (Figure 3).

Dijkstra reviewed the concept of subroutines in [16]. Following this review, the concept of subroutines served as the basis for a library of standard routines, which can be seen as a nice device for the reduction of program length. However, the whole program as such remained conceived as acting in a single homogeneous store, in an unstructured state space; the whole computation remained conceived a single sequential process performed by a single processor ([16], p. 46). In other words, the subroutine shares its data

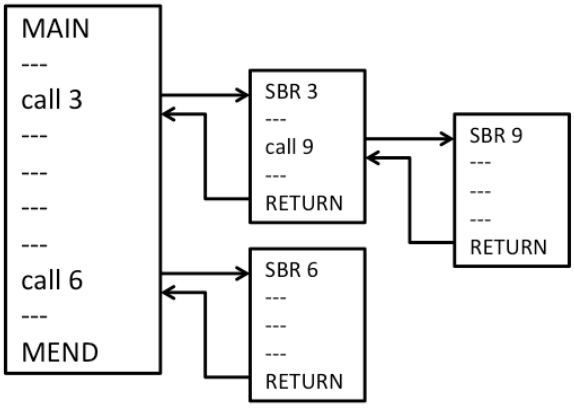


Figure 3. Subroutines

memory address space with the main program and other subroutines (if these exist). The return address of a closed subroutine can not be seen as a parameter. Rather, it looks like a well-placed jump.

In terms of Normalized Systems, progress is made towards the separation of concerns principle, but it is not fully addressed yet. Indeed, the details of the functionality in a subroutine is separated from the main program (which can be seen as a desired hiding of information for the reader of the main program), but the data of the subroutine is not. In fact, the lack of a local data memory address space in a 'closed subroutine' implies a violation of the separation of concerns principle. On the side of functionality the concerns 'main program' and 'closed subroutine' are separated, but on the side of data these concerns are not separated. Because of the lack of separation of data memory address space, the separation of states principle cannot be met. The separation of states principle implies the buffering of every call to another module. As such, when the called module does not respond like expected, the calling module can handle the unexpected result based on the buffered state. In other words, every module needs its own local memory to store its state.

B. Variables

A variable is a storage location and an associated symbolic name, which contains a value. Note that this concept is very explicit exemplified in contemporary Simatic S7 PLCs, where the programmer can choose for usage of *absolute addresses* and *symbolic addresses* [18]. In this specific environment, the programmer has to manage the data memory address space. For computer scientists, this might look old-fashioned, but for contemporary PLC programmers this is an important subject. Moreover, data memory address space cross references are tools which are commonly used to heuristically prevent combinatorial effects caused by common coupling. More general, the variable name is the usual way to reference the stored value, and a compiler

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

is doing the data memory allocation and management by replacing variables' symbolic names with the actual data memory addresses at the moment of compilation. The use of abstract variables in a source code, which are replaced by real memory during compilation is undoubtedly an improvement for reusability of the source code. However, when the memory is still shared throughout the whole system, these variables are called *global* variables, and require a name space management to prevent name conflicts. In other words, the problem of potential address conflicts is moved to potential name conflicts. In terms of Normalized Systems, when modules need global variables to exchange data, this is not really an improvement in relation to the concept of closed subroutines of Wilkes et al. ([17]).

A group of research computer scientists abandoned the term 'closed subroutine' and called modules 'procedures' in the ALGOL 60 initiative [19]. The main novelty was the concept of *local* variables. In terms of memory address space, the concept 'scope' was introduced, i.e., the idea that not all variables of a procedure are homogeneously accessible all through the program: local variables of a procedure are inaccessible from outside the procedure body, because outside they are irrelevant. What local variables of a procedure need to do in their private task is its private concern; it is no concern of the calling program [16]. In terms of Normalized Systems, local variables contribute in addressing the separation of concerns principle. A point of potential common coupling is still the fact that global variables –which are declared outside the module– are still accessible from the inside of the module. When these global variables are used in the module, without documenting this for the user, we have a violation of the separation of concerns principle. The use of undocumented and thus invisible or hidden global variables in a module makes it impossible to evaluate compliance with the Normalized Systems theorems. In other words, code analyses or white box inspection is needed to decide whether the module can be (re)used in a specific memory environment. Providing a list of the used global variables in the module documentation would be an improvement, but passing the global variables to the module as parameters or arguments is even better. The reason why this is better, is because of a better separation of the local and global address space.

C. Parameters and arguments

Having a local data memory address space contributes in separating concerns, but since the aim of software programs is generally performing operations on data entities, we should be able to exchange data between these separated memory address spaces. The question is: how should this be done? In principle, there are two possible approaches: or we exchange data by way of global variables, or we use a modular interface, which consists of input- and output parameters or arguments.

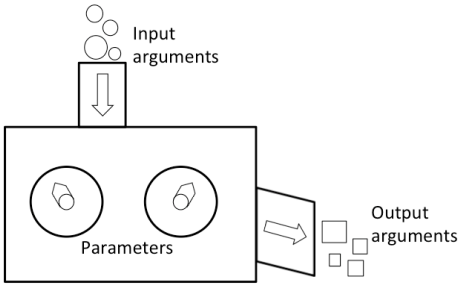


Figure 4. Function machine with parameters and arguments [20]

The terms parameter and argument are sometimes easily used interchangeably. Nevertheless, there is a difference. We use the function machine metaphor to discuss how functionality can depend on parameters (Figure 4) [20]. The influence of parameters should be seen as a configuration of the functionality, while the arguments are, following this metaphor, the material flow. This can also be exemplified with a proportional-integral-derivative (PID) controller. A PID controller calculates an 'error' value as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error by adjusting the process control inputs. The proportional, the integral and derivative values, denoted P, I, and D, are parameters, while the measured process value and the setpoint are the arguments.

From a software technical point of view, it is not important to treat parameters and arguments different when these values are exchanged between modules. However, from an application point of view, they should be aggregated differently. Like discussed in Section II, the functionality and data should be encapsulated as action entities and data entities, respectively. Since it is imaginable that the configuration of functionality (parameters) changes independently of a potential change of, e.g., the data type of the arguments, these data constructs should be separated following the separation of concerns principle. Also, the action entities, which manipulate configuration data entities, should be separated from action entities, which manipulate process data entities. Besides, the user access rights might be different, e.g., adjusting the configuration should be done by maintenance engineers, while process data might be manipulated by system operators. For simplicity reasons, in what follows, we use the term 'data passing' for both cases, in the assumption that the manipulation of arguments and parameters is separated in different modules. These separated submodules should collaborate based on stamp coupling. In its simplest form, the data structures which can be used for stamp coupling are called structs, records, tuples, or compound data. Conceptually, such data structures have a name and several data fields. In the next section, data objects

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

will be discussed.

To come back on our discussion about module dependencies, data passing can be based on a shared data memory address space between the calling and the called module (i.e., via global variables), or on the module's interface (i.e., via in/out variables). When we put ourselves into the position of a software engineer, who want to reuse a module, both the module and the definition of the global variables should be copied before the module can be reused. More specifically, to not create unused global variables in the target system (or to minimize potential name conflicts), the software engineer should only copy the global variable definitions, which are used in the module. It is imaginable that this is not in all situations straightforward, unless we provide a list or declaration of all used global variables as a documentation of the module. When the software engineer, into the process of module evolution, considers to change the module, any change on one or more of the used global variables, requires a corresponding change in the global variable definitions of the system. In case the global variables are also used in other modules, the need to perform a corresponding change in each of these modules is an occurrence of a combinatorial effect. In terms of Normalized Systems, passing data by way of global variables (common coupling) is a violation of the separation of concerns principle. Adding a global variable could be deemed to comply with the version transparency theorems, but this could be not so convenient if more engineers are working on the same project, and the chance on naming conflicts increases compared to the potential addition of a local variable.

To prevent these disadvantages, passing data by way of in/out variables, i.e., the module's interface, is more convenient and increases maintainability. The module as a construct is a way to separate the address space of the module with the address space of the 'outside', and the module's interface performs the function of a managed gateway for data passing. The reusability of the module is improved when strictly using local variables or in/out variables. However, other dependencies are still a point of interest, which will be discussed in the next section.

D. Pass by value or by reference?

Data passing *by value* means that an input variable is copied to an internal register of the module, and return *by value* means that a produced value is stored in an internal register, and copied to an output variable at the end of the processed functionality. In contrast, passing and return *by reference* means that the in/out variable is stored in a memory space outside the module, while only a reference or address to this memory space is used in the module. The in/out variable is never copied because the link with the memory outside the module remains available during the processing of the functionality.

It is not too surprising that, following our evaluation, data passing by value is isolating and separating the inside of the module better from the outside than if the same set of in/out variables would be passed by reference. In other words, in the case of pass by reference, the memory address space, which is surrounding the module, is a dependency of the module. To eliminate combinatorial effects, any dependency needs some attention. However, in this case, the dependency of memory address space is not necessarily causing combinatorial effects. In case the coupled modules reside in the same memory address space, passing parameters by reference does not cause combinatorial effects. In other words, one must make sure that the coupling is not crossing the borders of the memory address space of the considered system, which is 'hosting' the coupled modules. In case the coupling is crossing the borders of the memory address space, it has to be combined with message coupling, which implies data passing by value.

In an IEC 61131-3 environment, the length of arrays and strings are explicitly defined. This is safer in comparison with systems where this length is flexible at runtime. Note that a 'by reference' in/out variable is a pointer to the start memory address of a variable. When there is flexibility about the end address of this memory variable —e.g., an array with no explicit defined length— the pointer+index might refer to an address outside the scope of the intended variable. There is a risk that this situation becomes similar to content coupling. However, a lot of software systems tackle this problem by means of exception handling.

When we evaluate the choice between 'pass by value' or 'pass by reference' based on the Normalized System theorems, 'pass by value' contributes better towards the separation of concerns principle, by copying in-variables from the 'outside' to internal registers, and copying internal registers to out-variables after processing the functionality. In/out variables which are passed by reference always maintain a reference in the external address space, which can be seen as a dependency. Since this type of dependency can be automatically managed for every individual variable by the compiler —by way of memory (re)mapping during compilation— we do not call this dependency a violation of the separation of concerns principle from the point of view of the application software engineer. However, the approach has its limitations.

Kuhl and Fay emphasized that a static reconfiguration, which requires a complete shutdown of a system, is more costly than a dynamic reconfiguration, which can be performed without a complete shutdown [21]. Since we do not have control about how a compiler is doing the memory (re)mapping of (the reference address of) in/out variables which are passed by reference, we should assume that a dynamic configuration is limited by the data memory address space. More specifically, when a change is introduced in a module which processes in/out variables by reference, a

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

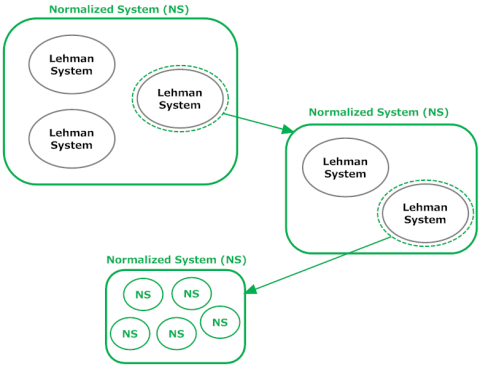


Figure 5. Different levels of modularity [22]

memory remapping of the surrounding system is necessary, and thus requires a shutdown of this system.

It is important that the application engineer is aware of this discussed limitation, especially when the choice has to be made to pass by reference or not. One should be aware that copying pass-by-value-variables costs processor time and memory space (which can be even more than strictly required when applying stamp coupling). Remember that the Normalized Systems authors advocate a higher granularity, i.e., smaller modules with the consequence that – for the same functionality– the amount of modules increases, including the (amount of) modular interfaces.

The definition of the theorem 'separation of concerns' has a focus on separation of 'tasks' (Section II), which might be interpreted as a separation of functionality. However, a concern can also be interpreted as a data memory address space, let it be on a different level of aggregation. More specifically, separation of functionality is advantageously on the lowest level of modularity, —decisions are supported with the concept of change drivers— but on a higher level the technical environment, e.g., the data memory address space, might be considered a concern. In other words, we propose that higher level constructs (aggregating one or more entities) can use the concept of passing by reference internally to let entities communicate mutually by way of stamp coupling, reusing the same interface for every entity. This might limit the consequences of the higher granularity by enabling the reuse of modular interfaces. More levels of this design might be possible in cascade, like suggested in the migration scenario's in Figure 5 [22].

E. Static and external variables

In his thinking on the recursive procedure, Dijkstra praised the concept of local variables, but he also mentioned the shortcoming of life-time of local variables. Local variables are 'created' upon procedure entry, and cease to exist when the procedure ends. The fact that local variables relate to an instantiation and only exist during that specific instantiation makes it impossible for the procedure to transmit information behind the scenes from one instantiation to the next ([16], p. 48). In this paper, we do not wish to advocate recursive procedures, but we do emphasize that the concept of static local variables (i.e., local variables which can remember their state of the previous run or incarnation) is advantageous towards the separation of states principle. The term *static* refers to the fact that the memory for these variables are allocated statically –at compile time– in contrast to the local variables, whose memory is allocated and deallocated during runtime. This concept is clearly exemplified in [18], where local (temporal) variables in a module of the form FC (Function) cannot remember their previous state, and local (static) variables in a module of the form FB (Function Block) can. For storing static variables, this type of PLCs use dedicated data memory constructs they call Data Blocks (DBs). In the case they connect such a DB to an FB they call it an *instance* DB.

The concept of external variables requires some explanation concerning definition and declaration. The *definition* of global variables decides in which memory address space they can be used, and the *declaration* of these global variables in the documentation of a module informs the potential user of the module that these global variables are needed to be able to use the module. The definition of a variable triggers the compiler to allocate memory for that variable and possibly also initializes its contents to some value. A declaration however, tells the compiler that the variable should be defined elsewhere, which the compiler should check. In the case of a declaration there is no need for memory allocation, because this is done elsewhere. The VAR_EXTERNAL keyword in an IEC 61131-3 environment indicates that the following variable is declared for the module where this keyword is used, and defined elsewhere (probably global).

Unfortunately, following a study of de Sousa, the details of defining global variables and declaring external variables are discussable to the letter of the IEC 61131-3 standard [23]. This author even doubt whether it is advantageous to have the possibility of external variable declarations within function block declarations, because passing a global variable via the keyword VAR_IN_OUT has a similar effect. In earlier work, we also advocated the use of in/out variables in an IEC 61131-3 project [9], but still, when we evaluate the concept of external variables based on the Normalized Systems theory, the explicit declaration of the use of global variables in a module eliminates potential combinatorial effects caused by common coupling. In this context, it is interesting that de Sousa considered VAR_EXTERNAL variables as belonging to the interface ([23] p. 317).

V. CONSTRUCTS FOR FUNCTIONALITY

In the previous section, we discussed mainly the concerns of data memory, and also how data memory relates to the first type of software modules, 'closed subroutines',

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

and its successor 'procedures'. The latter can have local variables, and an interface. The modular interface consists of a name for the procedure, and the input and output data variables, which are preferably data structures. We now discuss some other types of modules, which can be considered as extensions of the concept of the procedure and its interface.

A. Object-Oriented programming

The main new construct for implementing modules in object-oriented languages is the class. A class consists of both data variables (member variables) and functionality (methods). Methods can have their own local variables, but can also access the member variables and other methods of the class it belongs to. To allocate data memory and enable the methods to really work, a class needs to be instantiated or constructed to make an object. Objects of the same class can co-exist. Data and functionality are tightly coupled in an instance (object). Methods which are declared as public, are visible for other objects. Memory variables are normally considered as private to the class and, therefore, invisible for other objects. The interface of a method consists of a name for the method, and input and output variables. An object-oriented design consists of a network of objects calling methods of other objects, which can be implemented as data coupling or stamp coupling.

Since each method has its own interface, and a class can contain multiple methods, an object as a module can have multiple interfaces. Classes can be extended with the concept of inheritance. This concept envisaged to mimic the concept of ontological refinement. Just like a bird is a special type of animal, and a sparrow a special type of bird, inheritance was created to define classes as refinements of other classes. Such a subclass would inherit the member variables and methods of a superclass, and extend it. However, Mannaert and Verelst state that in practice, very few programming classes are in line with the assumption that object-oriented inheritance is based on ontological refinements ([2], p. 29). If we cannot count on ontological refinements, a class can also be seen as just an amount of methods, grouped together based on the intuition of the programmer, and sharing the same set of member variables. When the size of such a class grows, the situation becomes comparable with a system based on procedures, having their own local variables, but sharing the system's global variables.

In terms of Normalized Systems, we evaluate that the object-oriented programming paradigm is not guaranteeing compliance with the separation of concerns principle. First, in case the data type or data representation can change independently from the functionality, the tight coupling between data and functionality makes version transparency not straightforward. For example, consider that in an application, a house-number-field changes its data type from numeric to alpha-numeric, without any functional change. The datatype change might require the functionality to change, too. As such, it seems possible that combinatorial effects occur, which makes version transparency infeasible when the size of a system grows. Second, when the size of a class grows, the member variables are similar with (class-wide) global variables. Consequently, common coupling between methods is imaginable and combinatorial effects can occur. As a remedy, this dependency could be made explicit by declaring the use of every member variable in a method by way of declaration concept similar to the the declaration of external variables. Indeed, from the point of view of a method, a class member variable can be seen as 'external'.

Public methods can be called via their interface, as if they make part of the programming environment. However, they belong to a class. If someone wants to reuse such a method in another system, at least the 'hosting' class should be copied as well. In addition, other classes which contain coupled methods should be copied, too (note that a class can contain methods, from which the code include the construction of objects, based on other classes). In other words, public methods, which reside in classes, are available in a flat name space. Any public method can call any other public method, which can result in a complex network of calling and called method, residing in the same or different objects. In an evolving system, the required version management between the calling and called (public) methods (with additionally tightly coupled data), is not straightforward. To be able to keep track of all couplings, including the versions of these methods, we propose a similar explicitation like we did for memory variables. The method interface should include a declaration or documentation part, which informs the user of all methods which are called inside the method, including the object and class version to which they belong. This declaration might be done in a similar way as the declaration of external variables, i.e., the announcement that one or more functional constructs are used or called in the code of the concerning method. In terms of Normalized Systems, we evaluate that methods and classes might comply with the separation of concerns principle, but extra constraints are necessary. There should be only one 'core'-method containing the core functionality of the class, surrounded by supporting methods like cross-cutting concerns. Also version transparency should be an extra constraint when using the object oriented paradigm.

The concept of inheritance does not guarantee version transparency, because it is based on an anthropomorphical assumption, which is not realistic in all cases. It would be better to implement explicit version management, based on version IDs. This version management should be twofold: first, the versions of data memory entities (including type or representation) should be made explicit, and second, the versions of the functionality, how the versions of data memory entities relate to the versions of functionality and vice versa should be made explicit as well.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

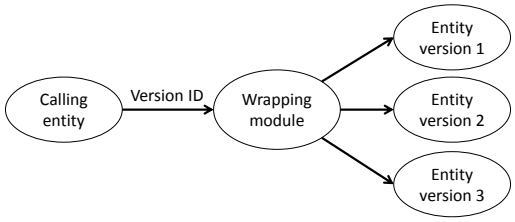


Figure 6. The concept of version wrapping

We do discuss some potential drawbacks of the object-oriented paradigm, but we emphasize that it is possible to build evolvable systems, based on the object-oriented paradigm, complying the Normalized Systems theorems. However, the object oriented paradigm itself does not guarantee the property of evolvability. Additional constraints are necessary to eliminate combinatorial effects. One of the key remarks is that an object should not contain more than one core functionality, and functionality should be separated from data representation. One of the possibilities is the introduction of data objects and functional objects. In addition, the use of memory variables and methods in a method should be declared on a similar way like the concept of external variables. We also think that polymorphism, combined with explicit version management might be an alternative for inheritance. This alternative could exhibit version transparency, but more elaboration and future work is needed to figure this out.

B. Modules in IEC 61131-3

In an IEC 61131-3 environment, we have Functions (FCs), which have in addition to the input and output variables only temporary local variables. The Function Block (FB) construct can have static local variables, too. More general, these constructs are called Program Organization Units (POU), and are stored in a flat program memory space. On the same level global variables and derived data types are defined (in IEC 61131-3 terms, as a *configuration* definition). Note that, besides the functionality, FBs need data memory before they can actually run. Several FB instances can co-exist with separated data memory. This concept is very similar to the object-oriented paradigm. Indeed, Thramboulidis and Frey state that the Function Block concept has introduced in the industrial automation domain basic concepts of the object oriented paradigm [24]. There is a restriction in the behavior definition of the FB: only one method can be defined. There are no method signatures as in common object oriented languages; actually there is no signature even for this one method defined by the FB body. This method is executed when the FB instance is called [24][4]. Note that the object oriented extension of the FB construct that is under discussion in IEC is not considered in this paper.

Polymorphism is not supported in version two of the IEC 61131-3 standard, nor is inheritance [4]. In a commercial IEC 61131-3 environment, the only way to implement version management is doing this explicitly. In earlier work, we proposed the concepts Transparent Coding and Wrapping Functionality [9]. Transparent coding is defined as the writing of internal code in a module which is not affecting the functionality of previous versions. When Transparent Coding is not possible (e.g., because of conflicting functionality of the versions, or when the combination of the functionality of different versions requires too complex code), Version Wrapping can be applied. Following this principle, different versions of a module co-exist in parallel, and a wrapping module selects the desired version based on the version ID (see Figure 6).

As a reflection with regard to the general object oriented paradigm, it is straightforward to implement only one core functionality in an (IEC 61131-3) FB, because following the analysis of Thramboulidis and Frey only one method is defined in a FB [24]. However, software application engineers tend to extend the possibilities of FBs by way of control coupling. In other words, it is possible to select different functionality based on parameters. In terms of Normalized Systems reasoning, control coupling should be restricted to version selection only. In this way, several versions can co-exist, but still not more than one core functionality resides in one module.

We also reflect on the issue of separation of data and functionality. If we would do this rigorously and strict, we would abandon the use of FBs and stick to the use of FCs only, because FBs can have static variables, and FCs cannot. This also implies that FBs can call other FBs, but FCs cannot call FBs. Indeed, FCs cannot instantiate FBs because they can not allocate the static memory FBs require in syntactical sense. However, we do advocate the use of FBs, because we think it is advantageous to separate technical data, which can be tightly coupled with the functionality, and content data, which has a meaning with regard to the algorithm which is processed in the functionality. For example, to detect the so-called rising or falling edges, e.g., the arriving of a bottle on a filling location, we need to remember the previous state of a sensor. The memory needed to detect these rising or falling edges is a technical matter, of which we might desire to be hidden. In contrast, the information that the event of arrival occurred, is something important for the process algorithm, e.g., to trigger the filling process of the arrived bottle. Another example is the case of the control of a valve, which includes an alarm state. The valve is operational when the feedback sensors (i.e., open or closed sensors) correspond to the output control (i.e., open or closed commands). However, the valve has a mechanical inertia, i.e., it needs some time to open or close, so having a discrepancy between feedback and control is temporary normal. Typically, a timer construct is used to temporary allow a discrepancy, while not entering the alarm state. The data needed for the technical instance of the timer construct is data we call a technical data entity,

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

which can be hidden and tightly coupled to the module which is performing the alarming algorithm. The result of the decision whether the valve is in the alarm state or not, is related to the control algorithm of the valve, and should be stored in a separated data entity, or more specifically, passed via the modular interface.

C. Libraries and packages

Libraries are collections of compiled modules, which can be shared among various application programs. In an IEC 61131-3 environment, they can also include the definition of the so-called derived data types, i.e., user defined data types, such as structs. Some libraries are called 'standard' libraries, because the content is specified in a standard (this kind of library functionality is also specified in IEC 61131-3). The functionality offered in a standard library is assumed to be widely known, and application engineers should be able to treat them as if they make part of the programming environment. However, in an IEC 61131-3 environment, the details of standard constructs might slightly differ from one brand to another, because this standard allows the so-called implementation-dependent parameters ([4], annex D).

At first sight, the concept of adding 'standard' or other constructs with a reuse potential by way of libraries sounds interesting. Indeed, when the set of shared functionality is small enough, this concept looks great. However, like Dijkstra already recognized back in 1972, one of the important weaknesses in software programs is an underestimation of the specific difficulties of size ([16], p. 2). Remember that the Normalized Systems theory emphasize the importance of separation of concerns. When we interpret a concern as a module or user defined data type, we can count on an unique identification of these constructs into the name-space borders of an individual library or package. However, when these libraries are selected in the library management tool of a programming environment, these constructs end up in a common flat name space. In other words, name space conflicts can occur when constructs of different libraries end up in the same flat module name space.

This might result in a so-called dependency-hell. This is a colloquial term for the frustration of some software users who have installed software packages, which depend on specific versions of other software packages. It involves for example package A needing package B & C, and package B needing package F, while package C is incompatible with package F. Again, when the amount of selected libraries is limited, one could avoid a dependency-hell. However, when constructs are shared between different developers, who perform maintenance activities or make extension of the same application over time, they might use constructs of the same library, but from a different library version. If it is desired that one construct of a library is used from a early version, and another construct of the same library is used from a recent version, it looks impossible to prevent dependency problems in a flat name space. Also, in [18] the modules have a number and a symbol. This number might conflict with existing modules, or with modules from another library.

To come back on the separation of concerns principle, let us interpret a concern as a library. When different libraries are selected in a programming environment, and all constructs of these libraries end up in the same construct name space, we evaluate this as a violation of the separation of concerns principle. This violation is even worse when two versions of the same library would be selected. If the name of the library is not including the version, it might be even impossible to select both. Having functional constructs or data type definitions in a flat name space is similar to common coupling. The use of a library construct in a module should be documented in order to make an evaluation whether the construct can be used in the concerning module or not. The addition of a module, which is using a conflicting name, indicates a bad separation of the constructs available in the used libraries. We derive that using modules from a library should be restricted to standardized functionality and constructs. The designers of the standard should prevent name conflicts in a similar way how keywords are reserved in a programming language. One should avoid to configure library constructs, dedicated for reuse in specific applications, in a flat name space.

As a remedy, constructs belonging to a specific library could be selected on the level of the module, not on the level of the programming environment. This would mean a kind of localization of library constructs. The declaration part of a module could include a library browser, to select a desired functionality or data type from that library. In addition, the version of constructs and libraries should always be included in the declaration part of the module. In this declaration, the 'hosting' library of a construct, accompanied with its version, should be included as a kind of path. As such, it would be even possible to use co-existing versions of a library construct in the same module, because the concerning constructs are well separated.

D. Distributed calling via messages

In an IEC 61131-3 environment or in truly object oriented languages, a module can only call other 'local' modules. Local means that they need to be available within the same program address space. Libraries are deployed locally in the sense that they are compiled and linked into the same program and memory address space. The concept of inter-process communication allows remote calls to a library or system, which is 'hosted' in another program and memory address space. Following a paper of Birrel and Nelson, remote procedure calls (RPC) appear to be a useful paradigm for providing communication across a network between programs written in a high-level language [26]. The idea of RPC is quite simple. When a remote procedure is invoked, the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

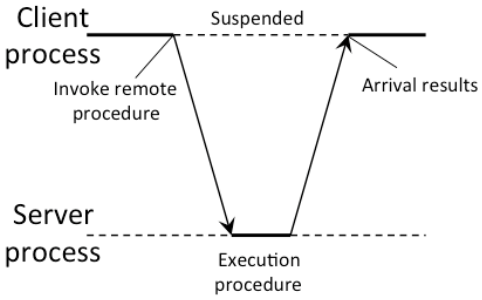


Figure 7. Principle of RPC between client and server [25]

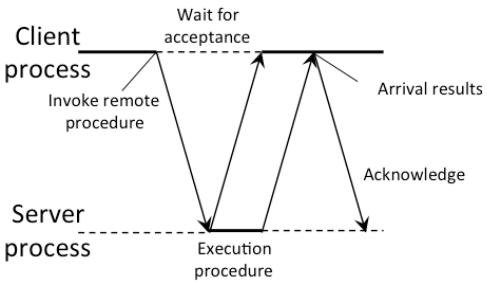


Figure 8. Deferred synchronous RPC [25]

calling environment is suspended, the parameters are passed across the network to the environment where the procedure is to be executed, and the desired procedure is executed there (Figure 7). The idea of RPC was older, but Birrel and Nelson were one of the first who implemented it [26]. This concept is further elaborated with the standards CORBA (Common Object Request Broker Architecture [27]) and DCOM (Distributed Component Object Model [28]). Also, the OPC Foundation based its first interoperability standard for industrial automation on DCOM. This first family of specifications is referred to as 'the classic OPC specifications' [29].

The ignorance on the part of the client about the fact that the server is located in a remote address space, was considered advantageous [25]. The client made use of a (local) library, which is dedicated for making a connection with a remote library, which was performing some tasks on the server side. Both libraries collaborate on a rather complicated mechanism to convert the client call to a message, and unpacking this message at the server side and convert it to a (local) call at the server side. All the details of the message passing are hidden away in the two libraries. Because of the message passing, this is message coupling, but for the user it looks like data or stamp coupling. Since the user cannot know whether there is a message coupling behind the data or stamp coupling, using or not using the concerned module cannot be a well considered choice or decision.

We evaluate that on top of the problems explained in the previous subsection about libraries and packages (subsection V-C) this concept, shown in Figure 7, is a violation of the separation of states theorem. Remember that a local module call is based on and thus dependent on the local address space. Hiding this dependency for the user also hinders the potential control over this dependency or assumption. For a local call, a fast reaction of the called module is assumed. For a remote call, the extra transfer time is not always negligible. Consequently, the suspension of the client during the call might be unfeasibly long. Also, when a communication failure occurs, the reply will not come at all, and the client will wait forever. In addition, the 'assumption' of the client that the call is local, does not discourages the user to pass variables by reference. While passing variables by reference assumes a local address space, this concept is not ideal in a remote call. When crossing the borders of a memory address space, each side of the coupling has to keep its own state. In other words, a reference to an item in an address space will become meaningless if the reference address is moved to another address space (and similar to content coupling). This would be an occurrence of a violation of the separation of concerns principle. In addition, because the value behind the reference is not copied in the respectively address spaces, we have a violation of the separation of states principle.

E. Synchronous versus asynchronous message passing

The concept of Figure 7, i.e., the client waits until the server replies before carrying on with its task, is called synchronous RPC. The action of communication on the client side can be summarized in one single line of programming code, there is a synchronization point between sender and receiver on message transfer. To minimize the 'wait for result' time, the concept of asynchronous RPC is introduced, where the client is not waiting for the reply, but only on an 'acceptance request' message. In combination with a similar call coming from the server (a so-called 'callback'), the client can receive the return results from the remote procedure in a comparable time frame as with synchronous RPC, but then without being blocked all the time (Figure 8). In comparison with synchronous communication, asynchronous communicates requires buffering to enable the program proceeding at the client side between request and reply. Before indicating this as an disadvantage, one should be aware that this buffering is exactly what the separation of states principle calls for. However, this principle is still not totally met, because the program at the client side can still hang when the 'acceptance request' message does not come, e.g., because of a network failure.

In the classic OPC specifications, both synchronous and asynchronous reading/writing functionality is available. However, experts indicated as a heuristic rule that asyn-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

chronous communication is preferable. Indeed, the authors of the new family of interoperability standards for industrial automation, i.e., the OPC Unified Architecture (OPC UA), have abandoned the synchronous communication concept [30]. Instead, the OPC UA based communication is asynchronous by definition [31]. In terms of Normalized Systems, asynchronous communication reaches further towards complying the separation of states principle. In DCOM, there was an attempt to handle the risk that the client hangs when the 'acceptance request' message does not come by introducing a time-out mechanism. However, experts of the OPC Foundation reflected, based on worldwide surveys, that practitioners still call this an issue (note that classic OPC is based on DCOM). Lange et al. state that the time-out of DCOM in case of communication failures is too long, and not configurable [32].

We evaluate further that RPC, and DCOM, do not exhibit version transparency. Any change to a server requires all (remote) clients to have corresponding updates. When the size of a (distributed) system grows, this becomes infeasible because of the occurring combinatorial effects.

F. Service based communication

Services are modular constructs for aggregating software. Internally, they consist of modules, and they have one or more modular interfaces, that is accessible to the outside world. The basic idea is that some client application can call the services as provided by a server application. This principle is very similar to what was aimed at with remote procedure calls, except that the message coupling part is not hidden for the user. Services were first proposed in terms of web services, as they adhere to a collection of standards that will allow them to be discovered and accessed over the Internet. However, the term service has become more broadly interpreted later on. A service refers to technology-independent modules, implementable in different ways, including web services.

Web services are described by means of the Web Service Definition Language (WSDL) which is a formal language, comparable with the interface definition languages used to support RPC-based communication. A core element of a web service is the specification of how communication takes place. To this end, the Simple Object Access Protocol (SOAP) is used, which is essentially a framework in which much of the communication between two processes can be standardized [25]. Strange as it may seem, a SOAP envelope does not contain the address of the recipient. Instead, SOAP specifies bindings to underlying transfer protocols. In practice, most SOAP messages are sent over HyperText Transfer Protocol (HTTP). All communication between a client and server takes place through messages. HTTP recognizes only request and response messages. For our evaluation, a key field in the request line of the request message and status line of the response message is the version field. In other words, HTTP exhibits version transparency. Client and server can negotiate with the 'upgrade' message header on which version they will proceed. SOAP is designed with the assumption that client and server know very little of each other. Therefore, SOAP messages are largely based on the Extensible Markup Language (XML), which is on top of a markup language also a meta-markup language. In other words, in an XML description the syntax as used for a message is part of that message. This makes XML more flexible than the fixed markup language HyperText Markup Language (HTML), which is the most widely-used markup language in the Web.

Web services can be considered as a successor to RPC, like OPC UA (based on services) is a platform- and technology independent 'alternative' for classic OPC (based on DCOM). We doubt to use the word 'alternative' here, because classic OPC and OPC UA are complementary. Indeed, services can internally consist of classes or components, including DCOM based constructs. Web services separate software components from each other. They enable self-describing, modular applications to be published, located, and invoked across the web. Being a standardized interface, OPC UA enables interoperability between automation systems of different vendors. The industrial working groups of the OPC Foundation introduced *a mechanism to bring interoperability on an abstract level, without leaving the practical implementability*. To achieve this ambitious goal, they emphasized the importance of a communication context, and made a connection management concept between clients and servers mandatory. Probably OPC UA is also implementable for interoperability in other sectors than industrial automation [31].

The concept of asynchronous web-based messaging allows clients to proceed functioning, even if the server does not respond. From a technical point of view, a client can just carry on based on its own state. From a functional point of view, OPC UA incorporated mechanisms of notification and keep-alive messages to enable handling communication or remote system failures. This complies with the separation of states principle. The version tag in the HTTP messages enables compliance with the version transparency theorems.

VI. Summary of evaluations and guidelines

The core recommendation of this paper is making hidden dependencies explicit in the module's interface. In other words, safe black box (re)use requires that a developer is able to anticipate which conditions are necessary for (re)use. A self-explaining interface is a good start, but typically dependencies like packages, libraries, global variables, implicitly used communication technologies, references to a local address space, are not included in the interface. We conclude that it should, and phrase the following rule.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

In order to design safe black box (re)useable software components, every (re)use of a library, package, global variable or implicitly use of a communication technology in a module, should include a declaration, reference, path or link to the identification of the dependency, accompanied with the used version.

We make the reflection that there is a similarity between global variables, which are not declared with the 'external' keyword and other dependencies, which are not declared in the module's interface. It can be interpreted that these dependencies can cause common coupling. Hiding these dependencies makes it impossible to evaluate them and let the user decide whether these dependencies can or cannot be made available in the environment in which the user is considering them to (re)use. Note that declarations to make these dependencies visible should include the versions of the external constructs, to prevent combinatorial effects in case of updates, and to enable the co-existence of different versions of the same core constructs in a library or external technology.

In addition to our rather general rule, we define some explicit guidelines:

1) Explicitation of global variables: Global variables should be treated as local variables of the main program, and passed to called modules by reference or via the in/out variables in an IEC 61131-3 environment. These variables could be passed further in cascade to submodules called by modules, where they are locally always treated as in/out variables.

Application example: Consider an IEC 61131-3 Function Block which is controlling a motor. This Function Block (FB) is calling other FBs on submodular lever, where the core functionality is a state machine of the motor. In addition, there are supporting FBs on submodular level, which provide functionality to manage manual/automatic mode, alarming, interlocking, hardware connection, and simulation. The FB on modular level (dispatching task) receives a data struct, which contains all the states, commands, and hardware IOs of both core and supporting functionality. This data struct is a global variable. The dispatching FB calls FBs on submodular level and passes the data struct to each of the supporting FBs as an in/out variable. This design has a modular structure with a high granularity. Since the functionality of the FBs on submodular level is limited and generic, the reuse potential is high.

2) Pass by reference should strictly adhere to one single address space: In/out variables, passed by reference, loose their meaning in another address space. Therefore, the pass by reference concept should be limited to the same environment or address space where the referred variable is defined. In case it is desired to cross the borders of the address space, a copy of the concerned variable or a pass by value is required.

Application example: Consider the same data structure which contains all the data about a motor. This data structure is defined as a global construct, and is passed to the dispatching FB by reference. This reference is passed further on submodular level to the supporting FBs. Now, outside the PLC, a low level HMI (Human Machine Interface) application is used to control the motor on submodular level on a Windows PC. This Windows PC cannot use the reference, which is only meaningful in the PLC. Instead, the entire data structure is copied via an OPC interface (message coupling) to the HMI application.

3) Explicitation of external modules: Couplings to external modules can be (re)used, library modules included, but they should be declared in a similar way like the 'external' keyword for global variables, including the path of the communication context. In other words, library management should be done on the level of the module, not on the level of the programming environment. In addition, the versions of the called modules should be declared.

Application example: Our data structure is defined as a global IEC 61131-3 configuration. In the main program, this is not visible, unless this data structure is declared as an external defined data structure in the main program (POU). As such, the data structure can be treated as local for the main program.

4) Abstraction of external technologies: It is allowed to hide information about an external technology, but an abstraction of the core functionality should be declared, including the fact that this functionality is abstract, and relying on a remote technology. The entity which is managing the connection with this abstract remote technology should exhibit state keeping, and notify autonomously unexpected behavior of the remote technology.

Application example: Suppose the motor is controller with a frequency drive. We do not have control over potential firmware updates of this frequency drive. It is also possible that at some moment in time the frequency drive will be replaced by another type or brand. Therefore, we include in data struct fields which are representing the core functionality like setpoint, ramp, speed, current, etc. A connection entity is responsible to convert the representation or data type of these fields. For every version another connection entity has to be written. A connection element selects the appropriate version based on a version ID.

VII. Conclusion

The reasons why properties like evolvability, (re)usability, and safe black box design are difficult to achieve, have most likely something to do with a lack of making the existing knowledge and experience-based guidelines on sound modular design explicit. Undoubtedly, the theorems of Normalized Systems contribute on this issue by formulating unambiguous design rules at the elementary level of software primitives. On a higher implementation level, it is expected that

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

not all implementation questions like those related to, e.g., a dependency-hell, are easy to answer. Experienced engineers will find that these are violations of the theorems 'separation of concerns' and 'separation of states'. However, for less experienced engineers, more practical oriented examples or manifestations of violations and how to avoid them, seem useful as well. We aim that — on top of these fundamental principles — some derived rules can make these violations easier to catch, also for less experienced engineers.

In this paper, we introduced the derived rule that any dependency should be visible in the module's interface, accompanied by its state and version. The way how this information is included in the interface, should be done in a version transparent way, to prevent violations of the 2nd and 3rd principle of Normalized Systems.

We made a study of a set of different kind of couplings on an abstract way, and evaluated these types of couplings against the Normalized Systems theorems. In addition, implications arise when modules are placed in an address space, based on a paradigm or construct in a concrete programming environment. Special attention is needed when a module, placed in the local address space, is coupled with another module, which is placed in a remote address space. After evaluating these implications, we derived four guidelines towards better controlling dependencies.

We designed the derived rules with the potential to become generic, independent of the application domain. As a first start, we exemplified the rules and analyses in a PLC (IEC 61131-3 based) environment. In future work, our aim is to investigate to which extent these rules can be implemented in other technologies and programming environments as well.

ACKNOWLEDGMENT

P.D.B. is supported by a Research Grant of the Agency for Innovation by Science and Technology in Flanders (IWT).

REFERENCES

[1] D. van der Linden, H. Mannaert, and P. De Bruyn, "Towards the explicitation of hidden dependencies in the module interface," in *ICONS 2012, 7th International Conference on Systems*, 2012.

[2] H. Mannaert and J. Verelst, *Normalized Systems Re-creating Information Technology Based on Laws for Software Evolvability*. Koppa, 2009.

[3] M. McIlroy, "Mass produced software components," in *NATO Conference on Software Engineering, Scientific Affairs Division*, 1968.

[4] IEC, *IEC 61131-3, Programmable controllers - part 3: Programming languages*. International Electrotechnical Commission, 2003.

[5] H. Mannaert, J. Verelst, and K. Ven, "Exploring the concept of systems theoretic stability as a starting point for a unified theory on software engineering," in *ICSEA 2008, 3th International Conference on Software Engineering Advances*, 2008.

[6] M. Lehman, "Programs, life cycles, and laws of software evolution," *Proceedings of the IEEE*, vol. 68, pp. 1060–1076, 1980.

[7] H. Mannaert, J. Verelst, and K. Ven, "The transformation of requirements into software primitives: Studying evolvability based on systems theoretic stability," *Science of Computer Programming*, vol. 76, no. 12, pp. 1210 – 1222, 2011.

[8] ——, "Towards evolvable software architectures based on systems theoretic stability," *Software: Practice and Experience*, vol. 42, no. 1, pp. 89–116, 2012.

[9] D. van der Linden, H. Mannaert, W. Kastner, and H. Peremans, "Towards normalized connection elements in industrial automation," *International Journal On Advances in Internet Technology*, vol. 4, no. 3&4, pp. 133–146, 2011.

[10] G. Myers, *Reliable Software through Composite Design*. Van Nostrand Reinhold Company, 1975.

[11] Wikipedia, "Coupling (computer programming)," *Wikipedia*, last accessed June 2013. [Online]. Available: http://en.wikipedia.org/wiki/Coupling_(computer_programming)

[12] D. Van Nuffel, H. Mannaert, C. De Backer, and J. Verelst, "Towards a deterministic business process modelling method based on normalized theory," *International journal on advances in software*, vol. 3, no. 1 and 2, pp. 54 – 69, 2010.

[13] E. Dijkstra, "Go to statement considered harmful," *Communications of the ACM 11(3)*, pp. 147 – 148, 1968.

[14] S.-M. Huang, C.-F. Tsai, and P.-C. Huang, "Component-based software version management based on a component-interface dependency matrix," *Journal of Systems and Software*, vol. 82, no. 3, pp. 382 – 399, 2009.

[15] T. D. Vu, "Goto elimination in program algebra," *Science of Computer Programming*, vol. 73, no. 2 - 3, pp. 95 – 128, 2008.

[16] E. W. Dijkstra, "Structured programming," O. J. Dahl, E. W. Dijkstra, and C. A. R. Hoare, Eds. London, UK, UK: Academic Press Ltd., 1972, ch. Chapter I: Notes on structured programming, pp. 1–82.

[17] D. J. W. Maurice V. Wilkes and S. Gill, *The preparation of programs for an electronic digital computer*. Addison-Wesley Press, 1951.

[18] *Programming with STEP7*, Siemens, 05 2010.

[19] "ALGOL 60," last accessed June 2013. [Online]. Available: http://en.wikipedia.org/wiki/ALGOL_60

[20] D. Nykamp, "Function machine parameters," last accessed June 2013. [Online]. Available: http://mathinsight.org/function_machine_parameters

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[21] I. Kuhl and A. Fay, "A middleware for software evolution of automation software," *IEEE Conference on Emerging Technologies and Factory Automation*, 2011.

[22] D. van der Linden, G. Neugschwandtner, and H. Mannaert, "Industrial automation software: Using the web as a design guide," in *ICIW 2012, 7^{th} International Conference on Internet and Web Applications and Services*.

[23] M. de Sousa, "Proposed corrections to the IEC 61131-3 standard," *Computer Standards & Interfaces*, pp. 312–320, 2010.

[24] K. Thramboulidis and G. Frey, "An MDD process for IEC 61131-based industrial automation systems," in *Emerging Technologies Factory Automation (ETFA), 2011 IEEE 16th Conference on*, sept. 2011, pp. 1 –8.

[25] A. Tanenbaum and M. Van Steen, *Distributed Systems: principles and paradigms*. Pearson Prentice Hall, 2007.

[26] A. D. Birrell and B. J. Nelson, "Implementing remote procedure calls," *ACM Transactions on Computer Systems*, vol. 2, no. 1, pp. 39 – 59, 1984.

[27] "Corba," last accessed June 2013. [Online]. Available: http://www.omg.org/spec/CORBA/

[28] G. Eddon and H. Eddon, *Inside Distributed COM*. Microsoft Press, 1998.

[29] *OPC DA Specification*, OPC Foundation Std. Version 2.05a, 2002.

[30] "OPC Unified Architecture Specifications," last accessed June 2013. [Online]. Available: http://www.opcfoundation.org

[31] W. Mahnke, S. H. Leitner, and M. Damm, *OPC Unified Architecture*. Springer, 2009.

[32] J. Lange, F. Iwanitz, and T. Burke, *OPC From Data Access to Unified Architecture*. VDE-Verlag, 2010.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Magnitude of eHealth Technology Risks Largely Unknown

An Exploratory Study into the Risks of Information and Communication Technologies in Healthcare

H.C. Ossebaard[1, 2], J.E.W.C. van Gemert-Pijnen[2], A.C.P. de Bruijn[1] and R.E. Geertsma[1]
hans.ossebaard@rivm.nl , j.vangemert-pijnen@utwente.nl , adrie.bruijn@rivm.nl , robert.geertsma@rivm.nl
[1] RIVM - National Institute for Public Health and the Environment, Bilthoven, The Netherlands
[2] University of Twente, Enschede, The Netherlands

***Abstract* – Many believe that eHealth technologies will contribute to the solution of global health issues and to the necessary innovation of healthcare systems. While this may be true, it is important for public administrations, care professionals, researchers, and the general public to be aware that new technologies are likely to present new or uncertain risks along with their great new opportunities. The present paper aims to assess the risks of eHealth technologies for both patient safety and quality of care. A quick-scan of scientific literature was performed as well as an analysis of web-based sources and databases. Outcomes were validated in a focus group setting against expert views of stakeholders from health care, patients' organizations, industry, academic research, and government. Risks at human, technological or organizational levels appear to be no subject of systematic research. However, they come into view as 'secondary' findings in the margin of these studies. Extensive anecdotal evidence of risks is reported at all three levels in web-based sources as well. Recent authoritative reports substantiate these outcomes. Members of the focus group generally recognized the findings and provided valuable, additional information. A realistic approach to the implementation of eHealth interventions is recommended, taking into account potential benefits as well as risks, and using existing risk management tools throughout the life cycle of the intervention.**

Keywords - risks; eHealth; health technology; patient safety; quality of care

I. INTRODUCTION

Trust in technology is of growing importance in view of the challenges for global healthcare [1]. Most countries face a serious increase in healthcare expenditures that corresponds to ageing, a growth in multi-morbid chronic illnesses, the enduring menace of infectious diseases, consumerism and other dynamics [2, 3]. eHealth technologies have frequently been hailed as a panacea for these challenges. We view eHealth as the use of information and communication technologies (ICTs) to support or improve health and healthcare. These technologies have proven their potential to contribute to the increase of (cost-) effectiveness and efficiency of care, the improvement of the quality of care, the empowerment of consumers, system transparency, and eventually to the reduction of health care costs [4-7]. However, expectations have recently been mitigated due to the publication of studies that emphasize the complex nature of innovation in healthcare and the lack of rigid, systematic evidence for the impact of eHealth technologies on healthcare outcomes so far [8, 9]. Moreover, the application of eHealth technologies in healthcare may introduce risks for patient safety and quality of care [10-12]. Nonetheless, trust in information and communication technologies seems to remain unaffected by these moderating results. This is remarkable against a backdrop of widespread declining trust in the legal system, in politics, finance, science and other public domains [13, 14]. Public administrations, care professionals, researchers and the general public are generally trustful and overly optimistic about the 'a-political' power of digital technology in virtually all public and personal domains [15, 16]. Common principles of evidence based medicine are apparently ignored regularly in this field, leading to fast introduction of promising eHealth interventions without carefully evaluating benefits versus risks.

Recently, we have reported on some drawbacks of eHealth technologies at another level and from a different perspective [17]. This study was based on a comprehensive analysis of eventually sixteen frameworks regarding the development and implementation of eHealth interventions over the last decade (2000-2010). The reported shortcomings are closely related to risks. Eventually, they imply equivalent and immediate hazards for the patient's safety or the quality of care. Therefore, we think it relevant for the present study to provide a short summary of these findings. Table I shows a summary of these risks phrased in conceptual terms.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE I. RISKS DERIVED FROM PREVIOUS RESEARCH*

Conceptual risk	Description
eHealth technology development as an expert-driven process	If project management fails to arrange stakeholder participation in the full development process risks for rejection by (end-)users increase.
eHealth technology development ignores evaluation	If the development is viewed as a linear, fixed and static process instead of a iterative, longitudinal research activity risks of suboptimal outcomes increase.
Implementation of eHealth technology as a post-design activity	If conditions for implementation are not properly accounted for right from the start in all subsequent stages stakeholders may drop out.
eHt development does not affect organization of healthcare	If it is ignored that eHealth technologies intervene with traditional care characteristics and infrastructure unexpected effects cause stakeholders to abandon.
eH technologies as instrumental, determinist applications	If eH interventions ignore users' needs for affective, persuasive communication and information technologies for motivation, self-management and support, they drop-out..
eH research fails to integrate mixed-methods and data triangulation	If conventional research methods keep falling short of assessing the added value for healthcare in terms of process (usage, adherence) and outcome variables (behavioral, clinical outcomes; costs) societal and scientific refutation follows.

* Van Gemert-Pijnen et al., 2011 [22]

Precisely the opposites of factors that improve the uptake and impact of eHealth technologies constitute risk for both patient safety and quality of care; they increase the probability of occurrence of harm and/or the severity of that harm. These are exactly the two components used in the internationally accepted definition for risk that we are applying in our investigation, i.e., "risk is a combination of the probability of occurrence of harm and the severity of that harm" [18]. This definition is also used in the international standard for risk management of medical devices [19], which is the regulatory sector in which part of the eHealth technologies can be classified, as well as in other standards more specifically relevant to ICT applications in health care.

In the present study, we investigate the nature and occurrence of any risk to patients´ safety and quality of care that may be associated with eHealth applications. These interventions include web-based and mobile applications for caregivers, patients and their relatives within a treatment relationship as well as technology regarding quality in healthcare. In view of the diversity and dynamics of the field, we have chosen to use multiple approaches to gather our data and to verify our findings. As a first approach, we searched for risks as established in randomized controlled trials and reported in scientific literature (see Section II). This provides an inventory of documented risks that impact on quality of care and the patients' well-being. Additionally we have searched a selection of web-based sources related to (inter)national health organizations/government agencies, incident databases, expert centers, and opinion papers in the medical field (Section III). While we were analyzing our search results, three authoritative reports with scopes closely related to our own were published, and we decided to compare their findings with our own as a method of independent control. The outcomes were eventually validated in a focus group setting against expert views of stakeholders from health care, patients' organizations, industry, academic research and government (Section IV). In Section V we present the outcomes of these approaches, to draw conclusions in the next section and discuss the in the last.

II. LITERATURE SCAN

The literature scan was designed to exploratory assess only those risks that are reliably documented in systematic studies, i.e., randomized controlled trials (RCTs). The scan was restricted to scientific publications regarding risks that affect the quality of healthcare and patient safety while public health was excluded. Issues concerning security of data-transmission, storage, encryption, standardization, data-management and privacy were excluded as well to avoid overlap and redundancy in view of other studies [20]. The search was limited to RCTs. This type of studies represents the highest power of evidence in the absence of meta-analyses or systematic reviews and allows for comparisons with alternative approaches.

The bibliographic database SciVerse Scopus was searched because of its broad content coverage including all Medline titles and over 16.000 peer-reviewed academic journals. The used search query combined the topic 'eHealth' with search terms regarding risk, healthcare-setting, and study design. The complete query can be found in Appendix I. One author reviewed the titles and abstracts of the identified publications to decide whether they should be examined in full detail. An overview of the inclusion criteria is presented in Table II. The study selection process is included in Appendix II.

TABLE II. INCLUSION CRITERIA FOR THE STUDY SELECTION PROCESS

Inclusion criteria
1. eHealth application
2a. in Title: outcome-measure and/or evaluation and/or risk
2b. in Abstract: risk and/or limitation found
3. Quality of care and/or patients´ safety/well being
4. Design: Randomized controlled trial
5. Publication year: between 2000 – 2011
6. Language: German or English

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Identified risks were structured according to a multi-level approach covering risks dealing with either human factors, technological factors or organizational factors, referring to the framework for health information systems evaluation as proposed by Yusof et al. [21].

III. WEB-BASED SOURCES

To broaden our view we have included 'grey literature'. The 'Prague Definition'[1] of grey literature states that "Grey literature stands for manifold document types produced on all levels of government, academics, business, and industry in print and electronic formats that are protected by intellectual property rights, of sufficient quality to be collected and preserved by library holdings or institutional repositories, but not controlled by commercial publishers, i.e., where publishing is not the primary activity of the producing body." This material cannot be found and disclosed easily through the usual channels. It may include government research and non-profit reports, dissertations and expert assessments, conference proceedings and technical reports, institutional repositories, investigations, and other primary resource materials such as records, archives, observations, data, filed notes and 'new' sources such as pre-prints, web logs, online preliminary research results, open data, unpublished theses, project web sites, standards and specifications collections, online data archives or other types of documentation.

Given the plethora of different types of organizations publishing information on eHealth, we decided to start with explorative searches in sources of different status without using a systematic selection procedure. Firstly, we have visited a series of websites of international and national health organizations/government agencies to see if they mention risks associated with eHealth technology in any way. Secondly, we have searched databases, respectively of the U.S. Food and Drug Administration and the ECRI Institute. Thirdly, we have accessed websites of three expert centers on medical technology: the ECRI Institute, Prismant (Dutch) and ZonMw (id.). Finally, a major Dutch professional journal on health care matters was queried on risk factors concerning eHealth and telemedicine (see Appendix V). On each website we searched for information on the risks involved with eHealth and telemedicine. The search terms used were ehealth, telemedicine and tele*. Results involving the monitoring, programming or diagnosis of pacemakers and other implantable cardiologic devices were excluded because they are considered to represent ancillary functions to those devices, rather than eHealth applications in their own respect.

[1] 12th International Conference on Grey Literature (Prague, Dec. 2010); http://www.opengrey.eu/item/display/10068/700015 [accessed Jan 15, 2013]

IV. FOCUS GROUP

To test the findings from literature against the opinions of stakeholders we organized an `invited expert meeting'. We selected experts from industry, health care, government, patient organizations, insurers and universities from our networks and requested them to participate. In advance, they received a working draft version of the research report. A focus group (n=38) could be composed representing the respective stakeholders. Its main goal was to identify important sources of data that were not yet included at that time, and to further discuss and develop the preliminary conclusions and recommendations from the literature scan.

A professional talk-host led the meeting that opened with an introduction and a summary of the study outcomes by the authors. This was followed by a one-hour 'knowledge café' method, an informal but systematic way to exchange and map opinions and ideas of participants. After a break and a philosophical reflection on technologies and risk, a discussion panel took place wherein representatives of stakeholders actively participated. Outcomes were noted down, analyzed and summarized.

V. OUTCOMES

A. Literature scan

The search was performed in SciVerse Scopus in July 2011 delivering initially 340 potentially relevant publications. Of these, 17 were eventually included after the selection procedure described sub II.

Human, technological or organizational risks appear to be no primary subject of the randomized clinical trials identified in the search. However, they are reported as secondary effects or unintended outcomes of eHealth technology effectiveness studies. In most cases, the observed risks are related to a lack of effectiveness in all or part of the target groups due to either the design of the intervention, implementation factors or intrinsic characteristics of the target groups. Other types of unintended adverse effects leading to harm for patients, users or third persons were rarely mentioned.

Identified risks have been structured with regard to their primary occurrence at a human level, a technological level and an organizational level (Table III). Appendix III contains a detailed overview of risks, the level where they occur, their classification and their source in eHealth literature.

1) Risks concerning Human factors

Masa et al. [22] compared conventional spirometry to online spirometry with regard to outcome measures like forced vital capacity, quality criteria (acceptability, repeatability) and the number of maneuvers and time spent on both of the two procedures. They found that the number of spirometric maneuvers needed to meet quality criteria was somewhat higher in the online mode as compared to

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

conventional spirometry. Online spirometry also took more time for patients (mean differences of 0.5 additional maneuvers and 0.7 minutes more). Higher time-consumption may also negatively affect the remote technician instructing the patient while the latter uses the spirometer. The spirometric values achieved online were very similar to the values achieved by conventional spirometry.

Some eHealth applications appear to be more beneficial for specific patient groups. Bujnowska-Fedak et al. [23] tested a tele-homecare application for monitoring diabetes. Older and higher educated patients, spending a lot of the time at home and having acquired diabetes recently, benefited most from the application. A positive association was found between educational level and ability to use the tele-monitoring system without assistance. Spijkerman et al. [24] evaluated a web-based alcohol-intervention without (group 1) and with (group 2) feedback compared to a control group in order to reduce drinking behavior in 15-20 years old Dutch binge-drinkers. They found that the intervention may be effective in reducing weekly alcohol use and may also encourage moderate drinking behavior in male participants over a period of 1-3 months. The intervention seemed mainly effective in males while for females a small adverse effect was found. Women following intervention group 1 were less likely to engage in moderate drinking and had increased weekly drinking a little, although significantly (p=0.06; 1.6 more drinks/week), at one month follow-up. Zimmerman et al. [25] performed a secondary analysis on data from an RCT on a symptom-management intervention for elderly patients during recovery after coronary artery bypass surgery. They found that the intervention had more impact on women than on men for symptoms such as fatigue, depression, sleeping problems and pain. Regarding measures of physical functioning no gender differences were found. Cruz-Correira et al. [26] tested adherence to a web-based asthma self-management tool in comparison to a paper-based diary. The tool was designed to collect and store patient data and provide feedback to both patient and doctor about the former's condition in order to support medical decision making. Patients' adherence to the web-based application was lower than in the control group. Willems et al. [27] tested a home monitor self-management program for patients with asthma where data such as spirometry results, medication use or symptoms were recorded. They found a low compliance of participants with the intervention protocol. Participants in the intervention group recorded in average less PEF tests (peak expiratory flow; lung function data): 1.5 per day versus the required number in the protocol of 2 tests per day. Verheijden et al. [28] tested a web-based tool for nutrition counseling and social support for patients with increased cardiovascular risk in comparison to a control group receiving conventional care. The authors found that the uptake of the application in the intervention group was low (33%) with most participants using the tool only once during the 8 months study period. Patients properly using the intervention were significantly younger than those who did not. Morland et al. [29] compared an anger management group therapy for veterans delivered face-to-face versus via videoconferencing. Group therapy via videoconferencing teleconferencing seemed effective to treat anger symptoms in veterans. While no differences could be found between the two groups regarding attendance or homework completion, the control group reported a significant higher overall group therapeutic alliance than the intervention group. Postel et al. [30] evaluated an eTherapy program for problem drinkers, where therapist and patient communicated online to reach a reduction of alcohol use, as compared to a control group receiving regular information by email. While effective for complying participants, they found high drop-out rates in the eTherapy group though quitting the program did not automatically mean that the participant had also relapsed or increased alcohol consumption. Ruffin et al. [31] tested a web-based application where participants received tailored health messages after giving information about family history of six common diseases. In the intervention group the authors found modest improvements in self-reported physical activity and fruit and vegetable intake. But participants also showed a decreased cholesterol-screening intention as compared to the control group who received standard health messaging.

In summary, higher time consumption, unintended adverse effects, and selective benefits differing for sex, education, age and other variables are the risks observed on the side of the human (end-)user. Frequently adherence (or: compliance, drop-out, alliance, up-take) is mentioned and associated with a negative impact on the desired effect of an intervention.

2) Risks concerning Technology

Evaluating a tele-homecare application for monitoring diabetes Bujnowska-Fedak et al. [23] observe usability problems among participants; 41% of them (patients with type 2 diabetes) were unable to use the system for glucose-monitoring needing permanent assistance. Patients who could easily use the application derived a greater impact from its use. Nguyen et al. [32] evaluated an internet-based self-management program for COPD patients but discontinued before the sample target was reached due to technical and usability problems with the application. Participants stated at the exit interview that decreased accessibility, slow loading of the application, and security concerns prevented them from using the website more frequently. Participants reporting usability problems had to complete (too) many actions on a PDA-device before being able to submit an exercise or symptom entry. Other problems dealt with limited wireless coverage of the PDA. The technical problems decreased participants´ engagement

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

with the tools. Decreased engagement was associated with the number of web log-ins and the exercise and symptom entered via the website and/or the PDA. While evaluating a web-based asthma self-management tool Cruz-Correira et al. [26] found nine patients reporting problems (19 in total) related to the use of a web-based self-management tool. Most problems concerned the internet connection and the graphical user interface. Two of the patients could not even use the application because of technical problems. Demaerschalk et al. [33] tested the efficacy of a telemedicine application (vs. telephone-only consultation) for the quality of decision making regarding acute stroke. They found technical issues in 74% of telemedicine consultations versus none in telephone consultations. The observed technical problems did not prevent the determination of treatment decision but some did influence the time necessary to treatment decision-making. Jansà et al. [34] used a telecare-application for type 1 diabetes patients having poor metabolic control to send glycaemia values to the diabetes team. They found that 30% of team-patient appointments were longer than expected (1h vs. 0.5h) due to technical problems with the application. Technical problems concerned the inability to send results of counseling caused by problems with the application itself, the server or internet-access. Using a telemanagement application for diabetes patients Biermann et al. [35] found that 15% of the participants had difficulties in handling the application, the consequences of which were not elaborated. In a study of an asthma self-management telemonitoring program by Willems et al. [27] 1/3 of participants experienced technical problems, mostly with malfunctioning devices. Practitioners had to contact patients, e.g., regarding a missed data transfer leading to logistical problems.

In summary, a variety of issues has been reported at the technology level affecting patient safety or quality of care. They range from usability problems and security issues to problem with accessing the server or malfunctioning devices.

3) Risks concerning Organization

Copeland et al. [36] tested whether a telemedicine self-management intervention for congestive heart failure (CHF) patients could be effective in terms of improving physical and mental health-related quality of life and cost-effectiveness as compared to a control group receiving usual care. They could not find substantial differences between groups, but overall costs related to CHF were higher for the intervention group. The authors state that this might be related to the intervention encouraging medical service utilization by facilitating access to care.

One tele-management application for diabetics allows patients to measure their blood-glucose values and send it to their care provider [35]. Though time-saving for patients, use of the application lead to 20% more time investment (50 vs. 43 min. per month over a 4-month period, and 43 vs. 34 min. per month over an 8-month period) on the side of the care provider compared to conventional care. The higher time expenditure did not reflect time necessary to manage the application itself: it was due to more access to the provider, so that patients tended to call more often. Montori et al. [37] also found a comparable risk concerning time-consumption. They tested a telecare-application for data-transmission for type 1 diabetes patients. The nurses needed more time reviewing glucometer data (76 min. vs. 12 min.) and giving the patient feedback (68 minutes vs. 18 minutes) in the telecare condition as compared to the control group. The authors found more nurse feedback time to be significantly associated with more changes in insulin doses; more changes of doses thus appeared in the telecare group.

Strayer et al. [38] tested a personal digital assistant (PDA) as a tool for improving Smoking Cessation Counseling (SCC) against a paper-based reminder tool. In semi-structured interviews, medical students providing SCC reported that they felt barriers for using the PDA in practice such as a lack of time or a lack of training. In addition, they felt uncomfortable to use the PDA in the presence of patients. The PDA tool did not increase key SCC behaviors of the participants of the intervention group as compared with the paper-based reminder.

In summary, increased time consumption, barriers for proper use and financial issues are the risks observed at the organizational level.

TABLE III. OBSERVED RISKS

Risk level	Description
Human level	Adherence (or compliance, drop-out, alliance, up-take)
	Unintended adverse effects
	Selective patient benefits (sex, education, age and other variables)
Technology level	Usability problems
	Access
	Security issues
	Malfunctioning devices
Organizational level	Higher time consumption
	Barriers for proper use
	Higher costs

In Table III, the identified risks have been summarized with regard to the various levels of their occurrence.

B. Web-based sources

From the mixed web-based sources it appears that the information on eHealth and telemedicine is overly positive. The risks, downsides or failures that are inevitably part of any project, are rarely mentioned - neither prominently nor implicitly. Nevertheless, a number of sources mention the imperative provisions that should be made to ensure that eHealth or telemedicine projects will be successful. It could be assumed that these are indicative of the risks they are often related to. These are grouped into three categories: the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

human factor, technology and organization, summarized in Table IV.

TABLE IV. SUMMARY OF OBSERVED RISKS IN WEB-BASED SOURCES

RISK LEVEL	DESCRIPTION
Human level	Lack of physical, mental, social, cognitive skills (eHealth literacy)
	Substitution human contact, doctor-patient relationship
Technology level	Problems with resolution, interference, bandwidth, connections
	Incompatibility, sub optimal interoperability
	User-unfriendly technology
	Insufficient error handling, no emergency plans
Organizational level	Money, lack of training/instruction, data-management, hardware
	Home (unclear liability, accountability, insurance issues)
	Uncertain response speed of care organizations 24/7

1) The human factor

eHealth and telemedicine are not intended to substitute patient-physician contact. Use of technology may reduce the number of contacts, thus increasing the efficiency of health care. For patients it may be beneficial that the number of visits to the physician can be reduced, thus saving time and expenses. Nevertheless, periodic direct in-person contact should not be completely replaced. Any project should primarily be driven by needs and not by technology. Before a project starts, a needs-analysis should be performed and the added value should be proven. Scientific evidence of effectiveness in a large scale settings seems to be missing in many cases. Safe application of eHealth and telemedicine requires that patients are capable of self-management and are physically and mentally able to handle the technology and the tasks that come with an intervention. The patient should be motivated to use the technology correctly, follow instructions and procedures, be well-trained and function without cognitive or communication difficulties. The patient should be confident to use the technology, but at the same time not completely rely on it.

2) Technology

The early initiatives of eHealth and telemedicine suffered from technological shortcomings such as the limited resolution or the narrow band width for transmitting data. These limitations are largely overcome, but others appear. With more and more wireless applications that transmit digital signals, problems arise like interference and frequency overlap. Where eHealth or telemedicine depend on a continuous online connection, the risk of a failing connections should be taken into account. Equipment should be designed to match the skills of the user, ergo shall be self-explanatory, as simple as possible to operate and be 'layman proof'. The databases from the FDA and ECRI clearly show that medical technology is known to fail and may subsequently cause harm to the patient. Where there is a physical distance between the patient and the care provider it may occur that a device is not working properly, while this is not noticed by the patient or the care provider. Mechanisms should be implemented to detect and identify errors in transmission, equipment failure and software bugs. An emergency plan for alternative treatment or monitoring should be in place. Where medical devices and equipment from different manufacturers are used together or are connected to generate, store or process data, these shall be interoperable. The same applies for electronic patient records and health files, and where possible internationally.

3) Organization (incl. legal and financial issues)

All stakeholders should be identified and there shall be a common understanding of tasks and responsibilities of the stakeholders. Training of the users of the technology should be well organized and should include actions that need to be taken in case of emergencies, e.g., patient distress, or failing equipment. If the technology sends messages to the health care provider these should be followed up without delay. The health care organization should consider hiring dedicated personnel to handle the technical side of eHealth or telemedicine services, so that the physicians can focus on the medical aspects. Depending on the type of eHealth service or telemedicine it may be necessary to have a 24/7 care response service available. The staff that provides the response service should be adequately trained. The supply and management of equipment, including maintenance, response to malfunction and training of the patient shall be organized. To sum up, the management of the technology must be well embedded in the organization of the health care provider and not be an isolated entity. Legal issues include licenses and credentials (especially when patient and physician do not reside in the same country), liability, data confidentiality, data storage and patient privacy. eHealth and telemedicine projects may benefit from local electronic patient files and a national (or even international) health file. The tasks and responsibilities of all the parties involved in the implementation and use of the technology must be documented. Financial issues appear to be an important 'show stopper'. eHealth and telemedicine need to mature into accepted forms of health care that can operate without special funding. To convince policy makers and financers, every eHealth or telemedicine project needs to be evaluated to demonstrate the added value and that the project goals are met.

C. Authoritative reports published during data analysis

Near the end of our data analysis process, three reports were published that we considered particularly relevant to our own study. The first is the report 'National

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Implementation Agenda eHealth' [39], a joint policy paper (Dec. 2011) of the Royal Dutch Medical Association (KNMG), the Dutch Federation of Patients and Consumer Organizations (NPCF) and the Dutch Health Care Insurers Association (Zorgverzekeraars Nederland). The second is the report 'Health IT and Patient Safety: Building Safer Systems for Better Care' (Nov. 2011) published by the U.S. Institute of Medicine [40]. The third is 'State of Health Care 2011. In health care, patient information exchange challenges is not resolved with ICT without the standardization of processes' (Oct. 2011) a report by the Dutch Healthcare Inspectorate [41]. These authoritative reports exemplify that eHealth technology will substantially change the health care system in the coming decade. They confirm that inconclusive evidence exists when it comes to risks for patient safety and quality of care. If risks are to be contained at an acceptable level, some serious hurdles have to be taken.

The policy paper of three main stakeholders in Dutch health care, which was also sent to the Parliament by the Ministry of Health, demonstrates the present political dynamics necessary to bring about such a change. However, the scientific back-up for their claims is not as strong as their political determination. For instance the statement that eHealth "contributes to affordable, accessible, high-quality health care and more direction for patients" is not supported by prevailing evidence as of yet. The National Implementation Agenda also neglects the considerable risks as outlined by the Institute of Medicine (IOM) and the Dutch Healthcare Inspectorate (IGZ). At the same time, it is true that reports are available of successful practices and promising outcomes in the whole range of health care services. These developments render a certain urgency to the issue of risk control and prevention, which until recently did not receive much attention.

IOM advances safety as an essential value in health care and favors an holistic approach to improve overall safety of the health care system. Transparency, education and collaboration of all stakeholders are the main components of the approach. IGZ emphasizes the importance of safe and secure information exchange as a vital to risk reduction. Both organizations provide a series of recommendations to improve patient safety.

D. Focus group

The preliminary conclusions of the draft report were generally accepted and supported by the experts. From their respective angles they advanced valuable additional subjects related to the present paper. We inferred the following cross-cutting themes from the discussion, that are vital for risk control in eHealth:

- Patient-centeredness;
- Interoperability and standardization;
- Risk management tools and regulations;
- Integrative approach of risk-management in eHealth;
- eHealth affects organization of care;
- Transparency in risk documentation;
- Education.

The integration of these themes in the implementation of eHealth is expected to considerably reduce the incidence of risks in healthcare.

VI. CONCLUSION AND FUTURE WORK

Randomized clinical trials and studies of the immediate risk of eHealth technology for patients' safety or quality of care have not been found. In the margin of studies aiming to evaluate the effectiveness of eHealth interventions risks are reported as unintended, secondary outcomes. The selected studies suggest evidence for risks at all three levels of the multi-level approach applied. Ten studies mention risks concerning the patient at the human level, especially where adherence issues lead to suboptimal use of an intervention and corresponding low effectiveness. But also adverse effects were reported, as well as the fact that not all patient groups can equally benefit from an eHealth intervention. Issues at a technological level were found in seven studies, revealing considerable rates of usability problems, limited access or other technical problems. Organizational issues were found with regard to higher use of resources (time, money, staff) affecting quality of care in two studies. Table III shows the level and nature of the risks observed in our study. In some cases the causes of the risks were qualified as study (design) artifacts. In many instances the consequences have not been elaborated.

In the web-based sources we studied, a positive attitude towards eHealth prevails and risks or failures are rarely mentioned. A number of sources mention conditions for eHealth projects to succeed (Table IV). These may be used as input in risk analysis and should be reinforced through risk management and continuous surveillance.

The focus group outcomes demonstrate the significance of stakeholder involvement at all levels. Our findings from literature and web-based resources are reflected in the resulting themes. We conclude that while not much is known about the magnitude of risks associated with eHealth, a lot of non-systematic, anecdotal material indicates that risks happen at the level of human functioning, technology and organization.

We intend to further contribute to risk awareness in eHealth and conduct follow-up research in this field.

VII. DISCUSSION

Increasing use of eHealth technology is one of the major developments in today's healthcare [42]. The opportunities of web-based and mobile eHealth technologies should

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

therefore remain central to the global health discourse. At the same time, however, it is required to explore the risk potential of these technological advancements.

Risk is a complicated issue that refers to a lack of knowledge along subjective and objective dimensions. The observed lack of academic interest for risk assessment in eHealth technology should be a matter of concern. Patient safety and quality of care deserve a higher level of risk awareness when it comes to new technologies. At present risks emerge in the margin of trials and interventions in eHealth. They are conceived as problems, issues, disadvantages, costs or other designations that one way or another affect human, technological or organizational functioning in a detrimental manner.

Though both quantity and quality of the reported issues do not seem to be disturbing at first glance, a wider search would almost certainly deliver a disquieting range and diversity of risks. Given the outcome of our study that none of the systematic studies were designed to study risks, we must conclude that they do in fact not represent the studies with the highest evidence level related to our research question. Therefore, a follow-up search, including review articles, controlled clinical trials, and perhaps observational studies should be performed.

Furthermore, in databases such as MAUDE (Manufacturer and User Facility Device) of the U.S. Food and Drug Administration, in grey literature, articles in professional magazines and other (online) sources of different organizational, consumer and academic nature a variety of incidents involving risks have been recorded. These are often viewed as avoidable or improvable intervention flaws, or explained as study (design) artifacts, but they should not be played down. Their presumed occurrence give rise to careful reconsideration when it comes to exploring the opportunities of web-based and mobile eHealth technologies for global healthcare innovation. eHealth is not an exotic domain in health care and should be treated as a such. The indications for risks found in the present study should play a role in keeping the health care community alert with regard to risk management. The participants of the focus group would certainly acknowledge this.

This implies the need for extensive research that explicitly focuses on establishing the volume and nature of such risks in order to prevent or minimize them. It also implies an improved way of monitoring to advance transparency in the reporting of risk occurrence and safety incidents. Finally, it implies a higher level of health care risk management, continuity of care and understanding of how risks affect patients through risk identification, operating ways to avoid or moderate risks and developing contingency plans when risks cannot be prevented or avoided. Available tools and standards should preferably be used to achieve this.

The results of the present scan are in accordance with outcomes from the ceHRes study that covers over a decade of eHealth technological development [17]. The 'conceptual' risks (Table I) represent the same categories of risks that result from the literature scan. For instance expert-driven eHealth interventions that neglect the essential role of patients may lead to adherence issues as mentioned sub V-A.1). Or disregarding conditions for implementation may imply the underestimation of issues such as high time-consumption, mentioned sub V-A.3). To minimize and avoid such risks a 'Roadmap' has been developed to design, develop, implement and evaluate eHealth interventions (see Appendix IV). It applies concepts and techniques from business modeling and human centered design [43]. The roadmap serves as a guideline to collaboratively improve the impact and uptake of eHealth technologies. For this purpose it has been published as a wiki (ehealthresearchcenter.org/wiki/).

For now the ubiquitous trust in technology seems to be unjustified and it needs to be put in perspective. We have the instruments, in particular risk management approaches, and the knowledge to reconsider the implementation of eHealth to achieve this, so eHealth can become part of evidence based medicine.

VIII. LIMITATIONS

The inclusion criteria of the study, such as the requirement for RCTs in the review of scientific literature, were found to be limiting, since we are looking to novel technologies in tele-/eHealth. Moreover, RCTs in eHealth environments tend to mitigate the impact and uptake of interventions because of costs, timelines and limitations.

We have probably missed a number of British publications and websites because of the choice of the term 'eHealth', which appears to be not widely used in the United Kingdom, and generally is assumed to refer to electronic patient records and transmission of acute health information electronically. Furthermore, we may have missed important websites such as NHS networks (see: http://www.networks.nhs.uk/ because of the federal nature of the NHS as well as more regional online outlets.
Exploring the full spectrum of 'grey literature' would have delivered much more indications on the occurrence of risks though we expect it would not have helped in quantifying their magnitude.

ACKNOWLEDGMENTS

The Dutch Health Care Inspectorate commissioned the National Institute for Public Health and the Environment (RIVM) to conduct this study of which we here present the outcomes. It was carried out in collaboration between the Centre for Health Protection and the Health portal kiesBeter.nl (RIVM), and the Center for eHealth Research

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

and Disease management (IGS - Institute for Governance and Innovation Studies, University of Twente).

The full report is disseminated by RIVM [44]. We thank ms. Fabiola Mueller for her work in data collection. We also thank the participants of the expert meeting d.d. 25th of November 2011, Utrecht, Netherlands.

We are indebted to the members of the Special Interest Group Telemedicine of the EC New and Emerging Technologies Working Group for their useful comments to the draft version of the report on which the present paper is based.

Parts of this paper have been presented as an original research paper at eTELEMED, the 4th International Conference on eHealth, Telemedicine and Social Medicine [1].

REFERENCES

[1] H.C. Ossebaard, R.E. Geertsma and J.E.W.C. van Gemert-Pijnen, "Health Technology Trust: Undeserved or Justified?," in: Proceedings 4th International Conference on eHealth, Telemedicine, and Social Medicine eTELEMED 2012, Jan 30-February 4, 2012, Valencia, Spain, J.E.W.C. van Gemert-Pijnen, H.C. Ossebaard, A. Smedberg, S. Wynchank and P. Giacomelli, Eds. Red Hook: Curran Associates Inc., 2012, pp 134-142.

[2] WHO, "The World Health Report 2003 – Shaping the future," Geneva: World Health Organization, 2003.

[3] WHO, "Global Status Report on Noncommunicable Diseases 2010," Geneva: World Health Organization, 2010.

[4] S.M. Kelders, J.E.W.C. van Gemert-Pijnen, A. Werkman, N. Nijland, and E.R. Seydel. "Effectiveness of a Web-based intervention aimed at healthy dietary and physical activity behavior: a randomized controlled trial about users and usage," J Med Internet Res. 2011 Apr 14;13(2):e32.

[5] F. Verhoeven, K. Tanja-Dijkstra, N. Nijland, G. Eysenbach, and J.E.W.C. van Gemert-Pijnen, "Asynchronous and Synchronous Teleconsultation for Diabetes Care: A Systematic Literature Review," J Diabetes Sci Technol. 2010 May; 4(3): pp. 666–684.

[6] Resolution WHA58-28, "eHealth," in: "Fifty-eighth World Health Assembly, Geneva, 16-25 May 2005. Resolutions and decisions," Geneva: World Health Organization, 2005. http://apps.who.int/gb/ebwha/pdf_files/WHA58/WHA58_28-en.pdf [accessed 9 May 2011].

[7] R.R. Glasgow, "eHealth Evaluation and Dissemination," Research American Journal of Preventive Medicine 32(5), 2007, pp. 119-126.

[8] A.D. Black, J. Car, C. Pagliari, C. Anandan, K. Cresswell, T. Bokun, B. McKinstry, R. Procter, A. Majeed and A. Sheikh, "The impact of eHealth on the quality and safety of health care: a systematic overview," PLoS Med (8) 2011, e1000387-doi: 10.1371/journal.pmed.1000387

[9] A.A. Atienza, B.W. Hesse, T.B. Baker, D.B. Abrams, B.K. Rimer and R.T. Croyle, "Critical issues in eHealth research," Am J Prev Med 2007, 32(5), S71-S74.

[10] R.E. Geertsma, A.C.P. de Bruijn, E.S.M. Hilbers, M.L. Hollestelle, G. Bakker and B. Roszek. "New and Emerging Medical Technologies - A horizon scan of opportunities and risks," RIVM report 360020002, 2007. Bilthoven: RIVM.

[11] IGZ (Dutch Healthcare Inspectorate),"State of Health Care. 2008 Medical technological risks underestimated" [Staat van de gezondheidszorg 2008. Risico's van medische technologie onderschat]. The Hague: IGZ, 2008.

[12] National Academy of Sciences, "Health IT and Patient Safety: Building Safer Systems for Better Care," Washington: Institute of Medicine, 2011.

[13] D. Barben, "Analyzing acceptance politics: Towards an epistemological shift in the public understanding of science and technology," Public Understanding of Science 19 (3) May 2010, pp. 274-292.

[14] M. Dierkes and C. von Grote (Eds.), "Between understanding and trust: the public, science and technology," Harwood Academic, 2000.

[15] WRR - Dutch Scientific Council for Government Policy, " iGovernment" [iOverheid]. Amsterdam: Amsterdam University Press, 2011.

[16] M. Beeuwkes Buntin, M.F. Burke, M.C. Hoaglin and D. Blumenthal, "The Benefits of Health Information Technology," Health Affairs (30)3, 2011, pp. 464-471.

[17] J.E.W.C. van Gemert-Pijnen, N. Nijland, H.C. Ossebaard, A.H.M van Limburg, S.M. Kelders, G. Eysenbach and E.R. Seydel, "A holistic framework to improve the uptake and impact of eHealth technologies," J Med Internet Research 13(4), 2011, e111 doi:10.2196/jmir.1672.

[18] ISO/IEC, "Guide 51:1999. Safety aspects - Guidelines for the inclusion in standards." Geneva: ISO, 1999.

[19] EN ISO 14971:2009, "Medical devices - Application of risk management to medical devices (ISO 14971:2007, Corrected version 2007-10-01)." Brussels: CEN/CENELEC, 2009.

[20] IGZ (Dutch Healthcare Inspectorate), "State of Health Care. 2011. In health care, patient information exchange challenges not resolved with ICT without standardization of processes" [Staat van de gezondheidszorg 2011. Informatie-uitwisseling in de zorg: ICT lost knelpunten zonder standaardisatie van de informatie-uitwisseling niet op]. Utrecht: IGZ, 2011

[21] M.M. Yusof, J. Kuljis, A. Papazafeiropoulou and L.K. Stergioulas, "An evaluation framework for health information systems: human, organization and technology-fit factors (HOT-fit)," Int. J. Med. Inform. 77(6), 2008, pp. 386-398. PMID:17964851

[22] J. F. Masa, M. T. González, R. Pereira, M. Mota, J.A. Riesco, J. Corral and R. Farré, "Validity of spirometry performed online," European Respiratory Journal, 37(4), 2011, pp. 911-918. doi: 10.1183/09031936.00011510

[23] M. M. Bujnowska-Fedak, E. Puchała and A. Steciwko, "The impact of telehome care on health status and quality of life among patients with diabetes in a primary care setting in Poland," Telemedicine and e-Health, 17(3), 2011, pp. 153-163. doi: 10.1089/tmj.2010.0113

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[24] R. Spijkerman, M. A. E. Roek, A. Vermulst, L. Lemmers, A. Huiberts and R. C. M. E. Engels, "Effectiveness of a Web-based brief alcohol intervention and added value of normative feedback in reducing underage drinking: A randomized controlled trial, " Journal of Medical Internet Research, 12(5), 2010, e65p.61-e65p.14. doi: 10.2196/jmir.1465

[25] L. Zimmerman, S. Barnason, M. Hertzog, L. Young, J. Nieveen, P. Schulz and C. Tu, "Gender differences in recovery outcomes after an early recovery symptom management intervention," Heart and Lung: Journal of Acute and Critical Care," 40(5), 2011, pp. 429-39. doi: 10.1016/j.hrtlng.2010.07.018.

[26] R. Cruz-Correia, J. Fonseca, L. Lima, L. Araújo, L. Delgado, M.G. Castel-Branco and A. Costa-Pereira, "Web-based or paper-based self-management tools for asthma--patients' opinions and quality of data in a randomized crossover study," Studies in health technology and informatics, 127, 2007, pp. 178-189.

[27] D.C.M. Willems, M.A. Joore, J.J.E. Hendriks, R.A.H. van Duurling, E.F.M. Wouters and J. L. Severens, "Process evaluation of a nurse-led telemonitoring programme for patients with asthma," Journal of Telemedicine and Telecare, 13(6), 2007, pp. 310-317, doi: 10.1258/135763307781644898

[28] M. Verheijden, J.C. Bakx, R. Akkermans, H. van den Hoogen, N.M. Godwin, W. Rosser, W. van Staveren and C. van Weel, "Web-Based Targeted Nutrition Counselling and Social Support for Patients at Increased Cardiovascular Risk in General Practice: Randomized Controlled Trial'" J Med Internet Res 6(4), 2004, e446(4).

[29] L. A. Morland, C.J. Greene, C.S. Rosen, D. Foy, P. Reilly, J. Shore and B.C. Frueh, "Telemedicine for anger management therapy in a rural population of combat veterans with posttraumatic stress disorder: A randomized noninferiority trial," Journal of Clinical Psychiatry, 71(7), 2010, pp. 855-863. doi: 10.4088/JCP.09m05604blu.

[30] M. G. Postel, H. A. de Haan, E.D. ter Huurne, E.S. Becker, E. S. and C. A. J. de Jong, "Effectiveness of a web-based intervention for problem drinkers and reasons for dropout: Randomized controlled trial,"Journal of Medical Internet Research," 12(4), 2010 e68p.61-e68p.12. doi: 10.2196/jmir.1642

[31] M.T. Ruffin, D.E. Nease, A. Sen, W.D. Pace, C. Wang, L.S. Acheson and R. Gramling,"Effect of preventive messages tailored to family history on health behaviors: The family healthware impact trial," Annals of Family Medicine, 9(1), 2011, pp. 3-11. doi: 10.1370/afm.1197.

[32] H.Q. Nguyen, D. Donesky-Cuenco,S. Wolpin,L.F. Reinke, J.O. Benditt, S.M. Paul and V. Carrieri-Kohlman,"Randomized controlled trial of an internet-based versus face-to-face dyspnea self-management program for patients with chronic obstructive pulmonary disease: Pilot study," Journal of Medical Internet Research, 10(2), 2008, doi: 10.2196/jmir.990

[33] B. M. Demaerschalk, B.J. Bobrow, R. Raman, T.E.J. Kiernan, M.I. Aguilar, T.J. Ingall and B.C. Meyer, "Stroke team remote evaluation using a digital observation camera in arizona: The initial mayo clinic experience trial, " Stroke, 41(6), 2010, pp. 1251-1258. doi: 10.1161/strokeaha.109.574509

[34] M. Jansà, M. Vidal, J. Viaplana, I. Levy, I. Conget, R. Gomis and E. Esmatjes, "Telecare in a structured therapeutic education programme addressed to patients with type 1 diabetes and poor metabolic control," Diabetes Research and Clinical Practice, 74(1), 2006, pp. 26-32. doi: 10.1016/j.diabres.2006.03.005

[35] E. Biermann, W. Dietrich, J. Rihl and E. Standl, "Are there time and cost savings by using telemanagement for patients on intensified insulin therapy?: A randomised, controlled trial," Computer Methods and Programs in Biomedicine, 69(2), 2002, pp. 137-146. doi: 10.1016/s0169-2607(02)00037-8

[36] L. A. Copeland, G. D. Berg, D. M. Johnson and R.L. Bauer, " An intervention for VA patients with congestive heart failure" American Journal of Managed Care, 16(3), 2010, pp. 158-165.

[37] V. M. Montori, P. K. Helgemoe, G. H. Guyatt, D. S. Dean, T. W. Leung, S. A. Smith and Y.C. Kudva, "Telecare for Patients with Type 1 Diabetes and Inadequate Glycemic Control: A randomized controlled trial and meta-analysis. Diabetes Care," 27(5), 2004, pp. 1088-1094. doi: 10.2337/diacare.27.5.1088

[38] S. M. Strayer, S.L. Pelletier, J.R. Martindale, S. Rais, J. Powell and J.B. Schorling, "A PDA-based counseling tool for improving medical student smoking cessation counseling. Family Medicine," 42(5), 2010, pp. 350-357.

[39] National Implementation Agenda eHealth, "A joint policy paper of the Royal Dutch Medical Association (KNMG), the Federation of Patients and Consumer Organisations (NPCF) and the Health care insurers Association (Zorgverzekeraars Nederland)," December 2011 http://www.rijksoverheid.nl/documenten-en-publicaties/rapporten/2012/06/07/nationale-implementatieagenda-e-health-nia.html [accessed Oct. 12, 2012].

[40] National Academy of Sciences, " Health IT and Patient Safety: Building Safer Systems for Better Care," Washington: Institute of Medicine, 2011.

[41] IGZ (Dutch Healthcare Inspectorate), "State of Health Care 2011. In health care, patient information exchange challenges not resolved with ICT without standardization of processes," Utrecht: IGZ, 2011.

[42] D.C. Duchatteau and M.D.H. Vink, "Medical-technological developments care. Background study" [Medisch technologische ontwikkelingen zorg 20/20. Achtergrondstudie], The Hague: Council for Public Health and Health Care [Raad voor de Volksgezondheid en Zorg], 2011.

[43] A.H.M. van Limburg, J.E.W.C. van Gemert-Pijnen, N. Nijland, H.C. Ossebaard, R.M.G. Hendrix and E.R. Seydel, " Why business modelling is crucial in the development of eHealth technologies," J Med Internet Res 13(4) 201, e124 doi:10.2196/jmir.1674

[44] H.C. Ossebaard, A.C.P. de Bruijn, J.E.W.C. van Gemert-Pijnen, R.E. Geertsma, "Risks related to the use of eHealth technologies - an exploratory study, RIVM Report 360127001," Bilthoven: RIVM, 2013. See: http://www.rivm.nl/bibliotheek/rapporten/360127001.pdf

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Appendix I

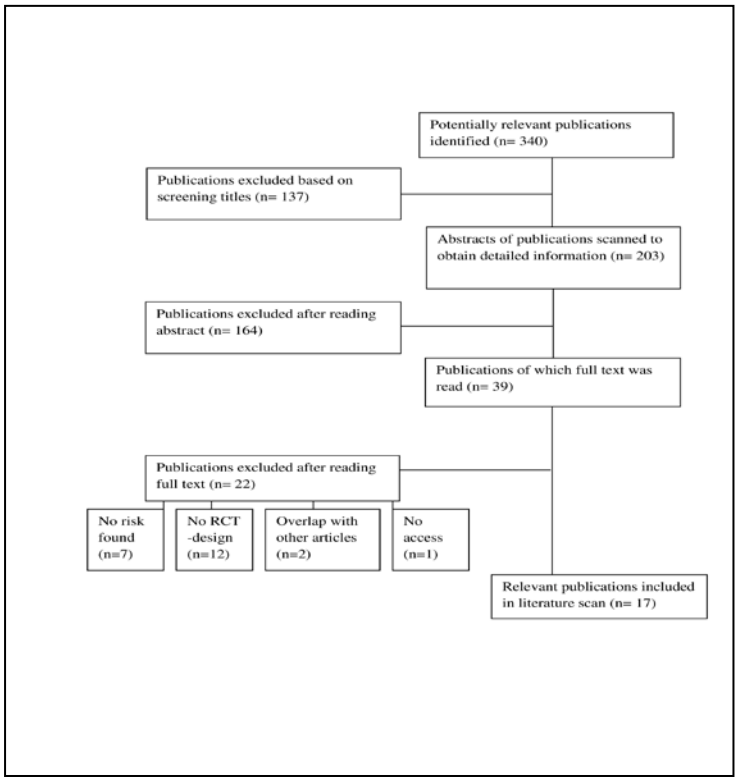
Search query used in SciVerse Scopus

(TITLE-ABS-KEY(ehealth OR e-health OR "e health" OR etherapy OR e-therapy OR "e therapy" OR emental OR e-mental OR "e mental" OR telemedicine OR telecare OR teleconsult OR telemonitoring OR telehealth OR teleconference OR "health information technology" OR "web based") OR TITLE-ABS-KEY("internet based" OR "web application" OR domotica OR “personal digital assistant” OR “pda”) AND TITLE-ABS-KEY(risk OR risks OR danger* OR threat OR threats OR limitation* OR barrier* OR problem* OR concern* OR challenge OR challenges OR “adverse effect*” OR quality OR drawback OR drawbacks) AND TITLE-ABS-KEY(health OR care OR “healthcare” OR healthcare) AND TITLE-ABS-KEY("randomized clinical trial*" OR "randomised clinical trial*" OR "randomized controlled trial*" OR "randomised controlled trial*" OR rct OR "RCTs" OR experimental)) AND PUBYEAR AFT 1999 AND PUBYEAR BEF 2012 AND (LIMIT-TO(LANGUAGE, "English") OR LIMIT-TO(LANGUAGE, "German"))

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Appendix II

Study selection process



2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Appendix III

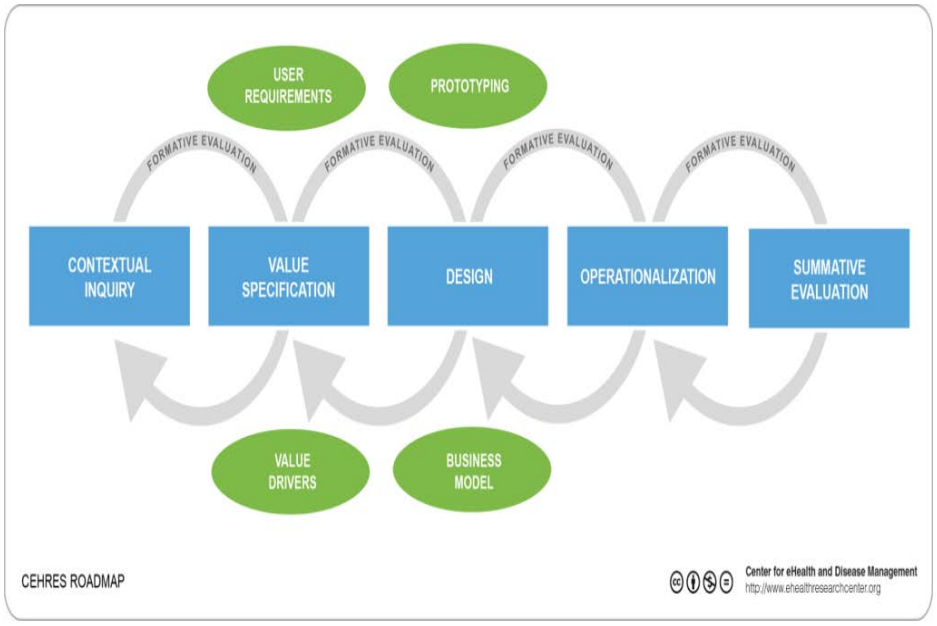
Classification of identified risks

Level	*Risk*	*eHealth application*	*Source*
Human (patient)	Time-consumption	Telecare	Masa et al. (2011)
	Selective benefit	Telecare	Bujnowska-Fedak et al. (2011)
	Selective benefits / negative effect	Web-based counseling	Spijkerman et al. (2010)
	Selective benefits	Telecare	Zimmerman et al. (2011)
	Low adherence	Web-based self-management	Cruz-Correia et al. (2007)
	Low adherence	Telecare	Willems et al. (2007)
	Low adherence / selective benefits	Web-based counseling	Verheijden et al. (2004)
	Low adherence/alliance	eTherapy	Morland et al. (2010)
	Drop-out	eTherapy	Postel et al. (2010)
	Pos. for 2 endpoints / Neg. for other	Tailored web-based counseling	Ruffin et al. (2011)
Technology	Usability	Telecare	Bujnowska-Fedak et al.(2011)
		Self-management via PDA	Nguyen et al. (2008)
	Technical problems	Self-management via PDA	Nguyen et al. (2008)
		Web-based self-management	Cruz-Correia et al. (2007)
		Telecare	Demaerschalk et al. (2010)
	Also time consumption as risk in this study	Telecare	Jansá et al. (2006)
		Telecare	Biermann et al. (2002)
	Technical / Logistical problems	Telecare	Willems et al. (2007)
Organization	Costs	Telecare	Copeland et al. (2010)
	Time-consumption	Telecare	Biermann et al. (2002)
		Telecare	Montori et al. (2006)
	Barriers using the application	PDA-based counseling tool	Strayer et al. (2010)

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Appendix IV

ceHRes Roadmap to improve the impact of eHealth interventions



2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Appendix V

Web-based sources

Sources	urls
International and national health organizations /government agencies	❖ World Health Organization (WHO http://www.who.int/goe/en/); ❖ European Commission (EC http://ec.europa.eu/health/medical-devices/index_en.htm); ❖ UK Department of Health (http://www.dh.gov.uk/en/index.htm); ❖ MHRA (http://www.mhra.gov.uk/); ❖ Scottish Government (http://www.knowledge.scot.nhs.uk/telehealthcare.aspx); ❖ Irish Medicine Board (http://www.imb.ie/); ❖ Bfarm (http://www.bfarm.de/DE/Home/home_node.html); ❖ Australian Department of Health and Ageing (http://www.health.gov.au/internet/main/publishing.nsf/Content/eHealth); ❖ Swedish Medical Products Agency (http://www.lakemedelsverket.se/english/product/Medical-devices/)
Databases	❖ MAUDE (Manufacturer and User Facility Device Experience) database (U.S. Food and Drug Administration) http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfMAUDE/search.CFM ❖ ECRI Health Devices Alerts (HAD) database https://members2.ecri.org/Components/Alerts/Pages/CPIssues/Issue.aspx?CH=1&ChName=Medical%20Devices&rid=0 ❖ ECRI Medical Device Safety Reports (MDSR) database http://www.mdsr.ecri.org/?pnk=healthdevices
Expert centers Medical technology	❖ ECRI Institute; https://www.ecri.org/Pages/default.aspx ❖ Prismant; www.kiwaprismant.nl/ ❖ ZonMw; www.zonmw.nl/
Dutch professional journal on health care	❖ Medisch Contact http://medischcontact.artsennet.nl/home.htm

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Optimized Testing Process in Vehicles Using an Augmented Data Logger

Karsten Hünlich
Steinbeis Interagierende Systeme GmbH
Esslingen, Germany
karsten.huenlich@steinbeis-ias.de

Ulrich Bröckl
University of Applied Sciences Karlsruhe
Karlsruhe, Germany
ulrich.broeckl@hs-karlsruhe.de

Daniel Ulmer
Steinbeis Interagierende Systeme GmbH
Esslingen, Germany
daniel.ulmer@steinbeis-ias.de

Steffen Wittel
Steinbeis Interagierende Systeme GmbH
Esslingen, Germany
steffen.wittel@steinbeis-ias.de

***Abstract*—The growing amount of electronic components in vehicles requires an increasing communication load between these components and hence an increasing load on the vehicles communication buses. Both aspects entail an increasing workload for the test engineer developing and executing test cases to verify the required system behaviour in the vehicle. This article considers a way to automate and reduce the workload for in-vehicle testing by augmenting the functionality of current data loggers. The idea is to use the data logger for supporting the testing process for test drivers. The introduced implementation shows a way to verify the test cases' execution on the fly in order to avoid finding erroneously executed test cases at a later point in time. Additionally, the presented implementation seamlessly includes the test environment for in–vehicle testing into the tool chain, which is already used on lower integration levels. This allows the test engineer to reuse test cases from the lower integration levels in vehicle tests and to compare the results from test runs on different integration levels. The paper describes two stages of the development process of the augmented datalogger and includes the first feedback collected in a case study with a prototypical implementation.**

Keywords – automotie, data logger, intelligent data logger, test case development, test case monitoring

I. INTRODUCTION

This paper offers a closer look at the augmented datalogger and the associated process of in-vehicle testing as they were shown in [1].

Many different data loggers are used in the automotive industry. Primarily, they are designed to record the communication between Electronic Control Units (ECUs) [2]. In more advanced systems, the data content of the Random Access Memory (RAM) of the ECUs is additionally recorded [3]. These data loggers become more and more important to the test engineers because the number of the networked ECUs and hence the testing efforts in a vehicle is continuously increasing. From each requirement on vehicle level, the test engineers have to derive test cases to ensure that the ECUs in a vehicle are performing the correct action within correct time constraints. To check this in an in-vehicle test it is necessary to record the bus traffic and the data content of the ECUs' RAM while executing a test case manoeuvre with a car. The result of the test is determined by evaluating the recorded data.

The amount of collected data can turn the evaluation of the recorded test case data into a time consuming challenge. In current solutions, the result of the evaluation can be classified as "passed" or "failed" In case of a passed classification, the recorded data show that the System under Test (SuT), e.g., an ECU, exhibited the expected behaviour described by the requirements. The classification failed shows a deviation of the measured data from the expected values and hence from the expected test result. But especially if human beings are involved in test execution, the recorded data might be "invalid" if there was a significant mistake during the test case execution. In this case, the evaluation of the recorded data is impossible with respect to the test case's definition.

Figure 1 shows the classification of the test results that can occur in in-vehicle tests. In the first step, the recorded data is usually examined manually by an engineer if it meets the constraints of a valid data record. Some possible cases for invalid data records are:

- The test driver has not driven the test case correctly
- The data logger configuration was incorrect
- The recorded data was incomplete because the measurement has stopped while the tests case was executed

If the recorded data is valid it can be compared with the expected results of the test case. The result of the test evaluation is passed if the system works as expected or failed if the system has not the expected behaviour.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

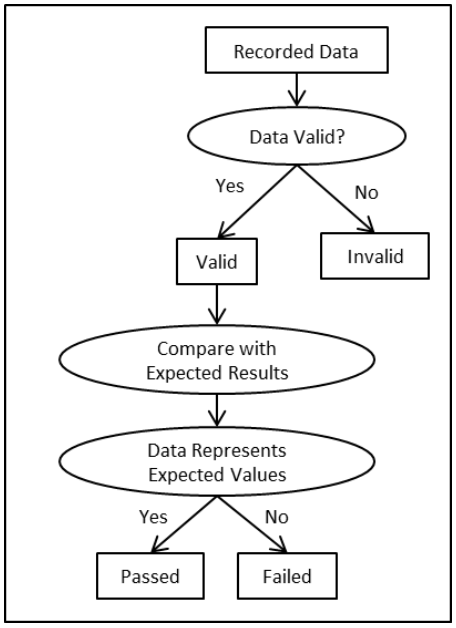


Figure 1. Classification of test results of in-vehicle tests

To minimize the cases of an invalid data record, and therefore the time for the test case execution and evaluation, the data logger can be augmented with additional functionality to monitor the correct execution of the test case. The necessary conditions are to be defined by the test engineer before test execution. This is possible if the data logger can be extended with instructions supervising relevant signals. For these signals boundaries may be defined. A test case can, e.g., be successfully accomplished if the signal stays within these boundaries. However, the goal is not to test the driver's behaviour as mentioned in [4]. The goals are to give instructions to the test case executor, which may be a driver, a robot or a test automation tool, and to additionally supervise the execution's correspondence to the conditions predefined by the test engineer. Especially a human driver is one of the — corresponding to our experiences — biggest error source in a vehicle during a test case execution. In the following, we show how the augmented data logger can help to avoid unnecessary work by evaluating a test case at runtime for being valid or invalid. If the augmented data logger is not only able to supervise the driver while executing a test case, but also guides him through the test case, the augmented data logger even helps to minimize invalid test executions.

In addition to meeting the challenges of in-vehicle testing, the introduced Augmented Data Logger (ADL) shall seamlessly integrate into a typical development process of the automotive industry. A short overview of the relevant aspects shall be given within the next paragraphs.

Figure 2 shows an example of a system development process according to the V-Model as shown in [5]. In this example, the test on vehicle level is the last level of testing within the integration process. Before this stage, many other tests have already taken place on lower integration levels. For efficiency reasons, it would be helpful if the test engineer could reuse test cases developed on lower integration levels, e.g., test cases from Hardware in the Loop (HiL) tests [6]. The reuse of these test cases minimizes the work for the test engineer to adopt the test cases to the desired test platform. The reuse also enables the comparability of the test results from a vehicle test with the results from lower integration levels. For guaranteeing the reusability of the test cases it is essential to specify the test cases platform independently. A test case language is needed, which is both platforms independent and suitable for all testing platforms. Figure 2 shows typical levels in an automotive V-model and the corresponding testing platforms.

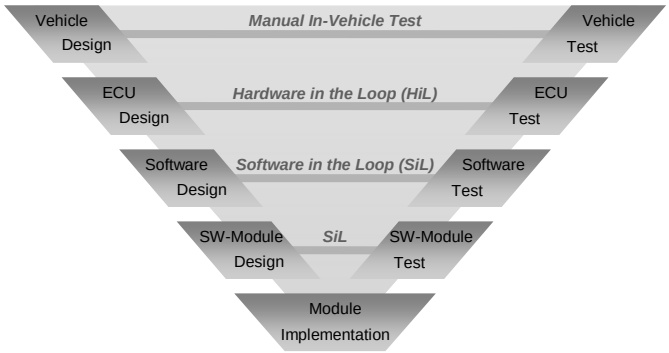


Figure 2. Commonly used application of the V-Model in the automotive industry

The solution described in this article is based on a test case language, which allows the reuse of test cases on Software and ECU levels. Within this article, a solution for extending this approach to "Vehicle Test" is discussed. The solution is based on test cases from lower integration levels by adding information to guide the driver through the test case and by adding instruction to supervise the actions of the driver. The article begins with a description of the state of the art for data loggers and discusses two prototypes of the augmented data logger. The added features are supported by a case study. The paper ends with a summary and ideas for future work.

II. DATALOGGER STATE OF THE ART

Today in-vehicle tests are usually executed without the support of a software tool for giving feedback on the quality of the test execution or a tool that guides the driver through a test case. This conclusion is based on our experience from several automotive companies and suppliers. Instead, the test cases are often written in plain human readable text which describes what a tester has to do in the vehicle to fulfil the test case. These textual test cases are stored for example in a database. For taking a set of test cases to the car, they are either printed out or downloaded to a robust handheld computer. In both cases, they are read before or during a driving manoeuvre. The quality of the execution of the manoeuvre thus depends on the skills of the test driver. Details of the execution quality can be determined offline on a parking lot or by evaluating the information on the data

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

logger. Especially if test driver and test engineer is not the same person, this process is error-prone and time consuming. Since the test cases are in natural language there is enough room for misunderstandings between a test manager who writes the test cases and a test driver who has to execute the manoeuvre. This fact tends to result in multiple iterations of in vehicle tests of the same test case.

There are several solutions that have the aim to optimize in vehicle tests and to minimize the time overhead. A touch-display can be used in vehicles getting rid of the printed check lists and directly sending the results of the test steps to a database. A more advanced system is shown in [7], which comprises of a driver guidance system and a feature to immediately evaluate if the test is passed or failed.

For testing driver assistance functions, manoeuvres have to be executed very precisely by the test driver. That means in a significant number of tests the tests are failed not because the system is not working correctly but the test driver has made a mistake. To minimize this number of invalid tests this paper describes a way to detect deviations of the given test case during its execution. This avoids a usually time consuming evaluation of invalid test cases.

Another solution is described in [8]. This paper describes a system of a car and robot. The robot drives the car inside a restricted area. Within this area the robot performs test cases very precisely. The robot is controlled by engineers from a base station. The approach needs a restricted area because the robot does not recognize its surrounding. This system was developed for executing very dangerous tests, e.g., collision mitigation/prevention tests at high speed rates. Since the system is very expensive and restricted to special test areas it is an addition for human driven cars but cannot replace the human tests.

A. Datalogger Setup

Current data loggers [3] are designed for recording data and neither for interpreting it nor for participating in the testing process. This section describes a way of augmenting the functionality of the data logger in order to support the testing process and to seamlessly integrate the vehicle tests in the system integration and testing process.

A data logger to record digital information in vehicles might be designed in the way described in [9]: i.e., a host computer is connected via a network interface, e.g., Ethernet, to the data logger. Over this connection, the data logger can be controlled and configured. The configuration defines which signals are stored in the data storage and on which bus interface the signals can be received. The host computer is mainly used to start and stop the data logger and to visualize an excerpt of the recorded data on the fly. The data logger hardware is responsible for the real time processing of the data. A commonly found feature is a trigger that starts a measurement when a predefined condition becomes true as it is described in [10].

For evaluating the trigger conditions the data logger needs information about the connected data buses and the data that is transferred over a particular data bus. Usually, this information is available in form of configuration and signal files that are interpreted by the host computer and transferred to the data logger.

In some parts of a data logger execution in real-time is mandatory. This is necessary because the test engineer needs to know exactly when some data have been transmitted on a particular bus. A common solution is that the communication on a bus system is recorded together with timestamps, which indicate the time instance when a message is transferred over a bus [11]. Figure 3 shows the procedure of recording a message from a bus. If the data logger receives a message a timestamp is taken. For the evaluation of the recorded data it is possible to correlate in time the different recordings with the help of the timestamps, which means that the more precisely the timestamp is taken the more precisely the situation can be reproduced and evaluated.

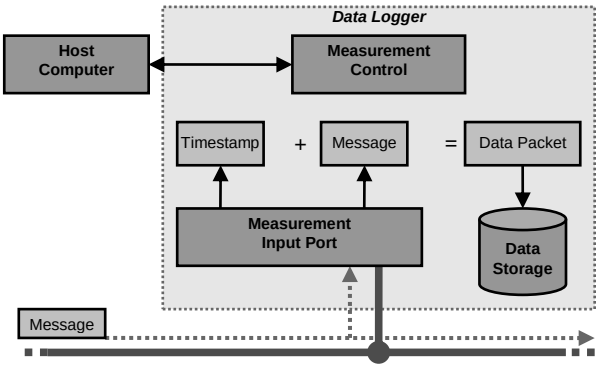


Figure 3. Schematic procedure of measuring a message on the bus

The example in Figure 3 shows a host computer that is connected over a communication interface with the measurement control unit within the data logger. The host computer is commonly a PC or a notebook with an operating system that does not support real time tasks. Via the host computer the engineer has access to and control over the data logger. Additionally, the host computer can access measurement data and visualize them to the user. Evaluating this data while conducting a manoeuvre is almost impossible since, in this case, the driver would have to fully concentrate on the monitor instead on his driving task.

B. Current Testing Process on Vehicles

In the common testing process, the test engineer starts looking at the requirements for the SuT. Based on these requirements the test engineer creates the corresponding test cases. How the test engineer writes down these test cases for in-vehicle tests is mostly not defined. In some way, the test cases have to be readable by the driver while he is executing the manoeuvre in the vehicle. After finishing writing a test case, the test engineer has to hand over the test case to the driver who executes the manoeuvre specified in the test case in the vehicle. This is usually supported by tools, which allow configured testing and sending them, e.g., to handheld devices. This test set is executed by the driver. The role of the test engineer and of the driver might be taken by the same person or by different ones. If the test engineer and the driver are different persons who write and execute a test case, the test case must be well defined to prevent

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

misunderstanding. If the test case specification is not complete and therefore, the driver does not execute the test case as intended by the test engineer, the following work might be unavailing.

After having recorded the data of the manoeuvre that is specified in the test case the driver hands over the recordings to the test engineer. Afterwards, the test engineer evaluates the data. Usually, this is done manually. The test engineer has to search through a database of signals with probably more than 10,000 entries. If the result of the test case is passed, the test case will be documented and closed. In case the result is failed, the test engineer has to find the exact reason. The SuT can either have a bug or the test case has not been executed accurately, which means that the test is invalid. If the test case was executed within all defined constraints by the test engineer the test case is valid and hence failed. Both cases generate lots of work of analysing and documentation for the test engineer. Especially, the work for the first case can be minimized by finding out the validity of the test case in an earlier stage of the process.

Generally, the biggest drawback of finding invalid test runs late in the process is the time that the test engineer spends on one test case. It must be considered that the number of test cases that must be performed for each major release can be up to several hundred test cases. As a conclusion two main issues can be identified that can be possibly optimized:

- The time for evaluating the test results by avoiding invalid test cases
- The number of times moving from the office to the vehicle and to the test track for repeating invalid test case

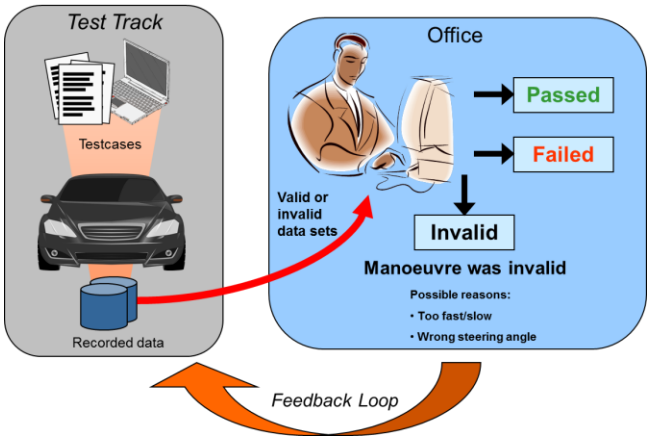


Figure 4. Sample of the current testing process on vehicle level

Figure 4 visualizes the current testing process. The test cases are executed in a vehicle and the recorded data is stored on a local disk of the test system. Later the data is transferred to a computer in the office for evaluation. An engineer evaluates the data and removes invalid data sets. The tests corresponding to the invalid data sets usually have to be executed again. This means going back to the vehicle on the test track. The feedback loop in this example is between two different places, which is time consuming as stated above. One approach to avoid going from the test track to the office and back for several times would be to evaluate the tests in the vehicle after having executed a test set. But while evaluating the tests, the vehicle cannot be used for executing other test sets. Since most of the time test vehicles equipped with measurement systems are rare and have to be shared by many engineers, this approach seems even more inefficient.

The introduced testing process on vehicle level is very different from the test processes on lower integration levels of the development process shown in Figure 1. In the lower levels, i.e., HiL or SiL, a test case is written in a defined way. The test case can be reused and usually returns a reproducible result. Another point is that the test result is directly available after the test has been finished. It can be said that the processes on different levels have mainly five important parts [12]:

- The SuT itself
- Test case execution system
- Environment simulation that simulates the environment of the SuT
- Measurement and data logging system
- Evaluation system

The evaluation system compares the measured values with the ones that are specified in the test case for the SuT. The test case execution system reads the test case and controls the environment simulation that affects the SuT. In a vehicle, the parts for the test process are different. The test case execution system in a vehicle is the test driver. The test driver has control over the environment of the SuT. The evaluation system in a vehicle test is the test engineer who evaluates the measurements.

The measurement and data logging system might be the same as the one used in the vehicle. For the in-vehicle test, an environment simulation is not necessary because the vehicle is used in a real environment. Sometimes both environments are mixed for the vehicle tests, e.g., foot passengers are simulated with synthetic dolls or imaginary sensor information.

III. AUGMENTED DATALOGGER VERSION I

This section introduces the first prototype of the ADL implementation. The focus of this prototype is the implementation of the basic features for giving feedback to the driver. The attached display is not optimized for intuitive feedback and only shows basic text output.

A. System Design

The first design of the data logger version was focused on the test case execution inside the vehicle. In this case, a test case is a sequence of instruction the driver has to execute. It also includes a set of rules that has to be met for a valid execution. Each step is s In case of an invalid hown inside a small display as a text message. In case of an invalid execution of the test case, the driver gets a response and the test case execution stops. A laptop is necessary in this version to control and configure the data logger. Only one test case can be stored on the data logger. So the test case

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

selection and loading has to be done by the driver manually. In this first prototype, the test case description has to be converted by a code generator into executable code before the test case execution can start. This approach was good enough for first experiments but far too slow for efficient in-vehicle testing.

B. *Testing Process Supported by the ADL*

To reduce the time for testing and evaluating of in-vehicle testing a new approach for the testing work flow should be considered. The first aspect is the form how the test case is written. A uniform platform independent language (see Section III C. for more detailed information) is used to define the test cases. With this uniform language, the test engineer can precisely describe the test case. The test case is now not only human readable but also machine-readable and can be interpreted by a program. Additional instructions extend the abilities of the data logger. The system now knows about the manoeuvre that has to be executed for a particular test case. With the knowledge of how a test case must be performed, driving errors can be detected directly and time can be saved.

The new work flow has a strict separation between the office work and the work in the vehicle. Right after performing a test case, the driver gets a result if the test case was executed accurately. The feedback also includes the information why the test has been invalid. This information depends on the test case description from the test engineer. If the test engineer describes the test case in many details more driving errors can be detected without looking at the whole measured data back in the office. The advantage of this new approach is that the driver:

- Is guided through the test case execution process through a unified notification
- Gets a response directly after the manoeuvre if the test is executed correctly and hence valid
- Gets the reason why a test case was classified as invalid

This reduces the evaluation work and the test case execution work. Since the data logger instructs and checks the manoeuvre, it makes the execution more precise.

For this new approach, parts of the evaluation system and the test case execution system are added to the data logger. The schematic of a data logger shown in Figure 2 can be extended to execute additional instructions given by the test engineer, which controls the data logger and guides the test driver through the manoeuvre. Figure 5 shows a simplified version of such a measuring system. The CPU (Central Processing Unit) has to fetch the messages from the bus, add a timestamp to each message and extract relevant signals. The values of the signals are internally decoded from the coded bus signals and provided for the test case code.

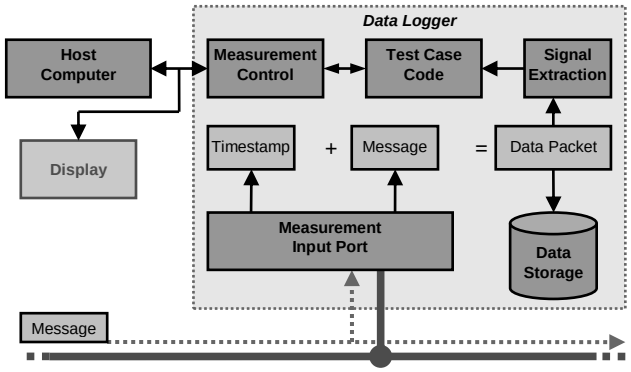


Figure 5. Schematic measuring system extended with the test case code

To control and configure the data logger the test case needs a connection to the measurement control module. On the first hand, the measurement has to be started at the beginning of a test case and stopped when it has ended. On the other hand, the measurement control module is responsible for monitoring the execution of the test case. In detail the measurement control module compares target values defined in the test case (in the following called "rules") with the corresponding signals transmitted on the vehicle bus. Furthermore, the measurement control module generates instructions for the driver depending on the current test step within the test case. These instructions are extracted from the test case and are provided to the driver, e.g., via a display. The ADL version 1 has an attached display that shows only human readable text generated from the machine readable test case description.

Figure 6 shows the testing process corresponding to an augmented datalogger. The test case is supplemented with instructions for the driver and with conditions for being valid. Based on this the test engineer is guided through the test while driving the car and the evaluation of the test case for being executed correctly is done by the data logger on the fly. Immediately after a violation of a rule within a test case the test driver gets informed and has the choice whether to finish the manoeuvre or stop immediately and start from the beginning.

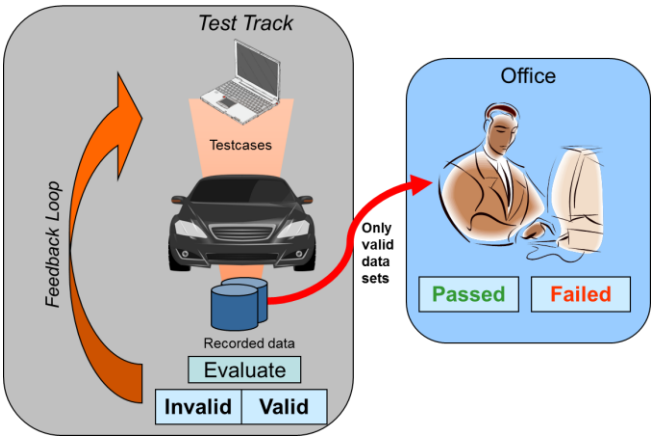


Figure 6. Optimized testing process

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

C. Test Case Implementation and Execution

In this section, the test case implementation and execution is shown using the following example:

Test Step 1:	Start engine
Test Step 2:	Accelerate to 60 km/h
Test Step 3:	60 km/h reached?
Test Step 4:	Full braking
Rule:	Steering wheel straight

Such a manoeuvre is used, e.g., to measure data of an Anti-Blocking System (ABS) and to evaluate if it has performed accurately during its intervention. A possible criterion for an invalid ABS-test execution is defined by looking at the steering angle. If the data show that the car did not drive straight, the test case has not been executed accurately. The manoeuvre can be described in a state chart manner represented by an XML (Extensible Markup Language) file [13].

The example in Figure 7 shows the ABS-manoeuvre in XML code. The definition of the XML code is described by Ruf [14] for Hardware in the Loop tests. The test case is composed of states, actions, events and rule. For the above test case the rule checks the steering wheel angle during the whole test case. The states are following in chronological order. Each state has one or more actions that have to be performed by the test driver. If the condition of an event is fulfilled the state machine enters the next state.

```
<?xml version="1.0" encoding="UTF-8"?>
<Testcase xmlns="http://www.ebtb.de/adl"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.ebtb.de/adl
http://www.ebtb.de/adl">

  <Rule SteeringAngle_deg_eqal="0" Tolerance_deg="5"/>

  <State num="1">
    <Action text="Get ready to start the manoeuvre"/>
    <Event wait_seconds="5"/>
  </State>

  <State num="2">
    <Action text="Start the engine"/>
    <Event wait_seconds="5"/>
  </State>

  <State num="3">
    <Action text="Accelerate to 60km/h"/>
    <Event velocity_kmh_equal ="60"/>
  </State>

  <State num="4">
    <Action text="Full braking"/>
    <Event velocity_kmh_equal ="0"/>
  </State>

  <State num="5">
    <Action text="Turn-off engine"/>
    <Event wait_seconds="5"/>
  </State>

  <State num="6">
    <Action text="Manoeuvre finished"/>
    <Event wait_seconds="3"/>
  </State>

</Testcase>
```

Figure 7. Listing of a test case in XML

D. Case Study

After having implemented a prototype of the described data logger with its additional features, a case study has been performed to determine the benefits of the augmented measurement system for test drivers. The case study was conducted with a group of eleven candidates. The group consisted of team leaders, developers and testers. In the first step, the content of the executed XML test case was explained to the candidates. With this knowledge the candidates were guided by the augmented measurement system to execute the test case in the role of a test driver.

1) Manoeuvre

The selected manoeuvre for the case study was more complex than the sample test case in Figure 7. The test addresses the safety deactivation of the cruise control when engaging the hand brake. For one test case example the test steps are as follows:

Test Step 1:	Start engine
Test Step 2:	Accelerate to 50 km/h
Rule:	Speed less than 55 km/h
Rule:	Steering wheel straight
Test Step 3:	Activate the Cruise Control with the "SET" button
Test Step 4:	Remove foot from acceleration pedal
Rule:	Don't turn the Cruise Control lever up
Comment:	*Cruise Control active*
Test Step 5:	Engage the hand brake
Rule:	Speed more than 45 km/h and less 55 km/h
Comment:	*Cruise Control disabled*
Test Step 6:	Decelerate to zero
Test Step 6:	Turn off engine

In this test case, the Driver should accelerate to the target speed of 50 km/h. But in the test the driver should not accelerate to a speed greater than 55 km/h and should not turn the steering wheel.

In most cars with a Cruise Control lever, the function can be activated on two ways:

1. Pressing the "SET" button to use the current speed as reference
2. Tip the lever up to activate the Cruise Control and accelerate or to resume to the speed set before

The implementation might differ between manufacturers. But if there is more than one way to activate the Cruise Control the test case shall address exactly one. The other ways are different test cases.

Engaging the hand brake turns the vehicle into the following situation: The brake leads to a vehicle deceleration. If the Cruise Control will not be disabled the controller tries to match the speed that is set and accelerates. To avoid this situation the Cruise Control has to be disabled if the hand brake is engaged.

2) Feedback

The vehicle for the case study was equipped with an extra display that is attached to the windscreen. The setup in the vehicle looks similar to an external navigation system. In this setup, the display shows the instructions and the current state of the running test case. The execution of the test case was done on a locked test track. This ensures a save

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

environment and that the candidates are not disturbed by surrounding vehicles.

After executing, the test cases multiple times the candidate was interviewed about his experience with the augmented measurement system. The collected feedback is summarized in Figure 8.

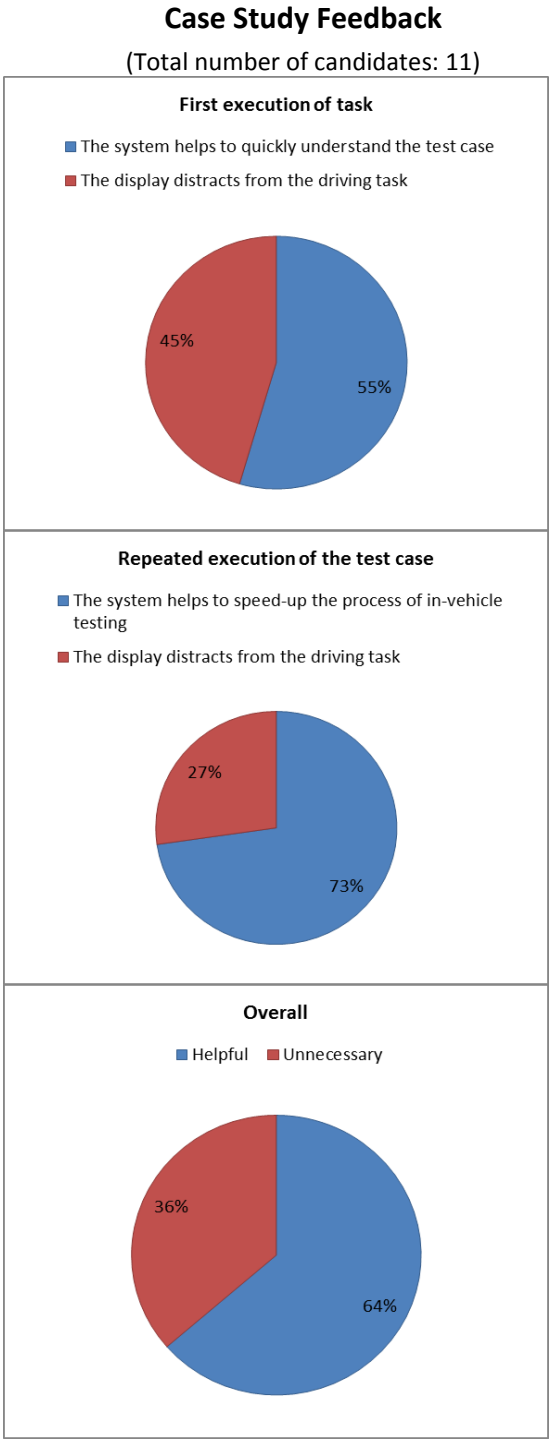


Figure 8. Case Study feedback results

Most candidates are confused and distracted by the information shown in the display driving the test case for the first time. The reason might be that the candidates do not yet intuitively follow the instructions on the display. As soon as the instructions are known to the candidate he can concentrate less on the display and more on his driving task. After a short learning curve the confidence and sureness working with the augmented data logger raised. In summary, 7 candidates are seeing a benefit of such a system to speed up and assist them in their daily work.

Furthermore, the feedback also includes suggestions for improvements. The four mostly mentioned suggestions were:

- Additional speech output for instructions
- Direct connection to quality and lifecycle management tools
- More detailed information in case of a invalid result
- Using LCD glasses instead of a display attached to the windscreen
- Adding a test case automation for processing several test cases in a sequence

The feedback of the case study indicates that the augmented data logger helps to speed-up the testing process for in-vehicle testing.

IV. AUGMENTED DATALOGGER VERSION II

The second version of the ADL has several improvements. The display shows more detailed information of the current state. To handle these information most actions, events and rules are displayed as icons. This might increase the reaction time of the driver while executing a first manoeuvre but after getting used to the icons they can be instantly understood. The icons have been designed in cooperation with the University of Applied Sciences Karlsruhe. They shall be intuitive to new drivers. It is planned to add the speech output for instructions at a later time and optimize the required visual aspect. As an additional feature in the second prototype a testing automation has been implemented. The driver has the possibility to execute several test cases in a sequence. Finally, the implementation can filter test cases from lower integration levels and skip test cases that are not suitable for the in-vehicle setup.

Another major improvement of the ADL version II is the way how test cases are loaded in the data logger. The first version generates code from an XML test case description and executes that code on the data logger. In the second version, the XML test case is transferred to the data logger. The data logger interprets the test case and executes it directly. This is a big benefit because the code generation step is no longer necessary.

A. *Display icons*

Figure 9 shows an example how a test case state might look on a display. Two actions should be executed:

- Switch gear selector in state “D”
- Accelerate to 80 km/h

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The event occurs if 75 km/h are reached. The grey number indicates the degree of fulfilment. The two "R" inside the squares depict the active rules:

- Steering wheel straight ±5 degrees (Tolerance is not shown in Figure 9)
- Speed less than 85 km/h

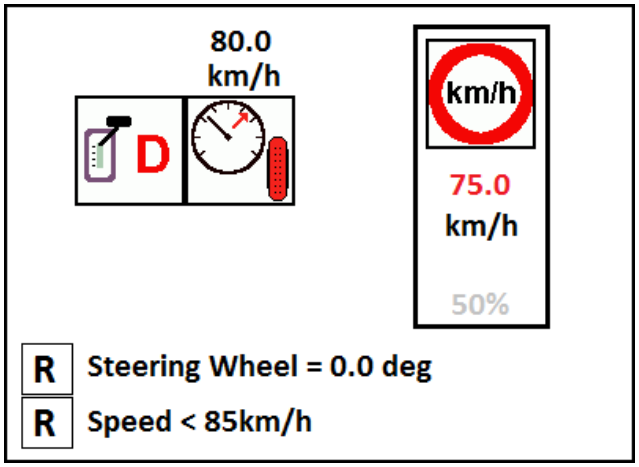


Figure 9. Possible display screen with actions, rules and one event

If a rule is not fulfilled the test case execution stops and shows a red screen with the broken rule. If all test steps are executed and no rule is broken the driver gets a green screen. Both the green and the red screen terminate with the test automation screen.

B. Test Automation

To give the test driver the ability to easily switch between the available test cases on the data logger, a test case automation system [15] was implemented and is shown in Figure 10.

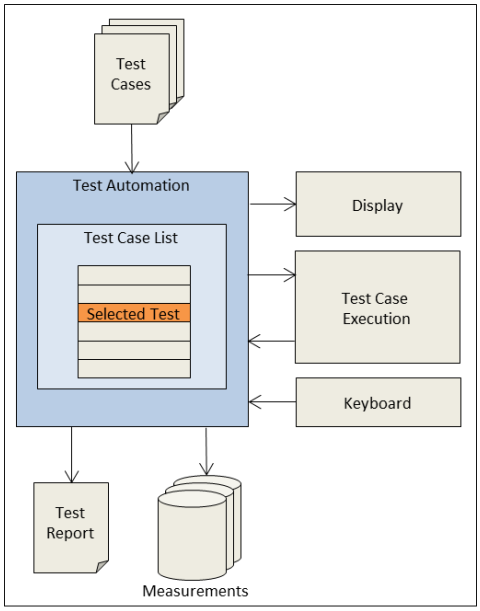


Figure 10. Test case sequence automation

The test driver can select a list of test cases for execution and upload these test cases to the data logger. Beginning with the first test case the driver performs all tests one after another. If the test is passed, the following test case is loaded for execution. In case of a failure, the driver has the choice to drive the test case again or to go to the next test case.

The results of the test runs are stored and will be presented in a report showing the valid and invalid test cases. The measurements are stored for both cases, valid and invalid tests. Running one test case multiple times will produce multiple test reports and measurements. It is up to the test engineer to select the relevant reports and measurements for evaluation.

C. Test Case Filters

As explained in the sections before, the test cases for in-vehicle testing can be reused from lower integration levels. Since the underlying test case language is manoeuvre-based, a large number of test cases can be reused without any change. But there are test cases that cannot be reused directly.

One reason is that test cases might use special actions that cannot be performed in each vehicle setup. These types of test cases can be called "platform dependent test cases". For example on a HiL platform the bus signals sent from the HiL's bus interface can be manipulated easily because the HiL simulates all other ECUs on the specific bus. In a vehicle these ECUs are existent. This means that the vehicle needs a special hardware that separates the bus between the SuT and all other ECUs on the bus. This hardware manipulates all incoming messages and signals to the SuT as specified in the test case. If this hardware is not available within a certain test vehicle a test case, which needs this signal manipulation cannot be executed.

Another example is a hardware interface manipulation. Some HiL platforms are able to apply hardware errors to the interface of the SuT such as a defective contact. These tests are platform specific and as well as can only be tested automatically inside a vehicle if the corresponding manipulation hardware is installed.

Complex test cases are using a predefined environment to test driver assistance functions. The environment can be a given driving track or surrounding objects like other vehicles, motor cycles and trucks. Lower integration levels are using simulated sensor ECUs to simulate the environment. Inside a test vehicle the sensors are real and will detect the given environment. A basic example is an adaptive cruise control manoeuvre. The test vehicle is behind another vehicle — called object vehicle. The object vehicle accelerates or decelerates and the test vehicle should do the same automatically to keep a safe distance between the two vehicles.

One approach for testing the acceleration algorithm in a vehicle is replacing the sensor ECUs with the environment simulation of a HiL. These test cases can only be executed in an in-vehicle test if the vehicle is modified as described. Another idea, which is left for future work is to distribute the test case to several ADLs in several vehicles.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The three examples show the ADLs ability of executing a test case is depending on the content of the test case and the setup of the vehicle. With the help of a filter test cases can be automatically sorted corresponding to the required test equipment.

A different use case for applying a filter is the way how the test engineers write the tests. As explained above, the test cases can have a variable amount of actions in a state that will be performed simultaneously. On SiL/HiL platforms it is possible to perform many actions at the same time because the simulated driver can perform the actions simultaneously. A real test driver gets all the information what he has to do in a certain state at once and has to perform all these tasks as fast as possible. The more actions are within a test case state the more tasks the test driver has to execute. He has to gather all the information and perform the required physical actions. The risk to forget to execute an action rises with the amount of actions in a state.

One result of the work with the University of Applied Sciences Karlsruhe has been that there should be only one action in a state, which has to be executed by the driver. Skilled drivers might be able to execute up to three actions. Since the test cases can be executed on HiL-platforms as well, it is important to know that Actions are executed on a HiL platform immediately after entering the state. A human driver has recognition and response times. These times are rising with each additional action. First tests and the case study show a very high range of the reaction times. These reaction times may be critical to get a valid test case result. An analysis of existing test cases showed that even the HiL test cases are modelled with a maximum of three driver actions per state. This means that the display layout can be optimized to show one to three actions at a time. Due to these limitations the number of actions within a state is limited to three for test cases that shall be used on vehicle level. Test cases with more than three driver actions per state are refused. An example, which shall clarify this statement, is the following test case:

1. Switch the gear selector to “R”
2. Press the turn indicator to right blinking
3. Press the brake pedal
4. Release the tightened parking brake

For a human driver it is almost impossible to perform all these actions simultaneously. In this case, the test case writer has to check the test specification whether the actions can be split into two states without altering the expected test case results. According to our experience splitting one state with several driver actions into several states with one driver action is mostly possible.

There are two ways to address the limitation of driver actions within the software tools, either globally limiting the allowed amount of driver actions within one state to three or by adding a filter to the in-vehicle test automation, which suggests skipping test cases with more than three driver actions. One topic for future work is to automatically detect and convert these test cases and to inform the author immediately after writing such a test case that an in-vehicle execution is not possible. Based on this approach, one future work might be to automatically forecast the dynamic criticality of a test case. The dynamic criticality is a factor that indicates how risky the execution of the test case is for the driver himself and the surrounding environment. Manoeuvres marked with a high dynamic criticality have a high potential for injury and vehicle or environmental damage. For example, a test case that requires high deceleration or gear movements at high velocities might be automatically marked as dangerous and only suitable for adequate test tracks.

This approach will enable the test automation to filter the test cases for the required environment. For example, all test cases can be executed, which have to be driven on a high-speed track. A first prototype of this semantic filter of the test cases has been implemented. Defining a metric for the combination of all driver actions and its evaluation is left for future work and prototypes.

V. Conclusion and Future Work

This work shows an approach how the process of in-vehicle testing can be improved. The introduced approach shows a way to reduce the costs for the testing process by reusing test cases from other testing platforms and by optimizing the workflow of in-vehicle testing. As a rule of thumb, we experienced that for complex testing scenarios comprising about 100 test cases over 30 per cent of the test cases are invalid when they are evaluated manually after test execution. A major part in the optimized workflow is the possibility for declaring a test case invalid.

The extended classification of a test case enables an early feedback about the quality of the executed test case and hence makes sure that only valid test cases are evaluated. In the introduced approach, a test case can be classified as "passed", "failed", "valid" and "invalid". The first two classifications are based on the requirements and can only be evaluated if the data is valid for the SuT, while the other two classifications reveal if the test case is executed within defined constraints that are based on additional testing requirements. The test engineer has only to look at the measurements of the test cases that are classified as valid. This helps to reduce the evaluation time especially if the test case manoeuvre is very complex or time critical.

A first prototype of the Augmented Data Logger has been discussed, which allow to use test case descriptions from lower integration levels and use them as a basis for the in-vehicle test. The test engineer needs no knowledge in programming languages for implementing and running a test case on the introduced augmented data logger.

While driving a test case the test driver has precise instructions on his current tasks and is guided through the test case manoeuvre. The test driver has immediate feedback if the constraints of the test case added by the test engineer are fulfilled. The augmented data logger observes the execution and the driver gets a response if the manoeuvre is valid or if the test driver has made a mistake during the execution. It is then up to the test driver to decide if he wants to immediately repeat the manoeuvre or continue with the next test case.

A case study shows that the approach is useful and has potential for improvements. The second version of the ADL

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

improves the visual recognition by using icons instead of text messages. The tool chain has been extended by a test automation that supports the driver by defining test sequences that can be executed at once.

The use of test cases from lower integration levels shows that they can be reused if the technical conditions are met. To detect these conditions the idea is to implement filters for the test cases. A filter can select the test cases that are suitable to run in the test vehicle.

For future work, a distributed ADL can be considered to support the in-vehicle test of advanced driver assistance systems where several vehicles are involved. Furthermore, augmented reality glasses instead of a display might be considered for informing the test driver. A semantic interpretation of the test cases might help to forecast the dynamic criticality of a manoeuvre and to recommend a test track.

REFERENCES

[1] K. Hünlich, D. Ulmer, S. Wittel, and U. Bröckl,, "Optimized testing process in vehicles using an augmented data logger", IARIA ICONS Conference, Febuary 2012, ISBN 978-1-61208-184-7

[2] K. Athanasas, "Fast prototyping methodology for the verification of complex vehicle systems", Dissertation, Brunel University, West London, UK, March 2005

[3] S. McBeath, "Competition car data logging: a practical handbook", J. H. Haynes & Co., 2002, ISBN 1-85960-653-9.

[4] L. Petersson, L. Fletcher, and A. Zelinsky, "A framework for driver-in-the-loop driver assistance systems", Intelligent Transportation System Conference 2005: Proceeding of an IEEE International conference Vienna (Austria), September 2005, pp. 771 – 776.

[5] E. Meier, „V-Modelle in Automotive-Projekten, AUTOMOBIL-ELEKTRONIK", Journal, February 2008, pp. 36 – 37.

[6] M. Schlager, „Hardware-in-the-Loop simulation", VDM Verlag Dr. Mueller e.K., 2008, ISBN-13: 978-3836462167.

[7] mm-lab, "Driver guidance system", Automotive Testing Technology International, September 2009, page 89.

[8] H-P. Schöner, S. Neads and N Schretter, "Testing and verification of active safety with coordinated automated driving", NHTSA ESV21 Conference 2009, http://www-nrd.nhtsa.dot.gov/pdf/esv/esv21/09-0187.pdf

[9] J. Park, and S. Mackay, "Practical data acquisition for instrumentation and control systems", An imprint of Elvester, 2003, ISBN-10: 075-0657-960.

[10] M. Koch, and A. Theissler, "Mit Tedradis dem Fehler auf der Spur", Automotive Journal, Carl Hanser Verlag, September 2007, pp. 28 – 30.

[11] D. Ulmer, A. Theissler, and K. Hünlich, "PC-Based measuring and test system for high-precision recording and in-the-loop-simulation of driver assistance functions", Embedded World Conference, March 2010.

[12] S. Dangel, H. Keller, and D. Ulmer, "Wie sag' ich's meinem Prüfstand?", RD Inside, April/Mai, 2010.

[13] B. Ruf, H. Keller, D. Ulmer, and M. Dausmann, "Ereignisbasierte Testfallbedatung - ein MINT-Projekt der Daimler AG und der Fakultät Informationstechnik". spektrum 33/2011, pp. 68–70.

[14] B. Ruf, H. Keller, D. Ulmer, and M. Dausmann, "Ereignisbasierte Testfallbedatung", Spektrum 33/2011, pp. 67 – 68.

[15] M. Spachtholz, "Mission Control - Automatisiertes Testen von Fahrerassistenzsystemen im Fahrzeug", Bachelor Thesis, University of Applied Sciences Esslingen, 2012

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Modeling and Synthesis of mid- and long-term Future Nanotechnologies for Computer Arithmetic Circuits

Bruno Kleinert and Dietmar Fey
Chair of Computer Architecture, University of Erlangen-Nürnberg, Germany
{bruno.kleinert,dietmar.fey}@cs.fau.de

***Abstract*—The paper presents a comparison between two future nanotechnologies that are suitable for arithmetic computation and non-volatile memory. An automatic synthesis procedure of an optical computing design principle onto long-term future Quantom-dot Cellular Automata (QCA) is presented. The goal of this work is to provide a contribution for the elimination of the lack of automatic design procedures for regular build-up QCA arithmetic circuits. A SystemC model of the mid-term future memristor technology is presented, to demonstrate the benefit in space efficiency as a four-value logic memory in a fast signed digit (SD) adder for a hardware implementation of the coordinate rotation digital computer (CORDIC) algorithm. A comparison between QCA and memristor technology presents the advantages of memristors in multi-value logic environments. In this sense, this work is a contribution to ease the automatic synthesis and choice of future nanotechnologies for arithmetic circuits.**

Keywords*—Nano computing, Memristor computing, Optical Computing, Quantum-dot Cellular Automata.

I. INTRODUCTION

MODERN computing devices, like processors or Systems-on-a-Chip are getting more and more powerful. Further raising clock frequencies but also energy-saving requirements for embedded and handheld devices, like smartphones and tablet PCs, push the state-of-the-art CMOS technology to its limits, concerning data throughput and manufacturing densities. At the moment of this writing, classic CMOS technology is close to frequency and density limit and new computing and memory technologies need to be developed and researched. A common answer on how to continue in the post-CMOS era, are nanosystems, that are predicted to allow higher manufacturing densities, like self-organization processes, higher clock frequencies and better energy efficiency [1].

Therefore, we investigate two different promising and complimentary nanotechnologies, each of which offer new possibilities for the design of arithmetic circuits in the post-CMOS era. This is, on one side, a mid-term solution, based on new storing capabilities, namely memristor technology, which offers to store multiple different values in a single storage device. This new feature, that is not offered by CMOS memory devices, can be exploited to speed up arithmetic circuits based on signed digit logic, which is not efficiently possible with current technology. On the other side, there is another new nanotechnology, that is to be considered as a long-term alternative, the Quantum-dot Cellular Automata (QCA) [2]. This technology is characterized by the potential of extremely low-power consuming logic cells, based on single electrons entailed in quantum-dots and a possible high-dense arrangement of such cells.

Both technologies lack support by design tools, which is obvious since they are new technologies. Therefore, we contribute in this paper for a removal of this lack. To support an automatic synthesis of arithmetic logic in QCA circuits, we identified an analogy to another unconventional computing technology: Symbolic Substitution Logic (SSL) [3]. It comes from optical computing and shows a lot of similarities concerning regular setup on pixel, respectively QCA cell processing schemes, that can be used to adapt SSL design techniques to synthesize QCA circuits. On the other side, we have the much more mature memristor technology [4] that can be compatibly manufactured with CMOS circuits. Therefore, we consider it worthy, to research on modeling techniques on the digital level, to allow the integrate memristors in an adequate manner to conventional CMOS circuits. We chose the SystemC modeling language for that purpose as it offers enough flexibility, to model the properties of multi-value memristor-based memory with appropriate data structures.

In this paper, we compare both nanotechnologies in the context of automatic design patterns and simulation of basic circuitry to derive building blocks that can be used to build complex logic circuits. We successfully applied Symbolic Substitution Logic (SSL) as a regular design pattern on QCA and present an abstracted prototype model of a memristor for SystemC digital system simulations. We identified challenges for the development of hardware design and synthesis tools to be reusable for the development of memristor-based systems and later on for QCA technology based systems.

The rest of the paper is organized as follows. In Section II, we present the basic principles of digital optical computing based on SSL. In Section III, we present the basic principles of digital optical computing based on SSL. In Section IV, we explain nanotechnology information processing based on QCA. In Section V, we present the mapping process between SSL rules and QCA cells for the example of one stage of a bit-serial QCA adder deduced from an SSL adder. Details and possibilities with memristors are presented and described in Section VI. In Section VII, we present and explain our abstract

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

model of a memristor for digital circuit simulations. Section IX concludes our findings and points out future work.

II. Symbolic Substitution Logic

Symbolic Substitution Logic (SSL) was invented by Brenner et al. [5] in 1986 as a new method for the design of optical computing circuits. It was exactly tailored to the constraints and possibilities of a high-dense pixel parallel processing offered by optical hardware. The idea behind SSL is to search for a certain binary pattern within a binary pixel image and to replace the found patterns by another pattern. This substitution process can be exploited to realize a digital arithmetic in a highly parallel manner. The key features of SSL are characterized by their strong regularity concerning the pixel processing and the focusing on operating on elementary binary information cells, namely pixels, arranged in a grid structure.

In particular, this situation is also given in Quantum-dot Cellular Automata (QCA) [6]. QCA is one of the promising nanotechnologies besides carbon nanotube field effect transistors and further nanodevice technologies based on tunneling effects that are considered as candidates for a new device technique to realize logic circuitry in the post CMOS area. Analogue to an optical computing scheme like SSL QCA are characterized by a highly dense implementation of binary information cells and a regular information flow. Whereas the elementary binary information cell in SSL was a pixel, which is either bright or dark, the binary information cell in QCA corresponds to two electrons, which are arranged in two distinguishable directions in a four dot quantum cell.

In literature, a really large number of proposals for QCA arithmetic circuits can be found, which have been developed largely manually (e.g., [7], [8]). However, there is still a lack of design methodologies that can be used for an automatic design process of arithmetic circuits based on QCA. There is an exception presented in [9], which proposes a methodology how to convert Boolean sum-of-products in an algorithmic way to QCA logic, in particular to QCA majority gates, which is the basic gate structure in QCA (see Section IV). However, most of the QCA arithmetic circuits are still developed in a time consuming try-and-error process by hand.

On the other side there was a lot of research in the 1980s and 1990s in the Optical Computing community on SSL (e.g., [10], [3]), which brought numerous proposals for digital optical computing circuits based on the basic SSL logic building block, the so-called SSL rule (see Section III). Due to this fact and the similarities given in the kind how elementary information is handled in QCA and SSL, we present in the following sections on-going research on developing strategies how SSL rules can be used for an automatic mapping process onto QCA circuits, which can be used in future design tools.

III. Optical computing with SSL

SSL [10], [3] has drawn a lot of attention during the 1980s and 1990s as a method for exploiting the space invariance of regular optical imaging systems for the set-up of digital optical hardware. The base of information processing in an SSL is the implementation of a so-called SSL rule. An SSL rule depicts a pattern substitution process and consists of two parts, a left-hand side (LHS) and a right-hand side (RHS) pattern (see Figure 1). By a corresponding optical hardware each occurrence of the LHS pattern is searched within a binary image and is replaced by the RHS pattern. Figure 2 shows schematically a possible optical set-up for the search process as it was frequently realized in SSL hardware demonstrators. The principle processing works as follows.

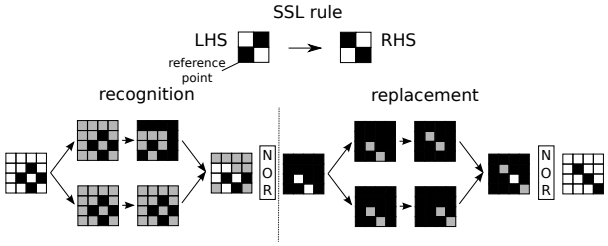


Fig. 1. Principle of SSL

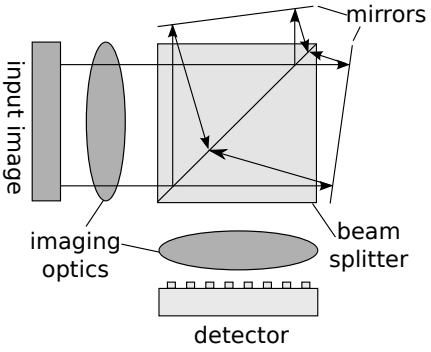


Fig. 2. Implementation of SSL with optical hardware

For each switched-off pixel, i.e., a black pixel, in the LHS of an SSL rule a copy of the image is produced, e.g., by a beam splitter. Furthermore, a reference point is defined within the LHS pattern, e.g., the lower left corner pixel. Each of the copies is reflected, e.g., by tilted mirrors, in such a way that the copies are superimposed and pixels, which have the same relative position to each other as defined in the LHS pattern, meet at the same location.

For the example of Figure 1, this means that one copy of the image is not tilted since it corresponds to the set pixel in the LHS pattern, which is already localed in the reference point. Whereas the other copy is shifted by the tilted mirror, such that each pixel in the copy of the input image is shifted one pixel position down and left. At each position, where two dark pixels meet, an occurrence is given of the LHS pattern in the original input image. The superimposed image is mapped onto an array of optical threshold detectors. Each detector operates on one pixel of the superimposed image as a NOR device. The detector output is used for switching on a LED or laser diode. As a result, one gets a high light intensity at each pixel position, which corresponds to the occurrence of the LHS search pattern in the input image. We denote this new image as a detector output image.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

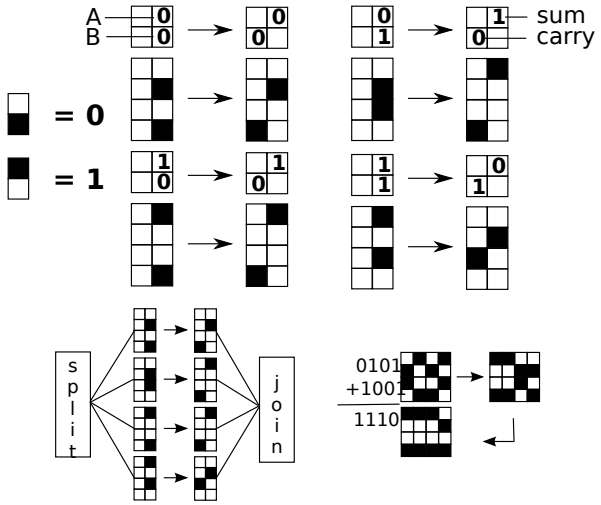


Fig. 3. Realization of a ripple carry adder with SSL. For reasons of improved robustness a dual rail coding is used for 0 and 1.

The recognition step is followed by a replacement step, which works analogue to the recognition step but in opposite direction. For each switched-on pixel in the RHS pattern a copy of the detector output image is again produced by optical beam splitter hardware in such a way that the copies are shifted towards the switched-on pixel in the RHS pattern. This means for the example of Figure 1 that two copies from the detected output image are generated and each of the copies are shifted one pixel up resp. right before superimposing the copies. Once again, the superimposed image is mapped onto a pixel-by-pixel operating NOR detector and LED/laser diode array. The reproduced output is a new image, in which each occurrence of the LHS pattern in the original input image is substituted by the corresponding RHS pattern.

Implementing appropriate SSL rules by splitting the input image into multiple optical recognition and replacement paths, which are applied simultaneously and joined at the end, have been used for the proposing and realizing of digital optical computer arithmetic circuits. Figure 3 shows this schematically for an optical ripple carry adder based on SSL. A large number of further arithmetic circuits using SSL or similar techniques like optical shadow logic [11] have been published in the past for optical adders, multipliers or image processing tasks. All these proposals can be used to transfer them to QCA due to the similarities between SSL and QCA we outlined above.

IV. Nanocomputing with QCA

The elementary information cell in a QCA is a kind of container that groups a few quantum dots, at which charged particles, i.e., electrons, are fixed (see Figure 4). Mostly a QCA cell consists of four dots, in which two electrons are grouped in opposite order. Consequently, the cell knows exactly two polarization adjustments, which are assigned to the binary values 0 or 1. Due to quantum mechanical rules it is possible that a cell can switch between the two states by tunneling of the charged particles between the dots. Concerning the two different particle arrangements one distinguishes between type 1 and type 2 cells (see Figure 4).

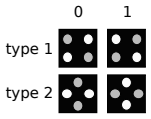


Fig. 4. Binary coding in QCA cells. White circles correspond to empty quantum-dots, gray ones represent dots occupied with electrons.

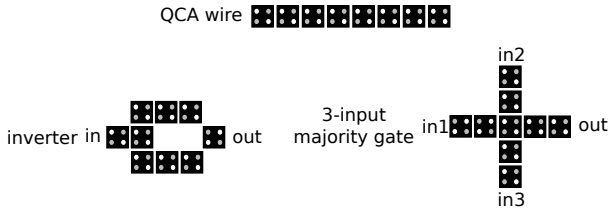


Fig. 5. QCA logic building blocks

A QCA cell serves not only as an information storage cell but also as a transport cell since neighboring QCA cells interchange by Coulomb forces. This means that a cell, which is fixed to a certain polarization, transfers its state to a neighboring cell because this arrangement shows the minimum electrical field energy between neighboring particles of the same charge. Consequently, a QCA wire can be built up, in which information is transported not by an electric current flow but by subsequent reordering of the quantum states in neighboring QCA cells. Due to the fact that no current is flowing and due to the small dimensions of a QCA cell, this technology offers very low power dissipation. Besides information transport one also needs logical gates to realize computing circuits. QCA logic utilizes an inverter and a so-called majority gate for this purpose. Figure 5 shows an inverter built with cells of type 1. In both circuits, the output cell adopts the opposite state of the input cell state, again due to Coulomb forces. In contrast to CMOS circuits, QCA gate logic is not based on the switching of parallel and serial connected transistors but on the states of the cells surrounding a certain QCA cell, serving as output cell of the gate. The majority of the states in these surrounding cells determines the state of the output cell. In Figure 5, a 3-input majority QCA gate is shown. The output cell adopts the same state, which at least is stored in two of the three neighboring cells. By fixing one of the inputs to a certain polarization 2-input AND, OR, NAND and NOR gates can be built.

Based on these three building blocks, QCA wire, QCA inverter and QCA majority gate, various proposals exist in literature for different typical digital circuits like adders, multipliers, shifters, multiplexers and registers, which have been found in a more or less try-and-error procedure. A very impressive collection of computer arithmetic QCA adder and multiplier circuits can be found in the work made by Hänninen [12]. The solutions proposed in this work are distinguished by their regular set-up that helps to realize QCA cells in the future.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

This is an important feature since QCA technology has a long-term perspective concerning its realization with real hardware.

Also design tools, which support the automatic synthesis of regular built-up QCA circuits, will encourage and give hints to device technologists how QCA technology should develop in the best way. In this sense, we propose to use optical computing SSL design procedure as a design entry point for the systematic design of nanocomputing QCA logic. How this mapping can be done is presented in the next section.

V. Mapping SSL rules to QCA logic

The procedure to map SSL logic to a regular built QCA layout is subdivided in three steps. These steps correspond (i) to the core of the logic circuitry, namely the synthesis of an SSL rule into an equivalent QCA circuit, (ii) the realization of the splitting process, because we want to realize systems, which apply multiple SSL rules simultaneously, and (iii) the realization of the join at the end of the recognition-substitution stages. We will demonstrate the generic approach for these mapping steps in the following subsections without loss of generality on the example of the ripple carry adder from Figure 3. Furthermore, we will use this example also to show generic applicable optimization measures for mapping SSL rules, which saves otherwise necessary QCA logic resources.

A. Mapping the split stage to QCA cells

As shown in Figure 3, the applying of multiple SSL rules starts with a split function. The mapping of the split stage onto QCA logic can be done in a straightforward manner. Producing copies of input cells can be simply done with branches of QCA wires running orthogonally to the input QCA wires. If one has to copy more than one input, as for example for the LHS rules in a ripple carry adder, one has to observe that crossing branches can interchange without conflicts. This can be done by crossing lines between QCA cells of type 1 and type 2. To connect both types of cells, a QCA cell has to be shifted by half height of a cell (see Figure 6, part split).

B. Mapping SSL rules to QCA cells

The mapping of SSL rules onto equivalent QCA layouts is divided in two substeps, (i) the mapping of the recognition step and (ii) the mapping of the replacement step. The recognition of an LHS of an SSL rule is mapped to an equivalent QCA majority gate realizing an appropriate AND gate. The number of inputs of this AND gate depends on the number of values in the LHS. For example, the number of relevant inputs for the rules of the ripple carry adder is two. This means that a three-input QCA majority gate can be used, if one of the three inputs is fixed to 0 (see Figure 6). For rules with a higher number of input values an appropriate majority AND gate has to be used. A lot of solutions for QCA gates with more than three inputs can be found in literature, e.g., in [13] an optimized solution for a five-input majority gates is presented. If the value in the LHS is 0, then an inverter has to be included in the path of QCA cells that leads the input value corresponding to the LHS entry to the input of majority gates. The output of the majority

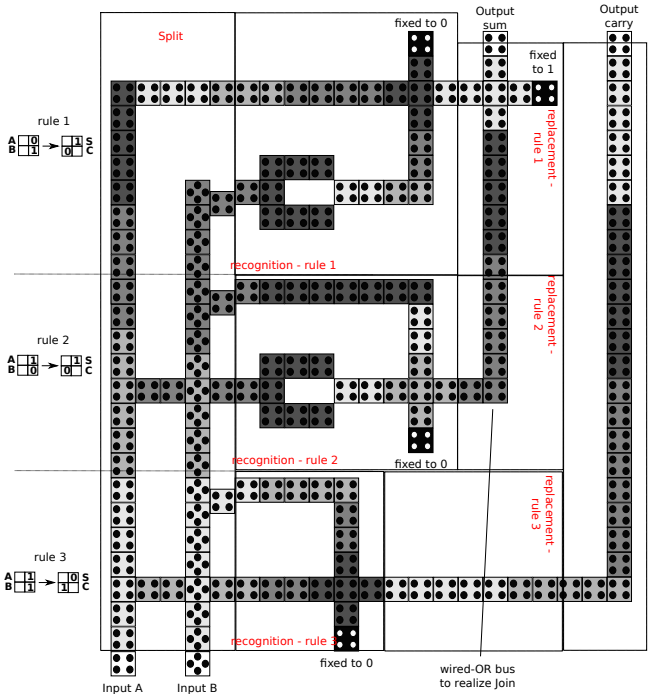


Fig. 6. Result of the mapping process of SSL logic onto QCA logic for the ripple carry adder. To synchronize the changing of QCA cell states' four different clock zones have to be defined. In the figure these four clock zones are marked with a different gray level in QCA cells. Electrons are not shown in this figure.

cell is exactly 1, if the LHS pattern is detected. In this sense the majority gate works analogous to the photo detector NOR device used in SSL (see Figure 1).

The following explanations correspond to the replacement stage in SSL. If the output of the majority gate is 0, then 0's are produced for all 1 values in the RHS of the corresponding SSL rule. If the output of the majority gate is 1, i.e., the LHS pattern was detected, a 1 is produced for each value 1 given in the RHS by an additional majority gate operating as an OR gate (see Figure 6, majority gate in replacement part with one input fixed to 1). If the RHS is 0 no majority gate is necessary since we will work with wired-OR buses in the join stage that carry already the 0 value, which is possibly inserted by the replacement stage located at the lowest position in the wired-OR bus (see rule 2 in Figure 6).

C. Mapping the join stage to QCA cells

As just mentioned the principle of the join stage in SSL is the realization of an optical wired-OR. The same idea was pursued for the equivalent QCA logic. If a 1 has to be inserted in the wire due to a relevant 1 from an RHS, which is output from a replacement stage, this can be done with 3-input majority gates with one input fixed to 1 (see Figure 6, wired-OR bus in block rules 1). In this case, a 1 is only injected in the wire if the output of the attached recognition stage to the wire is 1 or the third input coming from the wire is already 1. This functioning corresponds exactly to a wired-OR bus. A logical

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

1 is injected if an LHS was found and a 1 in the corresponding output of the RHS is given. If the detected rule requires a 0 in the RHS this is automatically given by the fixed injection of a 0 in the QCA wire by the lowest replacement stage attached to the wired-OR bus. If the rules are not in conflict, i.e., only the LHS of exactly one rule was found, then only the output of the RHS belonging to the LHS is injected. This can be either a 0 or a 1. If it is a 0 an explicit injection is not necessary. This causes that rules, which have only 0's in the RHS, must not be implemented if it is secure that exactly one of the rules is always valid. This is given for the case of the ripple carry adder. Therefore, the rule corresponding to (A,B)=(0,0) has not to be implemented with corresponding QCA cells. If it is the only rule that holds, then the corresponding 0's in the output are already on the QCA wires. Utilizing this a priori knowledge the requirements to QCA hardware can be optimized during the synthesis process from SSL logic to QCA logic.

VI. The memristor

In 1971, [14] stated that there must exist a fourth basic circuit element, that he called the memristor. He derived the word memristor from memory + resistor = memristor, a two-connectors circuit element, that works as an adjustable resistor. The resistance value is "memorized" by memristors without the need of energy.

Though this element was predicted to exist in 1971, several years passed by until an operational memristor was successfully built for the first time in the HP laboratories [4]. HP is the current leader in building nanoscale 3D layered memristors [15] and is still ongoing in research about this basic circuit element. At the moment of this writing, literature states that memristors can be built in size of down to $3nm^2$, which theoretically gives very promising densities of non-volatile memory. Industry is currently working on the manufacturing of memristor-based memory chips as drop-in replacement for flash memory.

Not only the resistance of a memristor can be used to control the flow of a current, the resistance value can be used to store information. When certain information is mapped to a certain resistance value, information can be stored in memristors, by setting a memristor to this resistance. The memristor then keeps this resistance value, i.e., the information, until is is later "read-out". Reading out means to find out the current resistance value, to which the memristor was previously set. Due to the mapping, the stored information can be obtained from the resistance of a memristor.

E.g., mapping 0 to the lowest possible resistance and 1 to the highest possible resistance, binary information, as known from current computers, can be stored. As it is possible to set a memristor to an arbitrary resistance value, not only binary information can be stored, but also multi-value information or encodings. E.g., when the range of resistance from lowest to highest resistance is divided into 10 specific resistance values, the values 0 to 9 could be stored.

Another area in which memristors can be used, are programmable nano-scaled crossbars. Crossings of nanowires, e.g., assembled from carbon nanotubes, can be connected or disconnected in a switchable manner by the layered 3D memristors from HP [15] with connectors on the bottom and at the top of each memristor.

Not only switchable connections or non-volatile memory can be build from memristors, but also computing, by building basic gates, is possible. [16] describes how the very basic, in CMOS technology widely used, NAND gate can be copied with a circuit build from three memristors. As for QCA, this also means for memristor-based computers, that computing unit and memory meld together.

VII. Memristor-based circuit simulation

Newly developed circuits are typically simulated before expensive prototypes are produced. This should also apply to future circuits that are based on memristors. Ideally, the use of new technology should be completely transparent for hardware developers. Though, it is possible to build arithmetic circuits, computation logic and memory from memristors in theory, their characteristics affect the design process of the whole system. I.e., they can not transparently replace transistors in existing circuits.

To simulate memristors in certain circuits, we used the SystemC hardware modeling language. We chose this language to analyze simulations of digital systems that make use of memristors and its challenges. SystemC has the advantage to be quick and easy to use and does not require a large toolchain assembled from a variety of different software tools.

A. Abstraction of analogue behavior in digital simulators

As presented in the previous section, we use SystemC to develop simulations of digital systems that are among others build from memristors. In our first step, we use memristors as a register.

Without going too deep into detail, which can be found in other literature (see [17]), we will only explain those characteristics of memristors in this paper, that are relevant to our research. The resistance of a memristor is set by (i) a current flow through the memristor, (ii) the time interval of this current flow and (iii) the polarization of the current. I.e., a higher current and a longer time interval of that current change the resistance in a greater amount in contrast to a low current for a short time interval.

To "read out" the resistance value of a memristor, [17] suggests to apply an alternating-current to the memristor. An alternating-current has the advantage that the resistance of the memristor does not get changed, disregarding a small delta, as the alternating-current, flowing through the memristor changes its resistance value up and down by approximately the same amount. By the help of comparators the current resistance value of a memristor can be obtained.

As we want to work with digital simulators, this analogue behavior has to be abstracted to model a memristor. As mentioned above the write procedure needs a current and time. A current can not be modeled in any way in a digital simulation, as a result we propose to drop it from the model. Though, the time interval in which current has to flow through the memristor can be modeled. We propose to require time

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

intervals in the memristor model from literature and use the clock frequency of the simulation as a reference. We propose that the input value has to be applied to the memristor, i.e., its model, for exactly as many clock cycles as necessary. If the input signal is applied for a shorter or longer amount of time, the memristor should store a lower or higher resistance value, as it would happen in reality.

Another challenge that memristors introduce, is that they should not be "blindly" set into a new state of resistance as this could burn the device if a high current flows through the memristor when it is in a low resistance state. This is especially important for newly produced memristors, as their initial resistance value is almost unpredictable, due to tolerances during manufacturing. For simulated memristors we suggest to set randomly picked resistance values to them during the initialization phase at the beginning of a simulation run. By doing so the unpredictable state of a new memristor can be simulated.

Furthermore, we suggest to use the simulators debug capabilities to display warnings about erroneous behavior to the user, to point out errors as obvious as possible. As the complete field of applications for which memristors can be used are unlikely to be ever known, simulators should not decide whether an access to a memristor at a certain point in time should lead to an error or not.

B. Impact on real circuits

In Section VII, we propose an analogue read-out circuit, to obtain the current resistance value of a memristor. This analogue circuit is hidden, i.e., not visible to the designer, in a high level hardware description language, like VHDL, Verilog or SystemC. When a hardware description is synthesized, these analogue read-out circuits have to be added implicitly to the later real hardware, which, of course, has extra costs of energy consumption of these chips and the required extra space on them.

We propose that hardware development software and analyzation and debugging tools have to be made aware of the extra added read-out circuits, otherwise the analogue circuits have to be added in a by-hand process which is typically erroneous. This gives hardware developers the ability to better understand and analyze memristor-based circuits before first prototype samples of chips are produced.

C. Memristors as memory in four-value logic

Multi-value logic memory is a promising field to build space efficient memory arrays. To demonstrate a possible use case for multi-value, i.e., four-value memory in our case demonstration, we chose the CORDIC [18] algorithm as an example. For this algorithm, a successive multiplicity of additions have to be computed. To obtain high performance, we do not use a binary representation of addends, but a signed digit (SD) logic representation, for fast signed digit adders. The high performance is achieved as no carry bits have to be computed in SD adders. Though the conversion from a number in SD representation to binary representation is expensive, this will not affect the overall performance of the CORDIC implementation significantly, as the repeated additions outweigh the expensive conversion.

TABLE I. BINARY ENCODING OF SD DIGITS

SD	Binary
-1	10
0	00
0	11
1	01

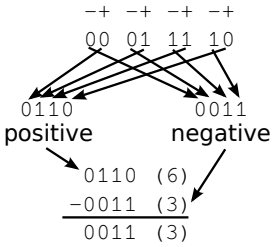


Fig. 7. Conversion from an SD number into a binary or decimal number.

TABLE II. MAPPING OF 4-VALUE LOGIC

resistance	SD	Binary
lowest	-1	10
low	0	00
high	0	11
highest	1	01

In SD logic each digit can take the values -1, 0, and 1. These digits are assembled of a positive and a negative *weight*. To obtain the decimal value of a SD number, the negative weight has to be subtracted from the positive one. As a result, a value of 0 can be composed from 0 negative and 0 positve weight, or from 1 negative and 1 positive weight. I.e., in SD logic exactly 4 values or states are necessary to store an SD digit. To encode one SD digit in binary, two bits have to be used to encode its value. Table I depicts the binary encoding of SD digits. The higher bit stores the *negative weight*, the lower bit stores the *positive weight*. To convert an SD number into binary or decimal representation, the negative weight has to be subtracted from the positive weight of the whole number as shown in Figure 7.

Memristors can be set to an arbitrary resistance value. We take advantage of this capability and define four resistance values for our case demonstration as follows:

- lowest resistance
- low resistance
- high resistance
- highest resistance

We map these for states to the SD values -1, 0 and 1. Of course the difference in resistance between each pair of encodings should be large enough to avoid faulty read-out results, that would lead to a misinterpretation. The four mappings are necessary, because there are two valid representations of the value 0 in SD logic, which can be expressed in a binary encoding as 00 and 11. As a result a possible mapping is shown in Table II.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

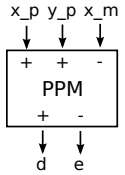


Fig. 8. PPM cell with its three input and two outputs connectors.

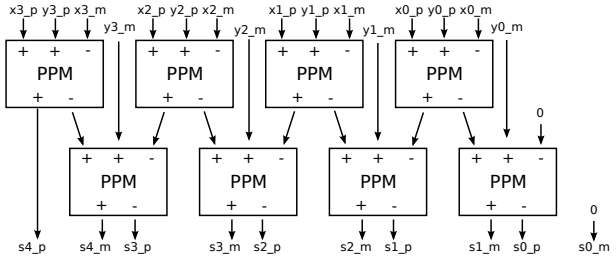


Fig. 9. 4 digit SD adder built from PPM cells.

D. SystemC memristor implementation

In this section, we describe our implementation in SystemC of the necessary modules (circuit components) to evaluate the use of memristors as memory or registers in fast SD adders. This adder should later be used in a simulation for a DSP that implements the CORDIC algorithm to compute trigonometric functions.

For our case demonstration, we implemented the SD adder as presented in [19]. This SD adder is assembled from so-called Plus-Plus-Minus (PPM) cells and its advantage is, that addends can be SD numbers, but also regular binary numbers with 0 as negative weight in all digits. A PPM cell is depicted in Figure 8. Its three inputs are from left to right the positive weight of the addend x (x_p), the positive weight of the addend y (y_p) and the negative weight of the addend x (x_m). The outputs are a positive weight d and a negative weight e, whereas d is computed by

$$d = x\_p \cdot y\_p \vee x\_p \cdot \bar{x}\_n \vee y\_p \cdot \bar{x}\_n$$

and e is

$$e = x\_p \oplus y\_p \oplus x\_n$$

(see also [19]).

The whole SD adder is assembled from two PPM cells per digit, e.g., for a four digit SD adder eight PPM cells are necessary. To perform the SD adding, the PPM cells have to be connected as shown in Figure 9 to build a 4 digit input SD adder (see also [19]).

Since the memristor is an analogue circuit element, we can not implement it in classic SystemC directly, and, as already mentioned in Section VII-A, it is not of interest to our research to obtain an accurate analogue simulation model, but a digital equivalent.

We implemented the memristor as a SystemC module. Its interface has three input signals and one output signal, whereas the input signals are a clock signal *clock*, a boolean input signal *w_en* to signal a "write" access and an input signal *in* of type uint8_t that characterizes the resistance value to be stored in the memristor. Internally the memristor module stores its state in a private member *state* of type uint8_t. Another private member is a boolean lock variable *lock*, its use is described later in this section.

At first glance the clock input signal might seem unnecessary as the memristor is not a naturally clocked circuit element, but as mentioned in Section VII-A, an input value should only be stored correctly if the input that is to be stored is constantly available for a certain time interval. Otherwise, the memristor should not store the resistance value or a different one from what was set at the input. The clock is used to trigger an incremental counter on each rising clock edge. It allows the memristor module to observe if the input signal is available unchanged for the correct time interval, that corresponds to the value to be stored, i.e., for the correct number of clock cycles.

Our SystemC model of a memristor implements an endless loop in a SystemC thread, which immediately blocks and gets woken up every rising clock edge. The loop checks if the write signal is set to *true* and if the lock is free. If that is the case the lock is taken and the loop blocks for 5ns with the SystemC *wait()* instruction. After waiting for 5ns the input value is copied into the internal state variable *state* and the lock is released so that the resistance value of the memristor can be changed again. As we propose in Section VII-A, a warning is displayed during the SystemC simulation, if the input is changed while waiting for 5ns or if the write signal gets set to *false*.

For a sole digital system, it is sufficient to use type bool, sc_bit or sc_logic for the input signal of the memristor. Though, we want our model to be able to store four-value logic but also very fast simulation for very large memristor-based systems in the future. The SystemC documentation states, that users should use C++ data types, where possible if one wants to achieve high speed simulations. For that reason, we chose uint8_t to store more than only binary information and to use a C++ primitive data type for good simulation performance in the future.

The choice of the input data type affects the way, how to use our memristor model in circuit models. A typical hardware description use boolean or sc_logic types that store and transport binary information. To attach our model to such a circuit a transformation between digital logic and the memristor inputs and outputs has to be performed. For this purpose we implemented two connector modules to fulfill this requirement for four-value logic: A conversion module that takes two-wire binary input and is to be connected to the memristor input, and a module that is connected to a memristor output and transforms it to two-wire binary value. The transformation modules have no memristor as a private member. We expect the user to connect memristor and transformation modules by herself, which leaves the option to use single transformation modules for a cluster of memristors and place multiplexers between the input and output ports of memristor modules. This causes an extra effort to the user, but we considered it more worth to save redundant conversion modules in situations when they could be reused to access a cluster of memristors.

In our model, the conversion modules model the analogue

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

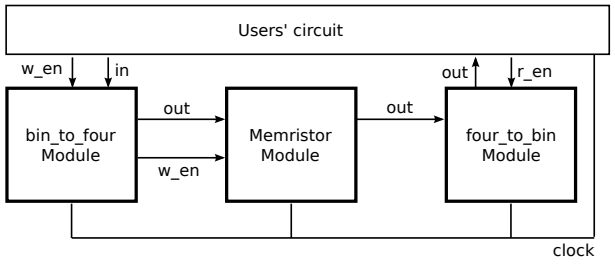


Fig. 10. Schematic depiction how memristor and conversion modules are to be connected.

read-out and write-in circuits as there have to be in a real chip that is based on memristors (see VII-A). Figure 10 depicts schematically the architecture, how to use our memristor model and the transformation modules in users' hardware descriptions. Data flow directions are pointed out with arrows. From the user's circuit, binary input *in*, data is stored in the memristor, when the write enable signal *w_en* is set to high. Then the data is transformed and output via the left *out* signal into four-value logic and stored in the memristor and the write enable input *w_en* of the memristor module is set to high to signal the write procedure. To read the stored value from the memristor, the user has to set the read enable input signal *r_en* to high and the *four_to_bin* module will output the value stored in the memristor binary encoded on the very right output signal *out*.

Above we presented all necessary modules to build the fast SD adder with a four-value register for successive additions for the CORDIC algorithm. The prototype model we have implemented in SystemC is depicted in Figure 11. Both addends x and y can be in SD representation and in binary representation with negative weights set to 0. The multiplexer *MUX* selects between the second addend or using a previously calculated sum *sum*, stored in the memristor register. The SD sum is computed by the PPM cells and available at the output *sum*. When the sum is valid at the output of the SD adder, it is also stored in the memristor register and can be reused as addend at a later time.

E. Memristor simulation results

We present our results of the memristor and the fast SD adder simulation in this section. By the help of a test bench that we implemented in SystemC, we verified our memristor SystemC model to be correct. To do so, we wrote a test bench that attempted to store all possible values in the memristor module, read it back and compared it to the previously stored input. In order to test faulty accesses, we interrupted the input to the memristor module during the write-in phase and verified that the stored value differed from the input data.

Furthermore, we proved the SD adder module to be correct. This implies that we also proved the PPM cell modules to be correct while verifying the complete SD adder. In order to verify the SD adder, we had it perform additions of all possible input permutations to verify that the output sum corresponds to the correct addition.

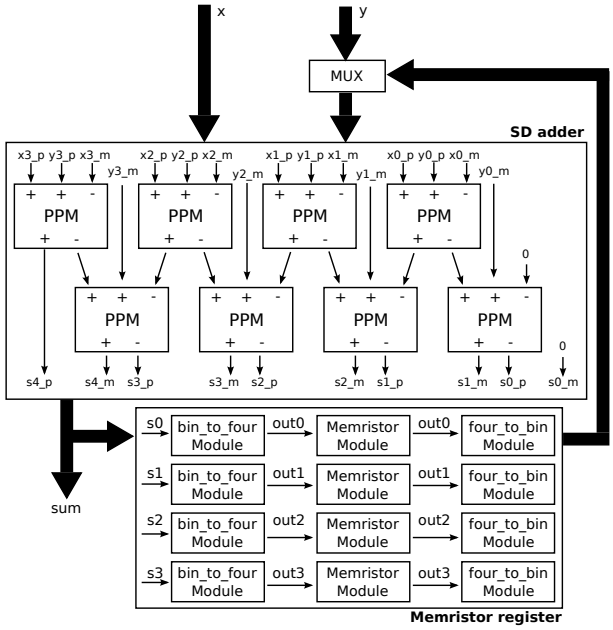


Fig. 11. Architecture of our fast SD adder with memristor register for CORDIC.

Due to the connections between the output ports d and e of the top PPM cells to the input ports x_p and x_n of the bottom PPM cells, a delay of one clock cycle is introduced. This results from the limitation that ports can not be connected directly or like wires, but are always transformed into Flip-Flops that present the input value to its output connector with a delay of one clock cycle.

Concerning the write access performance to the memristor it depends on the simulated clock speed, how many clock cycles are necessary to finish the successful storing of the value. In our simulation, we used the standard clock frequency of SystemC 2.3.0 that is 1GHz, i.e., a period of 1ns. As a result, a write access to the memristor model lead to a delay of five clock cycles in our simulation.

Both transformation modules, as presented in Section VII-D, added a delay of one clock cycle, as information has to pass through one stage of Flip-Flops that are connected to the output ports.

For both data paths through our SD adder with memristor registers, the data throughput is delayed only by Flip-Flops. The worst case were two additions, in which the intermediate sum was stored in the memristors an reused for a second addition. In that case, the overall delay, until the final result was displayed at the output *sum* (see Figure 11), composed as follows: During the first addition 1 clock cycle delay appears due to Flip-Flops in the interconnection between the PPM cells and a 2nd delay until the final computation result is displayed at the sum output ports. During the conversion from SD to four-value logic, a delay appeared in the *bin_to_four* transformation module and another 5 clock cycles delay until data is stored in the memristors. Information was written in parallel to the memristors, so the delay always remains

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

constant at 5 clock cycles, unless the frequency of the clock is not changed. When the data is read out from the memristors, 1 clock cycle delay is added in the *four_to_bin* transformation module. During the second addition, when an addend was added to the intermediate sum, the delay was limited to the same 2 clock cycles, exactly as during the previous addition. All in all, the clock cycles sumed up to 11 clock cycles delay in our worst case scenario.

Since we used the standard clock frequency of SystemC 2.3.0, 1ns was equivalent to one clock cycle. This allowed us to use the SystemC built-in function *sc_time_stamp()* to retrieve the delayed clock cycles.

VIII. Comparison of QCA and memristor technology

While both technologies are two very differing approaches to overcome the CMOS limitations, we identified overlapping similarities in both technologies. In this section, we will present our findings.

Though the memristor works by the transport of electrons and QCA relies on the propagation of Coloumb force impulses between electrons, both devices are stateful. While the memristor can theoretically be put into an infinite number of different states, i.e., it can be set to an arbitrary resistance, it is important to put it into a well-chosen limited number of states, e.g., in a digital system to low resistance for logic 0 and high resistance for logic 1. By its nature, the memristor will keep its resistance, to which it was previously set, without the need of energy. It will remain in this state until it is changed to another resistance.

Regarding QCA, which needs a clocked electric field as a clock signal [20], QCA cells keep their state, i.e., the arrangement of the electrons in the potential wells, without the need of energy. This is a common ground between QCA and memristor technology. Regarding the state-of-the-art efforts for current CMOS based computers, to put them into an energy-saving "sleep" state, we identify the possibility of future QCA- or memristor-based computers, this capability is automatically available by the underlying technology. QCA cells and memristors remain in their state without the need of energy. I.e., a computer based only on QCA or memristor technology is put into an energy saving state as soon as the power supply is disconnected. As soon as it is reconnected to its power supply it will continue computation at the very same point when it was disconnected from power.

Simulations of QCA systems predict very high clock frequencies, up to THz scale, while memristors limit data throughput by the necessary time interval to put the memristor in a specific resistance. This is an advantage of QCA over memristors. On the other hand, the memristor can be used in multi-value logic environments, whereas QCA can only compute and store binary information. In our SD adder model, the memristor is used to store four-value logic information, which is an advantage over a binary memmory. The four-value logic allows to reduce the number of memory devices to the half, in contrast to binary memory. In larger systems than our model, four-value memristor memory can improve the space efficiency on a chip enormous.

IX. Conclusion

We presented a generic design procedure for mapping digital optical computing circuits based on SSL onto nanocomputing QCA circuits. This will form both the base for future design tools for compact, regular build-up QCA circuits and supports the direct mapping of optical computing circuits to QCA technology. For example, we intend to map an integer arithmetic unit based on SSL, designed by us [21], onto a complete QCA integer unit. In addition, we have to verify the schematically shown QCA circuit of Figure 6 by simulation with the QCADesigner tool [2], the standard for simulating QCA layouts. Furthermore, the insertion of an exact clocking scheme for the QCA cells has to be considered in the synthesis procedure. Nevertheless, the basic step for an automatic synthesis of SSL arithmetic circuits to QCA layouts is established.

Furthermore, we presented our developed SystemC model of a memristor and the characteristics of the model for a digital circuit simulation, which we derived from its analogue behavior. We have shown that the memristor can work as a four-value logic memory in a fast SD adder circuit, which is our prototype of a building-block for a future implementation of a CORDIC implementation. Though the prototype needs some further improvement, we also demonstrated that it is possible to model this typical analogue device for a digital simulation and design process. For our prototype we also modeled in SystemC the analogue write-in and read-out circuits for a digital simulator. The prototype was completely verified and with some improvement will form a building-block for the implementation of memristor-based arithmetic circuits.

Our findings pointed out, that development and synthesis software tools for memristor-based circuits have to be aware of the analogue extra circuitry. We proposed that the hardware designer must be given the ability on a high level hardware description, to influence and optimize the utilization and reusability of underlying analogue circuitry, in order to gain maximum space efficiency on a chip.

We compared both nanotechnologies and found challenging differences in the requirements to automated design tools. If memristor-based arithmetic units become state-of-the-art in the mid-term future and hardware design tools get adopted to this technology, our findings point that further research on design tools is necessary to make them reusable for the long-term QCA technology.

Despite the differences, we found a promising common ground among both technologies for energy efficient future computers. In contrast to current CMOS-based computers, both technologies keep remain in their current state without the need of energy. We propose to leverage this natural property for mid- and long-term future computers to save energy. We suggest to cut off power supply during idle states of these devices as systems built from QCA cells and memristors will continue their computations exactly when the were powered off.

We identified the advantage of memristors over QCA technology, to be suitable for multi-level logic environments. With our model of a fast SD adder with a memristor-based intermediate register, we demonstrated the advantage of space efficiency of a four-value logic memory, that is implemented

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

with memristors. Our model needs only half of the memory elements, compared to a binary memory.

REFERENCES

[1] D. Fey and B. Kleinert, "Using Symbolic Substitution Logic as an Automated Design Procedure for QCA Arithmetic Circuits," in *FUTURE COMPUTING 2012, The Fourth International Conference on Future Computational Technologies and Applications*, 2012, pp. 94–97.

[2] K. Walus, T. J. Dysart, G. A. Jullien, and R. A. Budiman, "QCADesigner: A rapid design and simulation tool for quantum-dot cellular automata," *Nanotechnology, IEEE Transactions on*, vol. 3, no. 1, pp. 26–31, 2004.

[3] K.-H. Brenner, A. Huang, and N. Streibl, "Digital optical computing with symbolic substitution," *Appl. Opt.*, vol. 25, no. 18, pp. 3054–3060, Sep 1986. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-25-18-3054

[4] R. Williams, "How we found the missing memristor," *Spectrum, IEEE*, vol. 45, no. 12, pp. 28–35, 2008.

[5] K. H. Brenner, W. Eckert, and C. Passon, "Demonstration of an optical pipeline adder and design concepts for its microintegration," *Optics & Laser Technology*, vol. 26, no. 4, pp. 229–237, 1994.

[6] C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, "Quantum cellular automata," *Nanotechnology*, vol. 4, no. 1, p. 49, 1993. [Online]. Available: http://stacks.iop.org/0957-4484/4/i=1/a=004

[7] V. A. Mardiris and I. G. Karafyllidis, "Design and simulation of modular 2n to 1 quantum-dot cellular automata (QCA) multiplexers," *International Journal of Circuit Theory and Applications*, vol. 38, no. 8, pp. 771–785, 2010. [Online]. Available: http://dx.doi.org/10.1002/cta.595

[8] F. Bruschi, F. Perini, V. Rana, and D. Sciuto, "An efficient Quantum-Dot Cellular Automata adder," in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011*, 2011, pp. 1–4.

[9] R. Zhang, K. Walus, W. Wang, and G. A. Jullien, "A method of majority logic reduction for quantum cellular automata," *Nanotechnology, IEEE Transactions on*, vol. 3, no. 4, pp. 443–450, 2004.

[10] A. Louri, "Parallel implementation of optical symbolic substitution logic using shadow-casting and polarization," *Applied optics*, vol. 30, no. 5, pp. 540–548, 1991.

[11] Y. Ichioka and J. Tanida, "Optical parallel logic gates using a shadow-casting system for optical digital computing," *Proceedings of the IEEE*, vol. 72, no. 7, pp. 787–801, 1984.

[12] I. Hänninen, "Computer Arithmetic on Quantum-dot Cellular Automata Technology," Ph.D. dissertation, Tampare University of Technology, http://dspace.cc.tut.fi/dpub/handle/123456789/6337?show=full, 2009.

[13] R. Akeela and M. D. Wagh, "A Five-input Majority Gate in Quantum-dot Cellular Automata," 2011.

[14] L. Chua, "Memristor-the missing circuit element," *Circuit Theory, IEEE Transactions on*, vol. 18, no. 5, pp. 507–519, 1971.

[15] G. S. Snider, "Self-organized computation with unreliable, memristive nanodevices," *Nanotechnology*, vol. 18, no. 36, p. 365202, 2007.

[16] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, and R. S. Williams, "'Memristive'switches enable 'stateful'logic operations via material implication," *Nature*, vol. 464, no. 7290, pp. 873–876, 2010.

[17] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, 2008.

[18] J. E. Volder, "The CORDIC trigonometric computing technique," *Electronic Computers, IRE Transactions on*, no. 3, pp. 330–334, 1959.

[19] B. Kasche, "Entwurf eines optoelektronischen Rechenwerkes," Ph.D. dissertation, 1999.

[20] S. E. Frost, "Memory Architecture for Quantom-dot Cellular Automata," Ph.D. dissertation, University of Notre Dame, 2005.

[21] D. Fey and K. H. Brenner, "Digital optical arithmetic based on systolic arrays and symbolic substitution logic," *Opt. Comput*, vol. 1, pp. 153–167, 1990.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Developing an ESL Design Flow and Integrating Design Space Exploration for Embedded Systems

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

***Abstract*—This paper introduces a systematic development of design flows for embedded systems. The idea of an executable design flow provides a basis for the design automation starting at system level. The aim is to develop, manage and optimize design flows more efficiently. A seamless integration of design space exploration into a design flow is presented coping with the conflicting design goals of embedded systems at electronic system level. It is further shown that an abstract design flow model simplifies a derivation of domain-specific design flows. A novel programming language is introduced allowing for the development of design flows in a visual and textual manner. A case study of the heterogeneous multicluster architecture demonstrates a usage of the design approach and automation. A systematic dimensioning of the multicluster architecture, in terms of the necessary computation resources, is presented in detail. The case study addresses various design problems of future embedded systems at electronic system level. Finally, this paper presents design flow development and design space exploration for embedded systems being systematically, fully integrated, and automated in order to improve a system level design.**

Keywords-electronic design automation, electronic system level, design flow, design space exploration

I INTRODUCTION

It is commonly accepted by all major semiconductor roadmaps that only by raising the design process to higher levels of abstraction will designers be able to cope with the existing design challenges. This leads to an *electronic system level* (ESL) design flow. The term *system level* refers to a use of abstract system functions in order to improve comprehension about a system. Design space exploration (DSE) needs to be integrated in order to trade-off between the conflicting goals of ESL design, such as performance, power consumption, and area [1]. ESL design aims at a seamless transformation of a system specification into a hardware (HW)/ software (SW) implementation [2]. Hence, electronic design automation (EDA) requires a system specification, which is executable in a computer simulation. An *executable specification* is a simulation model of the intended system functions, also called a virtual prototype [3][4].

Today's ESL *design flows*, from now on shortened to flows, are typically based on a specify-explore-refine (SER) methodology [5]. Such flows include a sequence of design steps, from now on shortened to steps, successively refining a system model. Each step solves a design problem, such as application mapping. Moreover, a *specification model* defines the starting point representing the targeted application characteristics and requirements. "Specification model is used by application designers to prove that their algorithms work on a given system platform" [2]. Then, each exploration step creates a design decision continuously increasing the accuracy of the system model. Afterwards, the refined model is passed to the next exploration step. Recently developed EDA environments for ESL design, as proposed in the MULTICUBE project [6] and NASA framework [7], turn away from ad-hoc software infrastructure. The generic EDA systems provide modularization and well-defined interfaces. Despite these advancements, the problem of a large number of possible flow sequences has not been addressed yet.

Since future embedded systems will have an increasing design complexity, the number of steps in a flow is further rising. For example, an optimization of the resource management will require additional steps [8]. Furthermore, the huge design space will draw more attention to an ESL design at an early design stage in order to avoid time-consuming low-level simulations. A systematic methodology to develop, manage and optimize flows promises for a significantly improved design process. In this paper, the approach is denoted as the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

design of design flow (DODF). Similar methodologies have been developed in other scientific fields, such as physics [9], mechanical engineering [10], and software engineering [11]. Nevertheless, their degree of automation is limited and the main contribution of this paper is to address this drawback. The aim is to provide an EDA environment increasing the user's productivity.

The remainder of this paper is organized as follows. The related work and design approach are presented in the Sections II and III. In Section IV, the authors introduce the principle of an executable flow. The section also focuses on an explanation of the DODF approach. Then, the introduced concepts are exemplified via a functional exploration of a finite impulse response (FIR) filter. In Section V, the modeling of flows is explained. The idea of abstracting the flows and a corresponding derivation of a domain-specific flow are further introduced. In addition, DSE techniques, applicable to a step and flow, are covered. In Section VI, a visual and textual design flow language (DFL) are presented allowing to develop, manage, and optimize a flow. An according tool flow is introduced afterwards. Finally, Section VII applies the previously developed models and automation tools for an ESL design of the heterogeneous multicluster architecture [12]. The several flows are arranged in a sequence of flows. The flow for the multicluster dimensioning problem will be described in detail.

II RELATED WORK

The related work reviews representative design and specification languages. Moreover, state-of-the-art DSE environments are covered. Then, related studies on meta-modeling are presented. Finally, the use of scripting languages is discussed in the context of EDA.

Specification Languages and DSE Environments

There is a variety of graphical and textual specification languages and frameworks. They can be used to realize ESL design by following a given design methodology. Nevertheless, this is done in a less formal and less generic manner compared to our systematic development of flows. Hence, the reuse and interoperability across tools, designers, and domains are limited. An example is the specification and description language (SDL) [13] allowing for formal and graphical system specification and their implementation. In [14], HW/SW co-design of embedded systems is presented using SDL-based application descriptions and HW-emulating virtual prototypes. Moreover, SystemC [15] and SpecC [16] are system-level design languages (SLDL), which model executable specifications of HW/SW systems at multiple levels of abstraction. These simulation models support SW development. For example, SystemCoDesigner [17] enables an automatic DSE and rapid prototyping of behavioral SystemC models. In [18], a comprehensive design framework for heterogeneous MPSoC is presented. Based on the SpecC language and methodology, it supports an automatic model generation, estimation, and verification enabling rapid DSE. Using an abstract specification of the desired system as starting point, pin-and cycle-accurate system models are automatically created through an iterative refinement at various levels of abstraction. Another example is the specification in a synchronous language, e.g., via Matlab/Simulink. Opposed to that, Ptolemy [19] supports various models of computation to realize executable specifications including synchronous concurrency models. For both examples, DSE has to be realized through a dedicated implementation.

As mentioned in Section I, the MultiCube project [6] and the NASA framework [7] provide a generic infrastructure for ESL design including DSE. Nevertheless, the works do not provide a systematic development of flows and an according design flow language. Hence, they are limited to proprietary flows.

Meta-modeling

Our paper differs to existing work since it is the first using meta-modeling for developing a design flow for embedded systems. Meta-modeling has also been studied to transform from the unified markup language (UML) to SystemC at the meta-model level [20]. This guarantees reuse of models and unifies a definition of the transformation rules. In [21], meta-modeling enables heterogeneous models of computations during modeling. In [22], meta-modeling is used to improve the model semantics and to enable type-checking and inference-based facilities.

Electronic Design Automation

Principally, a general-purpose programming language, such as C/C++, Java, C#, etc., can define a flow via data and control structures. There are different implementation options for a flow description avoiding a unique representation of a flow. Moreover, compilation times prevent from a seamless programming. Hence, a scripting language, tailored to that task, would be rather suited. For example, the major EDA tool vendors Synopsys [23], Cadence [24], and Mentor Graphics [25] provide a scripting language interface for design automation. Therein, the EDA functions are accessible via the language commands in order to build custom flows. The first example is the tool command language (Tcl) [26]. The scripting language has been integrated in the EDA

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

tools of Synopsys and Mentor Graphics. Tcl is available as open source project without licensing. Another design automation language represents SKILL [27]. SKILL, also a scripting language, has been derived from Lisp, and is integrated in the EDA tools of Cadence. In addition, Perl, Ruby, and Python are used as EDA scripting languages, as presented in [28]. A major drawback of the languages is, they leave it to the designer how to develop, manage and optimize a flow. Hence, the realization of a systematic structure, parallelization, and debugging of flows can differ for each language and designer. This makes the understanding, maintenance and reuse of the flow descriptions a challenging task. This paper addressing the issue by supporting a systematic development of flows via DFL. Furthermore, DSE is directly considered in the language design and implementation, which is not the case for the existing EDA scripting languages.

III DESIGN APPROACH

This section provides an overview of the design approach. It includes two conceptual levels and one instance level related to the terms *method*, *methodology*, and *model*. This is illustrated in the Figure 1. The basic idea is that models and methods are used by a methodology. The classification and relationships will be explained in the following. A composition refers to an element, which is part of another element. Instantiation means that an element is derived from another element. Moreover, the term *meta* is used in order to describe an abstraction of a subject. An example is the meta-data, which means data about data.

The *meta-methodology* defines a methodology realizing another methodology. In Section B, a meta-methodology for the development of flows, also considered as DODF, is introduced. Hence, a flow represents a *methodology*, composed of steps, in order to build the intended design. A *view* allows for a partitioning of a flow resulting in a subset of the steps. Furthermore, a *step* solves a design problem via a method or simulation model. The step consumes inputs and produces outputs. An *input* can be an executable file, configuration, parameter, or constraint. A method or simulation model are compiled into an executable file or callable library. Moreover, an *output* will be a configuration, which is produced when the step has been finished. Each output needs to be validated via a subsequent step including a simulation model or evaluation method. In addition, a control loop between both steps will allow for several design iterations until an output conforms to the pre-defined constraints.

The meta-modeling describes the modeling of the modeling languages. This includes an abstract syntax and the semantics. For example, a *meta-model* enables heterogeneous models of computations in the ESL design, as presented in [29]. In this paper, a meta-model of a flow is introduced in Section A. The intension is to avoid a discussion about the best definition of the term *model*. The considered example is a suitable definition, found in Wikipedia [30]: "A model is a pattern, plan, representation (especially in miniature), or description designed to show the main object or workings of an object, system, or concept.". A flow model is derived from the meta-model. It defines a set of steps and views in order to build a flow. The λ-chart [8], described in Section C, represents a *flow model* following the meta-model. Meta-models can also be defined for the application and architecture models further being implemented in a simulation model and executable specification, respectively. The *application model* represents the functions and the data exchange between the functions of a target application. Moreover, an *architecture model* describes the structure and functions of the intended system, such as the computation architecture, interconnect topology, management infrastructure, communication protocols, etc. Referring to Figure 1, an application and architecture model for future embedded systems are introduced in the Section VII.

A *meta-method* is a method to analyze another method. For example, *meta-optimization* is an optimization method to tune another optimization method. In [31], a genetic programming technique has been used for the meta-optimization in order to fine-tune compiler heuristics. In Section VII, the author applies meta-optimization via an exhaustive search in the *Parameter Tuning* flow in order to find suitable input parameters of a genetic algorithm (GA). Referring to Figure 1, the *method* denotes a technique for solving an ESL design problem. Optimization and estimation methods are used in the case study presented in Section VII.

IV ESL DESIGN FLOW

Early EDA flows were dominated by capturing and simulating incomplete specifications. Later, the logic level and register-transfer level (RTL) synthesis allowed to describe a design only from its behavior and structural representations. However, a system gap between SW and HW design exists since SW designers still provide HW designers with incomplete specifications. An executable specification, such as implemented via C++, SystemC [15], LabVIEW [32], Simulink [33], Esterel [34], Lustre [35], and Rhapsody [36], closed the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

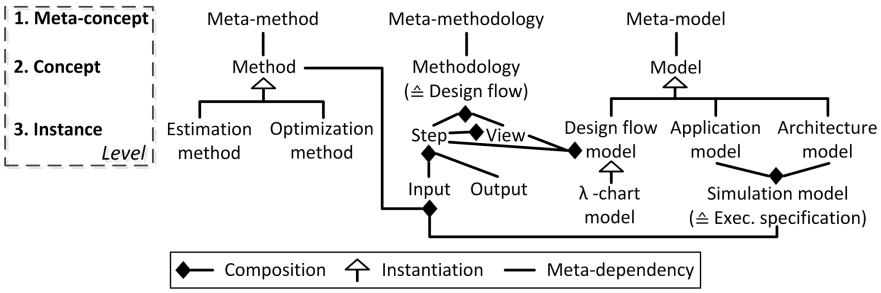


Figure 1. Overview of the design approach.

system gap by describing the system functionality [37]. An ESL flow copes with the design complexity of current multi-processor system-on-chips (MPSoCs). It is expected that the complexity of future many-core SoCs with thousands of cores will further increase the design space [38]. An increasing number of components and their interactions increases the complexity of implementing a many-core SoC flow. The result is a larger number of steps and the inputs/outputs consumed and produced by the steps. In addition, the control structure of a flow will become more complicated. For example, a step can be dependent on multiple steps. Moreover, the variation of multiple parameters/constraints may require nested looping and feedback loops. This section addresses the complexity problem by introducing an executable flow and the DODF approach. The result is a unified methodology to develop, manage, and optimize flows.

A. *Executable Design Flow*

In [1], the authors presented the concept of an integration of DSE into a system-level specification. From that, the idea of an executable flow [39] has been derived. An executable flow denotes a program solving certain design problems and being automatically interpretable by a machine. In an executable flow, methods and simulation models, assigned to steps, are called in the same way instructions of a computer program are called by an interpreter. Predefined methods and models for the steps, e.g., accessible via C++ libraries, would further improve the quality, time and costs of a design. In an executable flow, inputs and outputs are consumed and produced by the steps. The input parameters and constraints control an execution of the steps in a flow. Moreover, an output could comprise a configuration of a refined system model. Since several input values are most likely possible, it results in a huge input or design space of an executable flow. An optimization of the input combinations of each step aims at an adequate step result. Nevertheless, an optimum, comprising all step inputs, is most likely impossible due to the huge design space. This implies several local optima and according design tradeoffs. Moreover, a read access to inputs of a flow will allow for a detection of interfering, inadequate or missing inputs. A further goal is to execute as much as possible steps in parallel. This can be realized for the inputs of a step or by executing independent steps of a flow in parallel. A simple executable flow is illustrated in Figure 2. The flow includes two steps realizing the methods of *Dimensioning* and *Mapping*. First, the dimensioning, implemented, e.g., via an estimation method, extracts an HW architecture from the input configurations of the HW unit options and application. Then, simulation results can be obtained from the mapping of the application onto the HW architecture, as done in the mapping step. Referring to Figure 2, an executable specification implements the system functions necessary to evaluate the system performance.

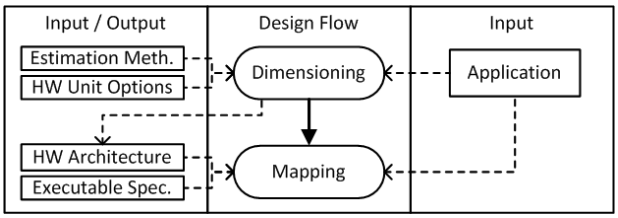


Figure 2. An example of an executable design flow.

B. *Design of Design Flow*

The structure of an executable flow and a methodology for developing flows are incorporated into the DODF approach [39]. The concepts and realizations of DODF are summarized in a hierarchical manner, as seen in Figure 3. The figure shows several members assigned to different hierarchical levels. By moving from the outer part to the inner part of the figure, the concepts are transformed into concrete realizations. The Section C includes an example of a digital filter design illustrating an executable flow and the DODF approach.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

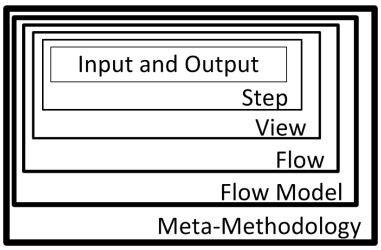


Figure 3. Hierarchy of concepts and realizations in the design of design flow.

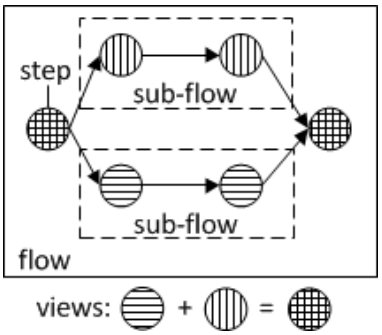


Figure 4. A design flow composed of sub-flows and steps filtered via the views.

First of all, the meta-methodology defines a methodology to create flows. Referring to Section III, the prefix "meta" is used since a methodology is considered as flow. The meta-methodology includes different stages in order to correctly determine and arrange the members defined in the DODF hierarchy. For example, once the steps, their inputs, and their outputs are detected, the steps need to be combined to a flow in order to realize the design goal. In the DODF hierarchy, seen in Figure 3, the flow model is a domain-specific composition of steps and views. The λ-chart [8] is an example of a flow model. As already mentioned before, a model for the modeling of other models is called meta-model. From a meta-model, flow models are created for a specific domain. Then, flows can be derived from the domain-specific flow model. Conceptually, flows are hierarchically composed in order to improve a division of work by assigning a sub-flow or step to specialists in a team. Hence, a flow can be a graph or subgraph with vertexes representing steps. The steps may further represent sub-flows, as indicated in Figure 4. Moreover, each step can belong to a view. Hence, a flow can also include several views, as illustrated in Figure 4. A view represents a level of abstraction in terms of a filter of selected steps. In contrast to a hierarchical division of flows into sub-flow, a view intends extracting a subset of steps assigned to the view. This allows to focus on selective steps and sub-flows. For example, Kogel et al. [40] define the four views: functional view, architects view, programmers view, and verification view. By defining views, a design can be explored from different viewpoints, such as computation topology, interconnect topology, etc. Then, the functionality can be separately analyzed to be explored together in a subsequent design stage. An example is a step assigning the scheduling of computation tasks and load/store tasks to separate views. After the scheduling is explored separately, the results are combined in order to apply the best scheduling technique for all task types.

As mentioned before, a flow is a combination of steps refining a specification model into a targeted system model. Each step uses inputs to apply a method or simulation model, which are compiled into an executable file. An input parameter relates to a description of the structure, behavior, and physical realization of a component or system. Parameters, configuring a design method, are also covered. Furthermore, an input constraint is a restriction of a component or system, such as latency, power consumption, or chip area. Then, the output of a step serves as input for the subsequent step.

As explained before, a flow is derived from a flow model using the meta-methodology and procedure, respectively, illustrated in Figure 5. The idea is to systematically determine, assign, and order sub-flows and the further members of the presented DODF hierarchy, seen in Figure 3. Moreover, an executable flow is built through an algorithmic ordering of the sub-flows and steps. That means, dependencies, loops, branches, etc., realize an execution order of sub-flows and steps in an algorithmic manner. Hence, the ordering of steps realizes a system-level design algorithm based on flow control structures and patterns, respectively, presented in Section B. The meta-methodology glues the members of the DODF hierarchy together in order to systematically follow the DODF approach. Referring to Figure 5, the design goals are first determined and sub-flows are extracted. For example, the design of system components, such as processors, memory, controller, etc., and the design in different levels of abstraction, such as ESL and transaction-level (TL), can be modeled into sub-flows. Then, an algorithmic ordering of the aub-flows needs to be formulated representing the structure of an executable flow. The next stage is to determine the design problems in order to assign each step the corresponding method or simulation model. A method is determined for a step in order to solve a design problem. The simulation mod-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

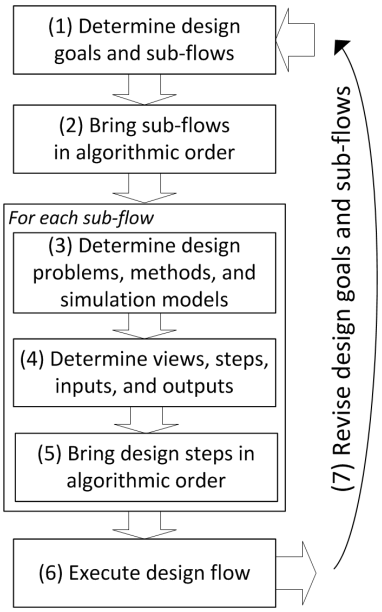


Figure 5. A meta-methodology for the proposed DODF approach.

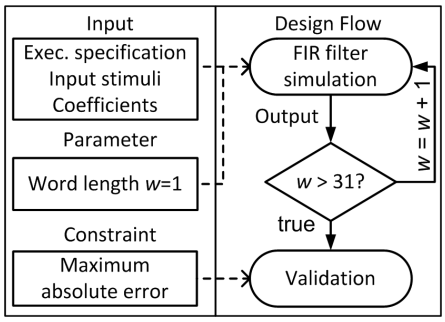


Figure 6. An executable design flow for a functional exploration of the FIR filter.

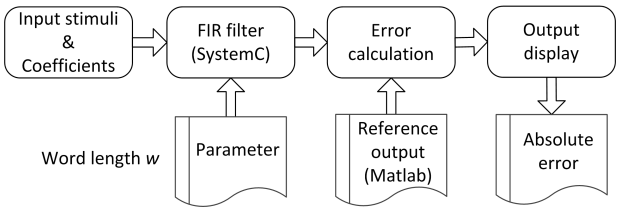


Figure 7. A functional simulation of the FIR filter via SystemC.

els are required for measuring the system performance. Afterwards, each step is assigned a view enabling a horizontal partitioning of the flow. In addition, the inputs and outputs are determined for each step. The next stage finalizes the design of an executable flow by bringing the steps into an algorithmic order. In the end, the flow is executed based on the algorithmic order and variation of the inputs. From the interpretation of the results, the design goals and sub-flows are revised in order to improve the structure and configuration of the flow.

C. A First Example - An FIR Filter

In the following, the flow development is illustrated considering a simple flow. An FIR filter, an ubiquitous digital signal processing algorithm, has been chosen and implemented in a simulation model and executable specification, respectively. Referring to the meta-methodology in Figure 5, the goal and flow are first determined. The goal is to minimize area and power consumption of the memory in an HW implementation of the FIR filter. This is realized via exploring a minimal word length for the bit representation of the FIR filter coefficients. The simple flow is composed of two steps *FIR filter simulation* and *Validation*, as seen in Figure 6. The flow realizes an algorithmic exploration of the FIR filter focusing on the functional view defined in [40]. Hence, the aim is to find the best configuration of the input parameters holding an error constraint. The filter coefficients are provided as real numbers. The word length of each coefficient radix can be varied separately. The step *FIR filter simulation* requires an executable specification, the input stimuli and filter coefficients as inputs. Referring to Figure 6, the step calls an executable specification simulating the FIR function. The simulation performance is evaluated by comparing the output values with a given Matlab reference and calculating a (mean) absolute error, as seen in Figure 7. The output of the step is a mean absolute error representing a degradation compared to the ideal Matlab reference. Referring to Figure 6, the step is executed until the word length w reaches $w = 31$. Then, the *Validation* step finds the best configuration that does not infringe the maximum absolute error constraint. Figure 7 shows the executable specification in terms of a functional simulation of an FIR filter implemented via SystemC [15]. The stimuli represents the input values of the FIR filter. The executable specification is configured with the inputs mentioned before. After the error calculation, a display function returns an absolute error representing the output of the *FIR filter simulation* step.

The following test results are automatically generated by executing the flow. A 16 taps FIR filter with a low-pass characteristic and a cutoff-frequency f_g = 4kHz was configured. In the simulation setup, 1000 uniformly distributed random values are used as input stimuli ranging from 1 to 100. Moreover, the radix of the FIR filter coefficients are jointly varied from 1 to 31 bits.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

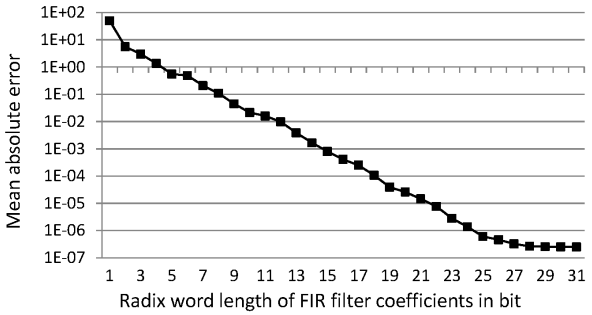


Figure 8. Experimental results of the FIR filter exploration.

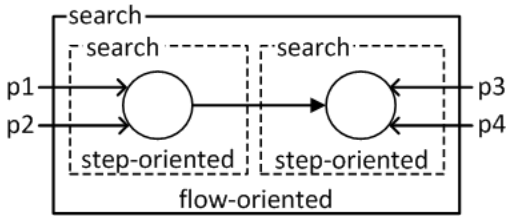


Figure 9. Step-oriented vs. flow-oriented search in an executable flow.

The results are shown in Figure 8. The curve saturates at around 28 bits radix word length with a mean error of $2.6 \cdot 10^{-7}$. In the flow, the maximum absolute error has been set to 10^{-8}. Nevertheless, the parameter variation in the flow needs to have its granularity refined since different coefficients might have different optimal word lengths. A further analysis is presented in the next subsection. The flow is limited to a functional analysis of the FIR filter. Hence, the results should be passed to a flow using executable specifications at a lower level of abstraction, such as TL and RTL.

D. Integrating Design Space Exploration

As mentioned before, an executable flow includes control structures allowing to vary the inputs. Hence, the systematic input variation realizes a design space exploration (DSE). On the one hand, the inputs of a step can be explored limiting the DSE to a step. This refers to a step-oriented search. On the other hand, the aim is to find a suitable combination of all inputs for the steps of a flow. This relates to a flow-oriented search. Step-oriented and flow-oriented search are illustrated in Figure 9. The step-oriented search is limited to the inputs of a step, i.e., the parameters p1-2 or p3-4. Instead, the flow-oriented search aims at exploring all input combinations of a flow, here in the parameters p1-4. The step-oriented search has been focused in this paper. So far, an exhaustive search (ES) and heuristic technique (GA) are developed both applicable to the step-oriented and flow-oriented search. The authors refer to [41] for a comprehensive overview of state-of-the-art search techniques. In general, the DSE methods can be divided into the problem space or the solution/objective space. In the problem space, the parameters, defined in a specification, are considered. An example is a design of a register bank, for which a discrete set of word lengths (columns) and number of words (rows) are available. Now, all possible parameter combinations of columns and rows can be searched within the problem space. In this scenario, the solution space is driven by constraints, such as latency, power, and area. An according DSE strategy can be realized in an unguided or guided manner. ES is a representative of an unguided type allowing for an unbiased view on the design space. Heuristic search, such as hill climbing and GA, is a path-oriented method. It incorporates knowledge in order to guide the search along a path. The advantage is that the intermediate search results may be reused.

As mentioned before, parameters and constraints are similarly represented in a step and flow. Depending on the number of inputs and their range of values, a design space may be divided into sub spaces. The realization of the ES is rather trivial, for example the inputs can be iteratively incremented or taken from a predefined list. In this paper, a GA is presented implementing a heuristic search in the design space. The GA needs to be configured in terms of a minimization or maximization problem. An one-chromosome individual is used to describe the DSE problem. The chromosome includes an one-dimensional array of genes. Each gene denotes an input and the gene value defines an according value. For example, a chromosome $\boldsymbol{g} = (3, 2, 5)$ includes three inputs. The corresponding gene values are in the integer range. Hence, a set or range of values has to be defined for each input. Given a randomly initialized population, the GA generates its offspring via variation. Each chromosome is evaluated by calculating a fitness value. The calculation is done externally in a step and the fitness value is gathered by the GA. In addition, the GA prevents from recalculating already evaluated solutions. Furthermore, variation through an one-point mutation and order crossover enables an iterative improvement of the offspring. In an executable flow, the implementation of a step-oriented and flow-oriented search is realized by an expansion of the executed nodes, namely steps and flows. Figure 10 shows an iterative execution of many steps/flows parallelized via a selection node and

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

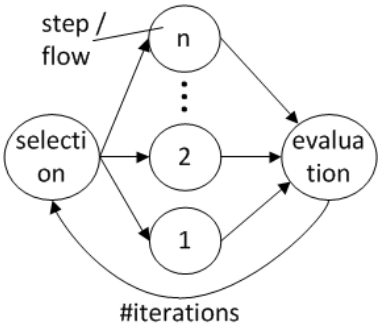


Figure 10. Step-/flow-oriented search via parallelization, synchronization, iteration, and feedback.

synchronized via an evaluation node. In case of an ES, only one iteration is necessary. For each iteration, the GA selects the steps/flows from the population and evaluates the individuals via a provided fitness value. Hence, the selection node performs the genetic operators, such as initialization, mutation, crossover, replacement, and selection. An end of the GA-based search is determined by the number of iterations (generations). This requires a feedback-loop between the selection and evaluation nodes. Further stopping criteria can be included. Moreover, the initialization of the GA population can be used to realize a random (Monte Carlo) search. Hence, the population size corresponds to the number of random samples and the number of generations is set zero.

The step-oriented search is demonstrated via the FIR filter example presented before. The number of taps (#taps) of the FIR filter is #taps=16 and the word length w of each coefficient radix is defined in the range of $1 \leq w \leq 31$ bits. Hence, 31^{16} input combinations motivate for solving the optimization problem via the GA search. Equation (1) defines the fitness (objective) function in terms of a minimization.

$$\gamma \cdot \underbrace{\left(\frac{1}{\#\text{taps}} \sum_{i=1}^{\#\text{taps}} \frac{w_i}{w_{\max}} \right)}_{\text{word length}} + (1-\gamma) \cdot \underbrace{\left(1 - \frac{e_{\text{abs}}^{\min}}{e_{\text{abs}}} \right)}_{\text{absolute error}} \rightarrow \min \tag{1}$$

As mentioned before, Equation (1) needs to be implemented in the FIR filter simulation in order to provide a fitness value for the GA. The fitness function finds a tradeoff between the conflicting goals of a minimal word length of the coefficients and a minimal absolute error. The weight γ realizes a prioritization between both goals. The first term minimizes the word length w_i of the taps i. In the example, the FIR filter requires 16 taps and coefficients, respectively. Then, $w_{\max} = 31$ bits denotes the maximum word length configurable in the FIR filter step. In addition, the second term targets a minimization of the absolute error e_{abs}. Referring to Figure 7, the error is calculated from comparing the filter output in case of quantized coefficients with a non-quantized reference generated via Matlab. Following, $e_{\text{abs}}^{\min}$ represents the minimum absolute error obtained from an FIR filter step by using the maximum word length $w_{\max} = 31$ bits for all coefficients. Figure 11 shows the GA search results in terms of two convergence plots. In the following, the GA is used in order to find a minimum fitness value. The maximum absolute error is set to $e_{\text{abs}}^{\max} = 10^{-3}$ in the FIR filter step. In case the constraint is violated, the FIR filter simulation returns a very large fitness value indicating an invalid solution. In addition, the weight $\gamma = 0.3$ prioritizes the error minimization according to the error constraint introduced before. From the FIR filter results in Figure 8, it is known that an average bit width of $\overline{w} = 15$ reaches a good solution holding the given error constraint. The goal is to reduce the average bit width $\overline{w}$ not violating the constraint. Hence, the bit width of the coefficients is varied in the interval $13 \leq w_i \leq 17$. Furthermore, the GA parameters are set as follows: $pSize = 50$, $nGen = 100$, $mRate = 0.1$, $cRate = 0.8$, and $rRate = 0.5$. In Figure 11, the upper plot shows that the GA converges after 85 generations with a fitness value of 0.6231. Please note, the small decrease of the fitness value at 85 generations is not visible in the figure. From the lower plot in Figure 11, the according absolute error $e_{\text{abs}} = 0.00081$ and average bit width $\overline{w} = 14.3125$ bits can be obtained. Hence, the applied GA search has reduced $\overline{w}$ by almost 5% compared to the result illustrated in Figure 8. In addition, the GA outperforms the average bit width $\overline{w}$, obtained via a Monte Carlo simulation and holding the error constraint, by around 12%. The GA generated 332 different solutions and the DSE finishes after 72 seconds on an Intel Core 2 Duo L7500 with 1.6 GHz utilizing one core. This shows the efficiency of the GA compared to the 5^{16} solutions of an exhaustive search and a solution via Monte Carlo simulation. Nevertheless, an optimal solution can not be guaranteed due to the heuristic nature of a GA.

V MODELING DESIGN FLOWS

This section introduces a meta-model representing an abstract flow model [39]. Moreover, flow patterns are shown in terms of reusable flow structures. Given the meta-model and patterns, a derivation of a flow is illustrated based on the modified λ-chart model [8].

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

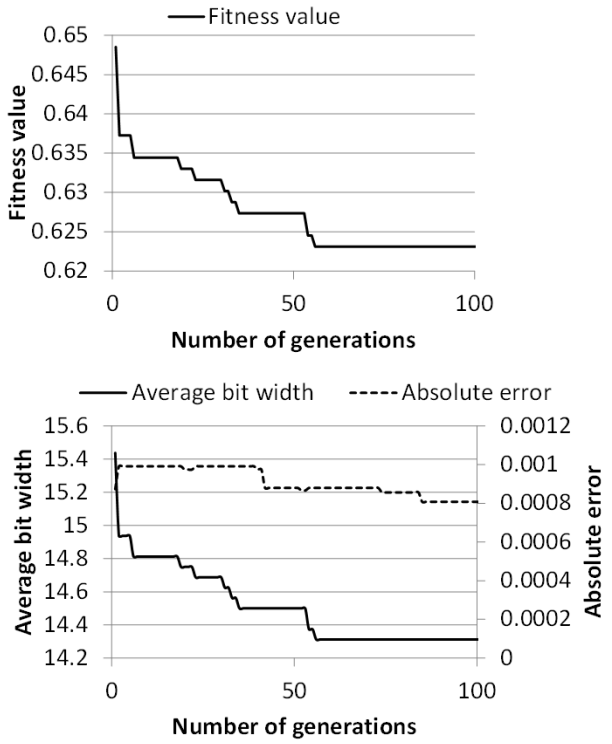


Figure 11. Convergence plots of the GA search in the FIR filter example.

A. *Meta-Model of Design Flows*

A meta-model has been developed in order to provide a minimal set of generic modeling elements necessary to build a flow. The meta-model is described via a UML class diagram, seen in Figure 12. It represents a fundament or kernel of the language design and implementation presented in the Section VI. The language elements relate to the meta classes. The *Element* class contains *Properties* and *Transitions* from/to elements. A transition between two elements is used to model a unidirectional dependency and a property represents an input, output, or further information added to an element. The transition also models a relationship between two flows. Moreover, both *Flow* and *Node* inherit from the element class. The assignment of elements to a view is realized via a property class. Moreover, a flow may include many nodes. Flows may have a nested structure consisting of many flows. This allows to reduce model complexity and to improve the reuse of available flows. Finally, a node represents an executable element, such as step, loop and branch nodes. Loop and branch nodes are further used to describe an algorithmic ordering of flows and steps, as introduced in Section B.

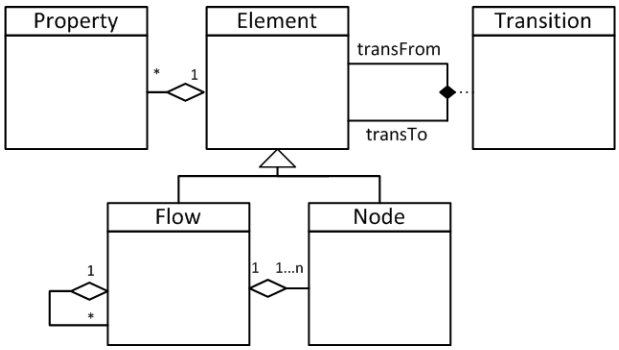


Figure 12. A meta-model for the derivation of design flows.

B. *Design Flow Patterns*

In addition to the meta-model described before, a derivation of recurring structures of flows allows to determine further modeling elements necessary for a systematic construction of flows. The flow patterns, illustrated in Figure 13, are a key enabler of the language design and implementation presented in Section VI. In principle, the patterns describe a parallel, iterative and conditional execution of flows. Pattern (a) models a data dependency between two steps. Hence, the subsequent step is fed with inputs produced by its predecessor. An example is that a scheduling step produces application mappings further being analyzed by a validation step. Moreover, a control dependency models decision making in a flow as seen in pattern (b). It shows a conditional statement deciding for one of two steps depending on the output of a previous step. An example is that only one of the two configurations of a scheduling step will be selected based on the output of a provisioning step. Moreover, pattern (c) describes a divide and conquer approach aiming at a recursive break down of a problem into sub-problems. A possible realization would be that a flow contains several sub-flows representing the sub-problems. In pattern (d), a parallel execution of many steps and the synchronization of the results are described. An example would be to execute the same step with different configurations multiple times in parallel and choosing the best output as input of a subsequent step. Moreover, pattern (e) and pattern (f) consider iterations in a flow. In pattern (e), a step is executed until an end condition reaches. For example, a step increments a parameter in order to find a suitable parameter value. Pattern (f) shows an iterative execution based on a feedback from a subsequent step. The information may allow for changing the selected inputs in order to improve a step result.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

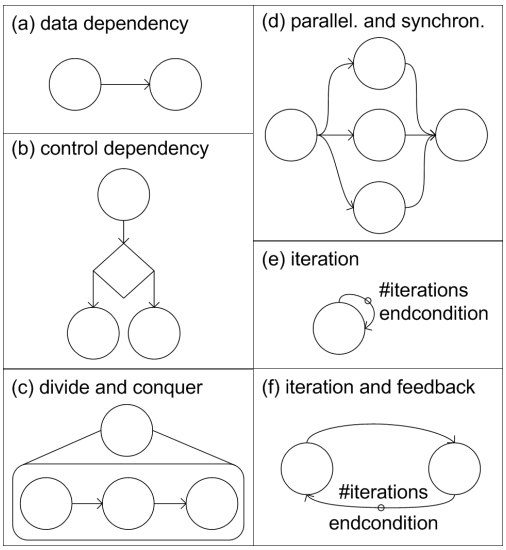


Figure 13. Reoccurring structures (patterns) in design flows.

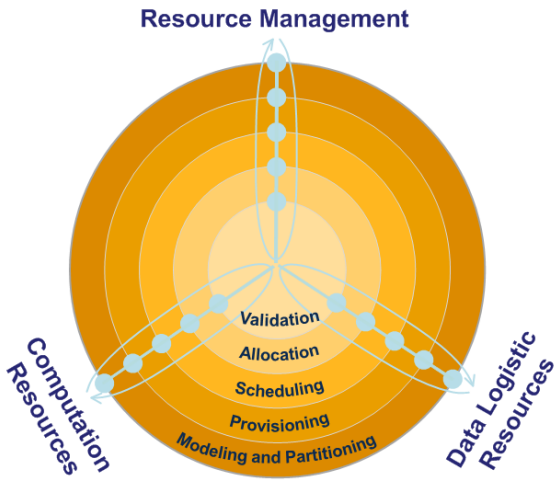


Figure 14. The modified λ-chart [8] - A model of design abstraction and exploration.

C. Domain-Specific Design Flow

In a previous work [8], the authors introduced the *λ-chart*, which represents a model of design abstraction and exploration. It addresses an ESL design of MPSoCs and future many-core SoCs at an early stage. The motivation was to provide the designer with a flow model allowing for a clear definition of the steps and a separation of the important system functions. Therefore, an *administration* view was included in order to highlight the rising importance of management functions in embedded systems. The model further allows to combine the different steps of a flow. In the following, the λ-chart has been slightly modified in order to focus more on the design and exploration of the system resources, as illustrated in Figure 14. In addition, the term administration has been replaced by a more management-centric point of view. Hence, the λ-chart defines three views allowing to separate the orthogonal system functions. A *resource management* view considers tasks for planning, assignment, monitoring, and control. Instead, a *computation resources* view relates to the code execution. Moreover, a *data logistic resources* view addresses a design of data storage and data exchange between components. Furthermore, the concentric bands underline the five steps of a unified process. The *modeling and partitioning* step describes a starting point in order to build the representations of the system structure and behavior. *Partitioning* focuses on the parallelization of applications. Following, *provisioning* means to select the type and number of components and behavior necessary to fulfill the purpose of the intended system. In *scheduling*, a temporal planning of the computation, data logistics and management is applied. This includes both the application and architectural components, such as determining an execution sequence, power-aware planning, monitoring, etc. Moreover, the *allocation* step focuses on spatial planning, such as placement and packaging of components, and application binding. Finally, *validation* proves whether the system fulfills a previously defined purpose. The authors refer to [8] for a more detailed explanation.

The λ-chart follows the meta-model presented in Section A. That means, a step is derived from the node element and a flow is a sequence of steps connected via transitions. Moreover, a view is modeled via the property element. An example of a flow, depicted in Figure 15, demonstrates the derivation of a flow from the λ-chart. Three steps, limited to the *computation resources* view, have been chosen. The combination of the steps and a connection via transitions build the flow. The block diagram in Figure 15 shows an equivalent representation of the flow. In addition, control primitives, such as a branch node (if-then-else, switch-case) and loop node (for/while), are inserted in a flow enabling a parallel, iterative and conditional execution of the flow. This allows to realize the flow patterns presented before. In Section IV, the DODF approach was introduced, giving the designer a methodology to select appropriate flows, views, steps, etc. The control structure is build via an algorithmic order of the steps. Figure 16 details the instantiation from the *Element*, *Transition* and *Property* classes defined in the meta-model. Figure 16 (left) shows

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

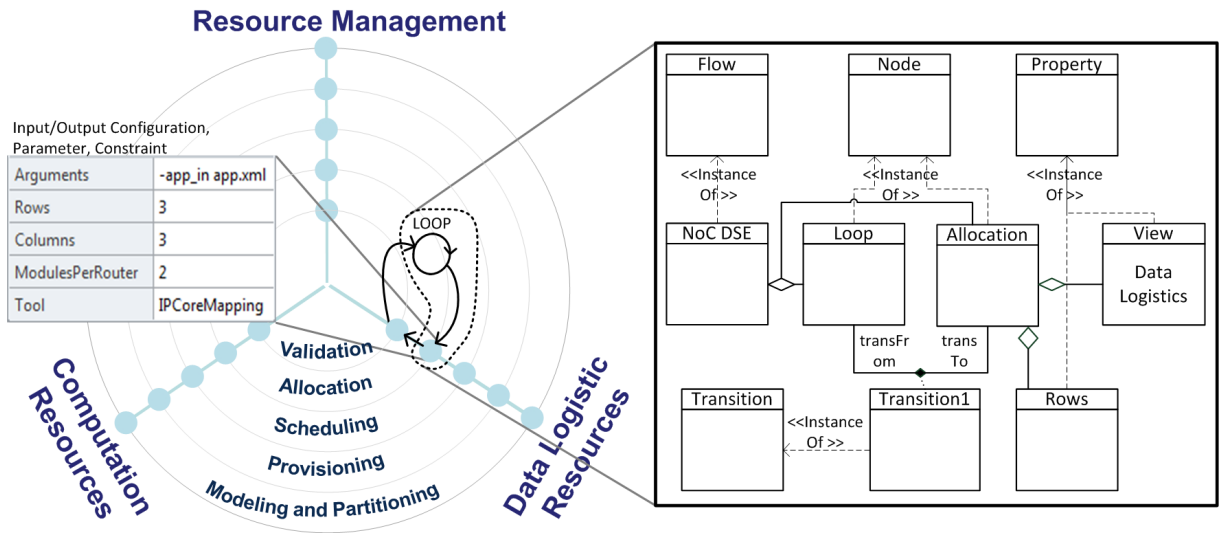


Figure 16. An example of a λ-chart flow with instantiation from the meta-model.

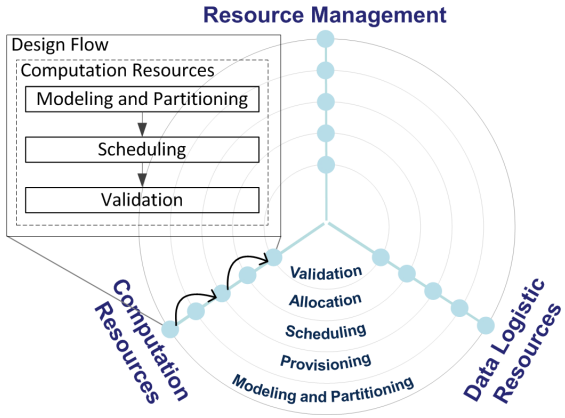


Figure 15. An example for the derivation of a flow in the modified λ-chart.

a flow traversing the *allocation* and *validation* steps iteratively. The DSE is restricted to the *data logistic resources* view. In the following, a limited part of the flow, marked by a dotted line, is considered. Referring to Figure 16 (right), the example focuses on the allocation step, loop node, and transition from the loop to allocation. The loop node controls an iteration of the input parameters of allocation and includes an exit condition. Moreover, the flow is named network-on-chip (NoC) DSE. NoC is a promising network design approach for scaling from MPSoC to many-core systems because the efficient communication infrastructure supports a large amount of IP cores [42, 43]. As mentioned before, an assignment of the allocation step to a view is realized via the *Property* class. The step also includes properties, such as the number of rows in a NoC. Hence, the properties are used as input parameters of a step.

VI ESL DESIGN AUTOMATION

A comprehensive list of academic and commercial EDA environments for ESL design can be found in [2]. Modern environments address DSE but with the limitation to a proprietary implementation for a specific design problem, such as optimization of the application mapping. Recent research introduces generic infrastructures turning away from ad-hoc software [6, 7]. Nevertheless, the complexity of flows for future embedded systems is not yet considered. The large number of flows, steps, inputs and outputs requires a more systematic development. In addition, commercial EDA systems allow for a flexible and efficient implementation of flows via scripting languages. The major drawback of academic and commercial EDA systems is that no systematic development, management and optimization of flows is supported. The user is either dependent on a proprietary implementation or has to develop a representation of a flow by oneself. This paper presents two programming languages [39] addressing these problems and supporting all aspects of our DODF approach. Therefore, the user is supported in developing, managing, and optimizing a flow. This includes flexible and efficient realization of DSE strategy in the flow via little program code. First of all, a visual programming language is introduced. This language has been evolved to a textual programming language, called

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

design flow language (DFL). A tool flow, enabling DFL, is presented afterwards.

A. Visual Programming of Flows

A visual programming of flows has been implemented via a graphical prototype based on Microsoft Visio by the authors [1]. It realizes the concepts introduced in the Sections IV and V. The implementation allows to instantiate steps and flows via drag-and-drop and copy functions using the λ-chart model. The graphical user interface (GUI) corresponds to the visualization in Figure 16 (left). The construction of a flow from the GUI has been realized via the visual basic for applications (VBA) programming language by detecting the dependencies between the steps and reading the properties of the steps. The prototype includes an import/export function in order to load and store the flows based on a predefined XML-format. The definition of the XML-format is explained via a simple flow, illustrated in Listing 1. The flow corresponds to the Figure 16 (left). The XML-file is read by an interpreter program implemented in C++. The interpreter allows for a sequential and parallel execution of the steps. Referring to Listing 1, the *flow* and *node* tags follow the meta-model presented in Section A. The step and loop nodes are connected via transitions and include many properties. Moreover, the loop node requires a loop/exit body and an exit condition in order to traverse the flow iteratively. In a *property* value, expressions and system functions are used to read and modify variables, directories, and files during a step execution.

```
1  <?xml version="1.0" encoding="UTF-8"?>
   <flow>
3    <node name="My Allocation">
       <property name="Step" value="Allocation"/>
5      <property name="View" value="Data Logistic
           Resources"/>
       <property name="Rows" value="3"/>
7      <property name="Arguments" value="-app_in
           lambda\\apps_state.xml ..."/>
       <property name="Tool" value="IPCoreMapping
           "/>
9      <!-- ... -->
     </node>
11   <node name="My Validation">
       <property name="Step" value="Scheduling"/>
13     <property name="View" value="Data Logistic
           Resources"/>
       <!-- ... -->
15   </node>
     <node name="My Loop">
17     <property name="type" value="LOOP"/>
       <property name="loop_body" value="Rows=
           Rows+1; ..."/>
19     <property name="exit_condition" value="
           Rows==4; ..."/>
       <property name="exit_body" value="
           renameDir(lambda\\maps,Rows); ..."/>
21   </node>
   <transition source="My Allocation" target="My
       Validation"/>
23 <transition source="My Validation" target="My
       Loop"/>
   <transition source="My Loop" target="My
       Allocation"/>
25 </flow>
```

Listing 1. XML source code imported/exported by the visual programming prototype.

Referring to Listing 1, the step "My Allocation" (lines 3-10) and the step "My Validation" (lines 11-14) are created. Therein, several properties are defined, such as *Step*, *View*, etc. Moreover, the *Rows* property (line 6) is initialized to three. Together with the Arguments (line 7), Rows will be used as input of the *IPCoreMapping* tool (line 8). Moreover, the loop node (lines 16-21) defines several expressions in order to increment the *Rows* property (line 18), to check for the exit condition (line 19), and to define an action after the exit (line 20). Finally, the flow is constructed by connecting the steps via transitions (lines 22-24).

Nevertheless, the XML-format makes it inconvenient to program multiple expressions, nested conditions, nested loops, and feedback loops. In addition, a reuse of flows and steps is not supported. The limitations motivated for an evolution towards the DFL representing an efficient and flexible programming language.

B. Design Flow Language (DFL)

DFL is specially targeted to a development, management and optimization of flows including the necessary control and automation capabilities. Moreover, design space exploration (DSE) is directly considered in the language design and implementation. The requirements and structure of DFL are shortly introduced in the following. A simple flow example illustrates the use of the language. For more details on the language, the authors refer to [39].

Language Requirements

The purpose of DFL is to make the design of future embedded systems more flexibly and efficiently via a systematic development of flows. This includes management and optimization capabilities. The requirements are summarized in the following. A clean syntax increases the user's productivity. Program commands for the construction of flows are necessary. As in modern programming languages, control structure and program modularization enable more complex applications. Moreover, an acceleration of flows via parallelization

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

should be realized. The use of DSE techniques within a step or flow will allow to find an optimal or feasible solution in design spaces with different complexity. Further requirements relate to the EDA tools and design data accessible via DFL. An executable file or library needs to be assigned to a step. Moreover, design data should be accessible via data structures, files, and data base operations. In addition, some kind of inter-process communication serves as interface between the EDA tools. Finally, non-functional requirements address an access from/to other programming languages. Moreover, data analysis and debugging support will be beneficial in a flow development.

Language Structure

The DFL is an imperative (procedural) programming language read by an interpreter program. The interpreter controls an execution of the steps defined in a flow. The syntax is derived from the C/C++ programming language widely known in HW/SW programming. The *Flow*, *Step*, *Property* and *Transition* classes, defined in the meta-model and introduced in Section A, have been integrated in the language design and implementation. Modularization is realized via subroutines and an #include statement. Basic data types (bool, int, double, string) and complex data types (vector, Flow, Step) are available. DFL is further a structural programming language supporting a full set of control primitives, such as for, while, if-then-else and switch-case. The language includes a limited number of keywords and various input/output names are reserved for the step and flow. DFL additionally supports typical arithmetic operators, logical operators, and vector indexing. Moreover, commands are case sensitive and single statements must be ended with a semicolon.

A Simple Design Flow in DFL

In the Listing 2, a simple flow is described in DFL illustrating its structure. The program accomplishes an execution of two dependent steps in a flow, which corresponds to Figure 16 (left). Lines 2-8 relate to the configuration of an *allocation* step. This includes an assignment of an executable file, called alloc(.exe), to the step (line 3). The executable requires arguments (line 6) and an input (line 7) in order to solve the IP core mapping problem (line 4). In addition, the *View* parameter corresponds to the λ-chart in Figure 16. Since the step allows for several input combinations, here indicated via the *rows* vector (line 7), it is configured for a parallel execution (lines 10-13). A *space* vector contains the variables defining the input combinations (lines 10-11). The input parameter *HPCJob* (line 13) configures an available high performance cluster (HPC) environment for a parallel execution of the steps. Then, a *validation* step (lines 15-16) is instantiated. Further assignments to the step are left out for simplification. Finally, the flow is constructed (lines 19-22) and executed (line 24). The steps need to be added to the flow (line 20) and the execution order is determined via the *connect* function (line 21). Line 22 saves the flow description in the visualization of compiler graph (VCG) format [44] allowing to check the flow structure.

```
1  /******** ALLOCATION STEP ********/
   Step s1 = Step("Allocation");
3  s1.add("Execution","alloc");
   s1.add("Tool","IPCoreMapping");
5  s1.add("View","Data Logistic Resources");
   s1.add("Arguments","-app_in lambda\\apps_state
       .xml ...");
7  vector<int> rows = [3:4];
   // ...
9  /******** PARALLEL EXECUTION ********/
   vector<string> space;
11 space.push_back("rows");
   s1.add("Space","space");
13 s1.add("HPCJob", "true");
   /******** VALIDATION STEP ********/
15 Step s2 = Step("Validation");
   s2.add("Execution","valid");
17 // ...
   /******** FLOW CONSTRUCTION ********/
19 Flow f;
   f.add(s1); f.add(s2);
21 connect(s1,s2);
   f.save("vcg", "flow.vcg");
23 /******** FLOW EXECUTION ********/
   execute(f);
```

Listing 2. Simple design flow in DFL.

DFL Tool Flow

In the following, the tool flow for the DFL is presented. As typical for modern programming languages, it is separated into frontend, middle-end, and backend. Figure 17 illustrates the tool flow. The frontend includes a scanner and parser to validate the DFL syntax. The scanner splits the DFL source code into tokens by recognizing lexical patterns in the text. GNU Flex [45] has been used to generate the scanner (lexical analyzer). Then, the parser applies syntax-rule matching. The parser has been generated using GNU Bison [46]. From the parsing results, an abstract syntax tree and a statement list are derived. In addition, a symbol table holds information about the program. The statement list and symbol table allow to interpret and optimize the program code,

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

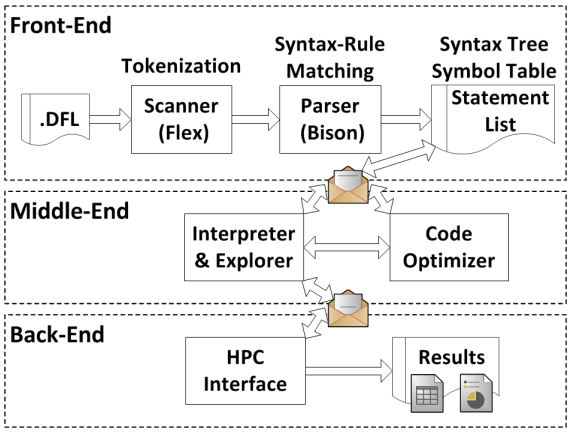


Figure 17. Tool flow for the design flow language.

as done in the middle-end. The interpreter is responsible for type checking, type erasure (conversion), and expression evaluation. The code optimization refers to an exploitation of the step-level and flow-level parallelism. As mentioned before, the interpreter supports an export of the flow structure in the VCG format [44] in order to visualize the graph. Moreover, the explorer includes an exhaustive, random and heuristic search allowing to explore design spaces with different complexity. Finally, the backend provides functionality executing a DFL program on a single computer or HPC. After a step execution, according design and validation data will be available for a further analysis. The next stage is to merge DFL and model/method design into an integrated development environment (IDE), presented in [39]. Therein, the design methods and simulation models are implemented via a native language, such as C/C++, in order to fulfil the critical performance requirements. The aim is to compile an executable or library and assign it directly to a DFL step in one IDE realizing a seamless development. Then, the flow can be executed, tested, and optimized in the IDE. The DFL implementation includes a full set of language features. Open topics relate to the implementation of performance analysis functions, plotting functions, and database access. Furthermore, a future DFL revision needs to address name spacing avoiding naming conflicts.

VII DESIGN FLOW CASE STUDY

This section demonstrates the concept of an executable flow and the DODF approach under realistic conditions. The case study targets an ESL design of the heterogeneous multicluster architecture, as introduced by the authors in [1]. The multicluster architecture represents a promising candidate for future embedded many-core SoCs [12]. The outline of this section is as follows: First, a description of the application and architecture model forms the basis of the underlying simulation models. Next, an according sequence of flows is introduced showing a separation of the addressed design problems. This allows to solve the complex problems more flexibly and more efficiently as compared to a proprietary and fully integrated design flow. Due to a lack of space, only the dimensioning of the multicluster architecture is selected for a more detailed explanation in terms of a design methodology, flow description, and the experimental results.

A. *Application and Architecture Model*

The models consider functionalities of the three views defined in the modified λ-chart, seen in Figure 14. The application model includes multiple, concurrently running applications and threads, respectively. A thread is represented by a high-level task graph and it sequentially executes tasks. Threads are only synchronized before or after execution. Then, a task is an atomic kernel exclusively executing on an intellectual property (IP) core, e.g., processing element (PE), memory (MEM) interface, control processor (CP) interface, etc. Tasks produce and consume chunks of data accessed via shared memory. Side effects are excluded by preventing access to external data during computation.

As shown in Figure 18, the architecture model is a heterogeneous set of multiprocessor system-on-chips (MPSoCs) and clusters, respectively. The management unit (MU) represents an application processor and includes a load balancer aiming at equally distributing thread load amongst the clusters. Moreover, an MPSoC contains heterogeneous types and numbers of IP cores. In the model, each MPSoC contains a network-on-chip (NoC) connecting the IP cores. Moreover, each cluster includes a CP responsible for dynamically scheduling arriving tasks to the available IP cores. The CPs are directly connected to the MU. The heterogeneous multicluster architecture, seen in Figure 18, includes a regular 2D mesh NoC. Each tile contains a router and n modules (IP cores). A module can be an MEM, CP, or PE, such as general purpose processor (GPP), digital signal processor (DSP), application-specific integrated circuits (ASIC), etc.

B. *Sequence of Design Flows*

This case study is composed of five flows using different design methods and system models. Figure 19 illustrates a sequence of the flows. Further flows can be added, such a memory optimization. The heteroge-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

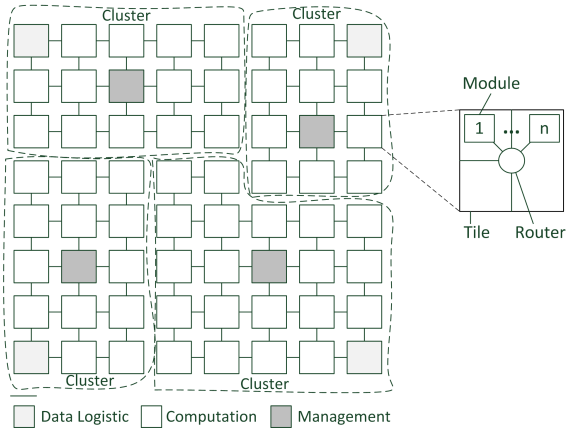


Figure 18. An architecture model for the heterogeneous multicluster.

neous multicluster architecture implies a wide diversity in terms of structural, behavioral (functional) and physical parameters. DFL programs have been developed for the flows. Therein, the view and step definitions follow the modified λ-chart model. Referring to Figure 19, this case study addresses input parameters of the design method, structural design, behavioral design, and physical design. In the following, the flows are shortly introduced and a DFL program for the sequence of flows is presented. The rest of this section will focus on the multicluster dimensioning.

- *Parameter Tuning* aims at finding the best tool parameters for a GA solving the IP core mapping problem [47];
- *Multicluster Dimensioning* creates a heterogeneous multicluster architecture by distributing the anticipated application load among clusters and solving the optimization problem via a genetic algorithm (GA) and mixed-integer linear programming (MILP) formulation [48];
- *IP Core Mapping* places IP cores in an 1-ary n-mesh NoC constrained by the number of modules at each router. The optimization problem is solved via a GA and MILP formulation [47];
- *NoC Arbitration* and *Multicluster Load Balancing* aim at finding suitable behavioral schemes from a selection based on simulation results. *NoC Arbitration* compares a locally fair with a globally fair arbitration scheme [49]. In addition, flit-based and packet-based switching are considered. *Multicluster Load Balancing* compares different estimators of cluster load, such as response time and queue size, used in the load balancing scheme of an MU.

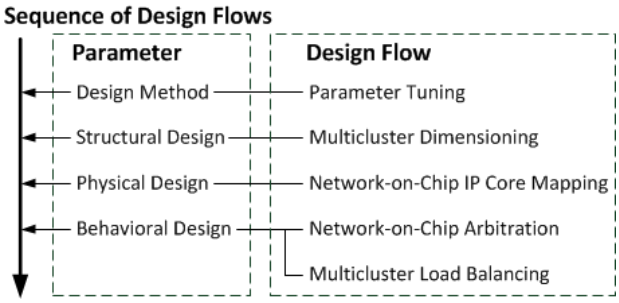


Figure 19. The sequence of flows in the case study.

In the following, a DFL program for the sequence of the flows is presented. The flows *Parameter Tuning* and *Multicluster Load Balancing* are used as examples. The source code of *parameter_tuning.dfl*, shown in Listing 4 in the appendix, gives a deep insight into a flow developed in DFL. Referring to Listing 3 (lines 2-3), the #include directive allows to insert predefined DFL source code as mentioned in the previous section. The variables *tun* and *bal* are declared in one include file and they represent the predefined flows of the *Parameter Tuning* and *Multicluster Load Balancing*. After the #include, an execution sequence is scheduled by inserting an identifier for each flow in the vector *flow_order* (lines 5-9). Thereafter, the vector is iterated (lines 11-42) and a switch-case statement (lines 17-38) lists the available flow choices. If a flow matches a case statement, it is executed and a status message is displayed (lines 40-41). The example further includes two specific inputs and vectors, respectively (line 15). The elements of each vector are used for a DSE purpose, such as searching for the best configuration. The DSE is declared in the steps. The input *arch_in* represents a set of available architecture configurations. The elements in the vector are used as parameter values for the *tun* step (line 21) and the *bal* step (line 28). In addition, the vector *config* includes different configurations of the simulation setup for the *bal* step in order to select a suitable load balancing scheme (line 28). The sequence of flows can be further extended in terms of additional flows, inputs, and, commands.

C. Multicluster Dimensioning

Given a set of target applications, the *Multicluster Dimensioning* flow realizes a provisioning of resources in the heterogeneous multicluster architecture [48]. The aim is to generate an appropriate distribution of the applications onto the clusters containing different types and numbers of PEs. The E3S Benchmark Suite [50] is

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

used as basis of the applied application scenario. E3S is largely based on data from the Embedded Microprocessor Benchmark Consortium [51]. The included task graphs describe periodic applications. The 20 applications range from automotive, industrial, telecommunication, networking to general-purpose applications. An application scenario is built from the concurrently running task graphs.

An overview of the methodology is illustrated in Figure 20. Besides optimization of the multicluster architecture, the flow applies further methods, such as estimation, (architecture) refinement, simulation, and validation. Referring to Figure 20, the first step is to extract a parallelism value matrix $\mathbf{\Phi}$ via parallelism analysis, introduced by the authors in [48]. The matrix is used as input for the optimization via a GA and MILP formulation. Given the optimized cluster configurations, the selected IP cores are used to generate an multicluster architecture. Then, the dynamic mapping of an application onto the refined architecture is simulated. Each task of an application is dynamically mapped onto an IP core at runtime assuming a point-to-point communication protocol between the directly connected IP cores. Each task is executable on at least one IP core of the refined architecture ensuring schedulability. Moreover, a task execution is prioritized based on its deadline. Afterwards, the mapping results are validated by an average thread response time quantifying the system performance. Response time defines the time from the request of a thread until its end including a possible network delay.

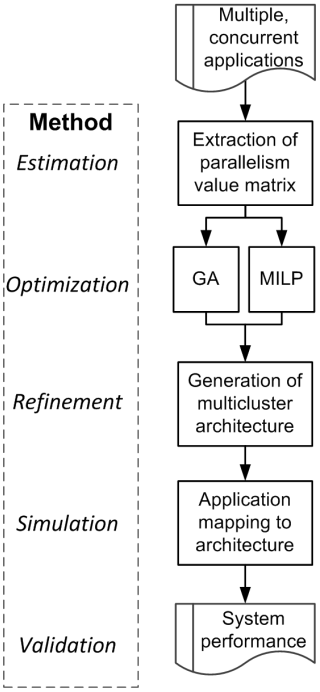


Figure 20. Methodology of the *Multicluster Dimensioning* flow.

A compact flow description, seen in Figure 21, is realized via the modified λ-chart. The flow focuses on a suitable computation infrastructure for the heterogeneous multicluster architecture. Hence, DSE is limited to the *computation resources* view. The *modeling and partitioning* step serves as a starting point without any further purpose. In the *provisioning* step, a target application and the available IP cores are used to generate the heterogeneous multicluster architecture. As mentioned before, the optimization problem is solved via a GA and MILP formulation. The subsequent *scheduling* step performs an application mapping via simulation. An according simulation model performs both a temporal and spatial mapping of the tasks to the available PEs dynamically at runtime. The results are analyzed in the *validation* step. Referring to Figure 21, a loop node increments a maximum allowed number of PEs in a cluster ($\#PEs_{max}$). For the simulations, the value range of the input constraint is set to $3 \leq \#PEs_{max} \leq 7$.

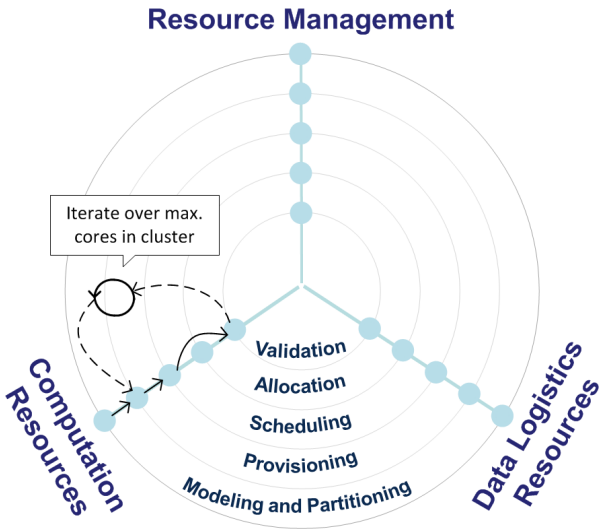


Figure 21. Overview of the *Multicluster Dimensioning* flow via the modified λ-chart.

In the literature, to the best knowledge of the authors, multicluster dimensioning was not yet applied for the E3S Benchmark Suite [50]. In order to compare the re-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

sults, a single-cluster configuration with nine PEs is provided as reference in Figure 22. It has also been generated with the *Multicluster Dimensioning* flow. Application mapping onto the single-cluster architecture results in over 40 % thread cancelation. Then, using the thread response time as a metric would be meaningless, hence the total amount of PEs is considered as a reference.

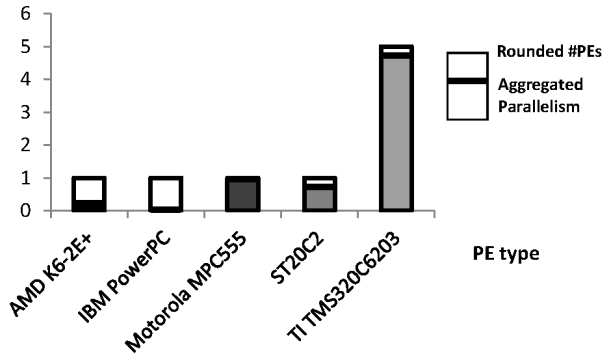


Figure 22. Single-cluster reference for the *Multicluster Dimensioning* flow.

Figure 23 shows the validation results in terms of a total number of clusters/PEs and (average) thread response time. The latter includes the impact of the dynamic scheduling scheme. The GA has been used to solve the multicluster dimensioning problem. All values have been normalized to the largest occurring value. The selection of a suitable solution bases on a tradeoff between the conflicting goals of a minimum number of resources and a minimum thread response time. In the figure, $\#PEs_{max} = 7$ (red arrow) is selected as the best tradeoff. Its application mappings did not produce aborted threads. It includes a minimum number of clusters of three and PEs of eleven. As mentioned before, each cluster contains a CP further increasing the number of resources in the system. In the result, the number of clusters and PEs do not change for the larger $\#PEs_{max}$ values. But due to its heuristic nature, the GA produced the best solution in terms of a thread response time for $\#PEs_{max} = 7$. The resulting configuration, depicted in Figure 24, represents a heterogeneous multicluster solution since all clusters are heterogeneous in terms of PE types (depicted by different shades of grey). In the Figure 24, it is shown that the PEs of the PE types *AMD K6-2E+* and *IBM PowerPC* are marginally used. The both GPPs are able to execute most of the tasks in the benchmark. The remaining PEs are well utilized using the anticipated application load based on the average parallelism values. The configuration shows improvement potential in the cluster C2. A solution would be to exclude the *IBM PowerPC* from the mapping option table in order to reduce the number of PEs in the cluster by one PE. This requires that the PE can be replaced and no additional PE is necessary to perform the tasks assigned to the *IBM PowerPC*. Hence, the total number of PEs decreases to ten.

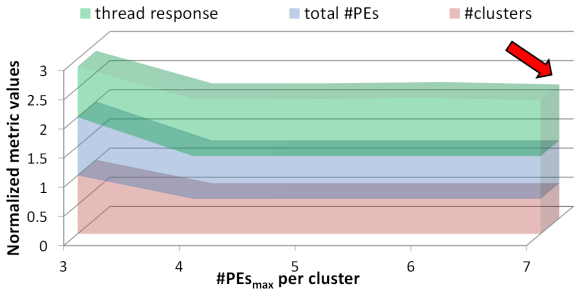


Figure 23. Normalized results of the *Multicluster Dimensioning* flow.

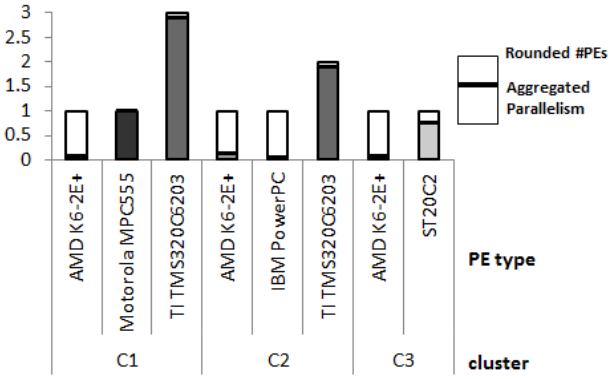


Figure 24. Best multicluster configuration of the *Multicluster Dimensioning* flow.

VIII CONCLUSION AND OPEN TOPICS

The large number of inputs and steps in the flows for future embedded systems necessitates the development of a systematic design of design flow (DODF) approach. Then, the concept of an executable flow allows for executing steps in the same way instructions of a program are processed. Both contributions of this paper are exemplified via a functional exploration of an FIR filter. Afterwards, the modeling principles of flows are explained. The idea of abstracting the flows and a corresponding derivation of a domain-specific flow are focused. The concepts are the motivation for a visual and textual design flow language. The design automation allows for a development, management, and optimization of flows.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Design space exploration is directly considered in the language design and implementation. Finally, a case study demonstrates a realistic ESL design of the heterogeneous multicluster architecture. The five flows are arranged in a sequence of flows. Each flow outputs experimental results representing suitable solutions for the individual design problems.

In the rest of this paper, a discussion outlines the future work. An open topic relates to the further development of DFL towards additional language features, such as name spacing, profiling, etc., allowing for more complex applications. In addition, the language should provide advanced access and functions to analyze the design data. It would be beneficial to support more DSE techniques, such as simulated annealing, hill climbing, etc. In addition, the flow-based search is an open topic. The implementation of DFL comprises a full set of language features opposed to the visual language, which requires several adjustments, such a support of sub-flows in a flow. In future, the design flow development should be extended towards a high-level synthesis for embedded systems.

VIII APPENDIX: DFL FLOW EXAMPLE

The appendix illustrates the DFL source code for the *Parameter Tuning* flow through Listing 4.

REFERENCES

[1] F. Guderian and G. Fettweis, "Integration of design space exploration into system-level specification exemplified in the domain of embedded system design," in *Proceedings of International Conference on Advances in Circuits, Electronics and Micro-electronics (CENICS)*, Aug. 2012.

[2] D. D. Gajski, S. Abdi, A. Gerstlauer, and G. Schirner, *Embedded System Design: Modeling, Synthesis and Verification*. Springer, 2009.

[3] R. Ernst, "Automatisierter entwurf eingebetteter systeme," *at - Automatisierungstechnik*, pp. 285–294, jul 1999.

[4] B. Bailey, G. Martin, and A. Piziali, *ESL design and verification: a prescription for electronic system-level methodology*, 1st ed., W. Wolf, Ed. Morgan Kaufmann, 2007.

[5] D. D. Gajski, F. Vahid, S. Narayan, and J. Gong, *Specification and design of embedded systems*. Prentice-Hall, Inc., 1994.

[6] W. Fornaciari, G. Palermo, V. Zaccaria, F. Castro, M. Martinez, S. Bocchio, R. Zafalon, P. Avasare, G. Vanmeerbeeck, C. Ykman-Couvreur, M. Wouters, C. Kavka, L. Onesti, A. Turco, U. Bondi, G. Marianik, H. Posadas, E. Villar, C. Wu, F. Dongrui, Z. Hao, and T. Shibin, "Multicube: Multi-objective design space exploration of multi-core architectures," in *Proceedings of IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, July 2010, pp. 488 –493.

[7] Z. J. Jia, A. Pimentel, M. Thompson, T. Bautista, and A. Nunez, "Nasa: A generic infrastructure for system-level mp-soc design space exploration," in *Proceedings of Embedded Systems for Real-Time Multimedia (ESTIMedia)*, Oct 2010, pp. 41 –50.

[8] F. Guderian and G. Fettweis, "The lambda chart: A model of design abstraction and exploration at system-level," in *Proceedings of International Conference on Advances in System Simulation (SIMUL)*, 2011, pp. 7–12.

[9] R. A. Fisher, *The Design of Experiments*. Oliver and Boyd Ltd., Edinburgh, 1935.

[10] G. L. Glegg, *The Design of Design*, 1st ed. Cambridge University Press, 1969.

[11] F. Brooks, *The Design of Design: Essays from a Computer Scientist*. Addison-Wesley, 2010.

[12] K. I. Farkas, P. Chow, N. P. Jouppi, and Z. Vranesic, "The multicluster architecture: reducing cycle time through partitioning," in *IEEE/ACM International Symposium on Microarchitecture (Micro)*, 1997, pp. 149–159.

[13] ITU-T, *Recommendation Z.100 (08/02) Specification and Description Language (SDL)*, International Telecommunication Union (2002).

[14] S. Traboulsi, F. Bruns, A. Showk, D. Szczesny, S. Hessel, E. Gonzalez, and A. Bilgic, "Sdl/virtual prototype co-design for rapid architectural exploration of a mobile phone platform," in *Proceedings of international SDL conference on design for motes and mobiles*, 2009, pp. 239–255.

[15] A. S. Initiative. (26 May 2013) Systemc, osci. [Online]. Available: http://www.systemc.org/

[16] D. D. Gajski, R. Zhu, J. Dömer, A. Gerstlauer, and S. Zhao, *SpecC Specification Language and Methodology*. Kluwer Academic Publishers, 2000.

[17] C. Haubelt, T. Schlichter, J. Keinert, and M. Meredith, "Systemcodesigner: automatic design space exploration and rapid prototyping from behavioral models," in *Proceedings of the 45th annual Design Automation Conference*, ser. Proceedings of Design Automation Conference (DAC), 2008, pp. 580–585.

[18] R. Dömer, A. Gerstlauer, J. Peng, D. Shin, L. Cai, H. Yu, S. Abdi, and D. D. Gajski, "System-on-chip environment: a specc-based framework for heterogeneous mpsoc design," *EURASIP Journal on Embedded Systems*, vol. 2008, pp. 5:1–5:13, Jan. 2008.

[19] J. Eker, J. Janneck, E. Lee, J. Liu, X. Liu, J. Ludvig, S. Neuendorffer, S. Sachs, and Y. Xiong, "Taming heterogeneity - the ptolemy approach," *Proceedings of the IEEE*, vol. 91, no. 1, pp. 127–144, jan 2003.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[20] L. Bonde, C. Dumoulin, and J.-L. Dekeyser, "Metamodels and mda transformations for embedded systems." in *FDL*, 2004, pp. 240–252.

[21] D. Mathaikutty, H. Patel, S. Shukla, and A. Jantsch, "Ewd: A metamodeling driven customizable multi-moc system modeling framework," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 12, no. 3, pp. 33:1–33:43, May 2008.

[22] D. Mathaikutty and S. Shukla, "Mcf: A metamodeling-based component composition framework–composing systemc ips for executable system models," *IEEE Transactions on VLSI Systems*, vol. 16, no. 7, pp. 792 –805, july 2008.

[23] "Synopsys Inc." 26 May 2013. [Online]. Available: http://www.synopsys.com

[24] "Cadence Design Systems Inc." 26 May 2013. [Online]. Available: http://www.cadence.com/

[25] "Mentor Graphics Inc." 26 May 2013. [Online]. Available: http://www.mentor.com/

[26] B. Welch, *Practical Programming in Tcl and Tk*, 4th ed. Prentice Hall, 2003.

[27] T. Barnes, "Skill: a cad system extension language," in *Design Automation Conference, 1990. Proceedings., 27th ACM/IEEE*, jun 1990, pp. 266 –271.

[28] Q. Nguyen, *CAD Scripting Languages: A collection of Perl, Ruby,Python,TCL & SKILL scripts*. Ramacad Inc.

[29] A. Sangiovanni-Vincentelli, G. Yang, S. Shukla, D. Mathaikutty, and J. Sztipanovits, "Metamodeling: An emerging representation paradigm for system-level design," *Design Test of Computers, IEEE*, vol. 26, no. 3, pp. 54 –69, may-june 2009.

[30] "Definition of model," 26 May 2013. [Online]. Available: http://en.wikipedia.org/wiki/Model

[31] M. Stephenson, S. Amarasinghe, M. Martin, and U.-M. O'Reilly, "Meta optimization: improving compiler heuristics with machine learning," in *Proceedings of the ACM SIGPLAN*, ser. PLDI '03. ACM, 2003, pp. 77–90.

[32] National Instruments, "Labview," 26 May 2013. [Online]. Available: www.ni.com/labview

[33] Mathworks, "Matlab and simulink," 26 May 2013. [Online]. Available: http://www.mathworks.com/

[34] G. Berry, "The constructive semantics of pure esterel." 26 May 2013. [Online]. Available: http://www-sop.inria.fr/esterel.org/

[35] P. Caspi, D. Pilaud, N. Halbwachs, and J. A. Plaice, "Lustre: a declarative language for real-time programming," in *Proceedings of the 14th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, 1987, pp. 178–188.

[36] IBM, "Ibm rational rhapsody," 26 May 2013. [Online]. Available: http://www.ibm.com/software/awdtools/rhapsody/

[37] D. D. Gajski, J. Peng, A. Gerstlauer, H. Yu, and D. Shin, "System design methodology and tools," *CECS, UC Irvine, Technical Report CECS-TR-03-02*, January 2003.

[38] S. Borkar, "Thousand core chips: a technology perspective," pp. 746–749, 2007.

[39] F. Guderian, *Developing a Design Flow for Embedded Systems*. Jörg Vogt Verlag, 2013.

[40] T. Kogel, A. Haverinen, and J. Altis, "Ocp tlm for architectural modelling," *OCP-IP white-paper*, 2005.

[41] M. Gries, "Methods for evaluating and covering the design space during early design development," *Journal Integration, the VLSI Journal*, vol. 38, no. 2, pp. 131–183, Dec. 2004.

[42] P. Guerrier and A. Greiner, "A generic architecture for on-chip packet-switched interconnections," in *Proceedings of Design, Automation, and Test in Europe (DATE)*, 2000, pp. 250–256.

[43] A. Hemani, A. Jantsch, S. Kumar, A. Postula, J. Öberg, M. Millberg, and D. Lindquist, "Network on a chip: An architecture for billion transistor era," in *Proceedings of NorChip*, 2000.

[44] G. Sander, "Vcg visualization of compiler graphs," 26 May 2013. [Online]. Available: http://rw4.cs.uni-sb.de/ sander/html/gsvcg1.html

[45] V. Paxson, "Fast lexical analyzer generator, lawrence berkeley laboratory," 26 May 2013. [Online]. Available: http://prdownloads.sourceforge.net/flex/flex-2.5.35.tar.gz

[46] "Bison - gnu parser generator," 26 May 2013. [Online]. Available: http://www.gnu.org/software/bison/

[47] F. Guderian, R. Schaffer, and G. Fettweis, "Administration- and communication-aware ip core mapping in scalable multiprocessor system-on-chips via evolutionary computing," in *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, june 2012, pp. 1–8.

[48] F. Guderian, R. Schaffer, and G. Fettweis, "Dimensioning the heterogeneous multicluster architecture via parallelism analysis and evolutionary computing," in *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, june 2012, pp. 1–8.

[49] F. Guderian, E. Fischer, M. Winter, and G. Fettweis, "Fair rate packet arbitration in network-on-chip," in *Proceedings of SOC Conference (SOCC)*, sept. 2011, pp. 278 – 283.

[50] R. Dick, "Embedded system synthesis benchmarks suite," 26 May 2013. [Online]. Available: http://ziyang.eecs.umich.edu/~dickrp/e3s/

[51] EEMBC, "The embedded microprocessor benchmark consortium," 26 May 2013. [Online]. Available: http://www.eembc.org/

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

```
/**** INCLUDE PREDEFINED FLOWS AND STEPS ****/
#include "parameter_tuning.dfl"
#include "multicluster_load_balancing.dfl"
/**** SEQUENCE IDENTIFIER DEFINITION ****/
int S_TUN = 1; int S_BAL = 2;
/**** DFFINE THE SEQUENCE OF FLOWS ****/
vector<int> flow_order;
flow_order.push_back(S_TUN);
flow_order.push_back(S_BAL);
/**** RUN CONFIGURED FLOWS ****/
for (int i=0; i<flow_order.size(); ++i) {
  string description;
  Flow eslDesignFlow;
  /**** DEFINITION OF FLOW SPECIFIC PARAMETERS
       ****/
  vector<string> arch_in, config;
  /**** SELECT FLOW CONFIGURATION ****/
  switch (flow_order.at(i)) {
    case S_TUN:
    {
      description = "Parameter Tuning";
      arch_in.push_back("lambda_tun/archs/*.
          xml");
      eslDesignFlow = tun;
      break;
    }
    case S_BAL:
    {
      description = "Multicluster Load
          Balancing";
      arch_in = getFilenames("lambda_bal/archs
          /*.xml");
      config = getFilenames("lambda_bal/
          configs/*.xml");
      eslDesignFlow = bal;
      break;
    }
    default:
    {
      println("Unknown Flow Choice: " +
          flow_order.at(i));
      continue;
    }
  }
  /**** EXECUTE SELECTED FLOW ****/
  println(description + " is running ...");
  execute(eslDesignFlow);
}
```

Listing 3. DFL source code for the flow sequence in the case study prototype.

```
/****** ALLOCATION STEP *****/
Step alloc = Step("Allocation");
vector<string> views;
views.push_back("Computation Resources");
views.push_back("Data Logistic Resources");
views.push_back("Resource Management");
alloc.add("View", views);
alloc.add("Execution", "Allocation");
alloc.add("-tool", "IPCoreMapping");
alloc.add("IPCoreMapping", "true");
string alloc_config_param = "-app_in
    lambda_tun\\apps_state_mod.xml -config
    lambda_tun\\dfConfigNoC.xml -
    arch_inlambda_tun\\arch_gen.xml -
    arch_dir_out lambda_tun\\archs -
    mappings_in lambda_tun\\mappings_ideal.xml
    ";
string alloc_static_param = " -AffinityWeight
    0.5 -star_size 2 -rows 3 -columns 3 -r 1 -
    s 50";
alloc.add("Argument", alloc_config_param +
    alloc_static_param);
/***** INPUT PARAMETER SPACE *****/
vector<int> ngen = [1000:10000:1000];
vector<int> popsize = [50:200:50];
vector<int> pmut = [0.01:0.1:0.01];
vector<int> pcross = [0.2:0.4:0.2];
vector<string> space;
space.push_back("ngen");
space.push_back("popsize");
space.push_back("pmut");
space.push_back("pcross");
alloc.add("Space", "space");
alloc.add("Strategy", "ES");
/***** PARALLEL EXECUTION *****/
string parallel = "true";
alloc.add("HPCJob", parallel);
alloc.add("workDirectory", "\\\\server\\hpc");
alloc.add("MaxCores", 15);
alloc.add("scheduler", "entmhpc3");
/***** VALIDATION STEP *****/
Step val = Step("Computation_Validation");
val.add("View", "Computation Resources");
val.add("Execution", "Validation");
val.add("-tool", "Evaluation");
val.add("Objective", "min");
val.add("Metric", "GAFitnessScore");
string val_config_param = "-mappings_dir_in
    lambda_tun\\maps -eval_out lambda_tun\\
    eval_mappings.xml";
val.add("Argument", val_config_param);
/***** FLOW CONSTRUCTION *****/
Flow tun = Flow("Parameter Tuning");
tun.add(alloc);
tun.add(val);
connect(alloc, val);
/***** FLOW VISUALIZATION *****/
tun.save("vcg", "parameter_tuning.vcg");
```

Listing 4. DFL source code for the *Parameter Tuning* flow.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

6LoWPAN Gateway System for Wireless Sensor Networks and Performance Analysis

Gopinath Rao Sinniah, Zeldi Suryady,
Usman Sarwar, Mazlan Abbas
Wireless Communication Cluster, MIMOS Berhad
Kuala Lumpur, Malaysia
{gopinath.rao, zeldi.suryady,
usman.sarwar, mazlan.abbas}@mimos.my

Sureswaran Ramadass
National Advanced IPv6 Centre of Excellence (NAv6)
Universiti Sains Malaysia (USM)
Pulau Pinang, Malaysia
sures@nav6.usm.my

***Abstract*—The importance of Wireless Sensor Network to be connected to the Internet can be observed with the emergence of Internet of Things. Applications that require WSN nodes to be connected to the Internet has been steadily increasing over the years. Knowing the fact that these low capability devices cannot handle TCP/IP protocol stack, a new format has been introduced. IPv6 over Low Power Personal Area Network (6LoWPAN) enables these devices to be connected to the Internet seamlessly and the important network device that interconnects the WSN network and the Internet is the gateway. In this paper, a gateway system that manages the packets from both the WSN and the Internet is proposed. The system ensures that WSN nodes would be IP addressable and provides end-to-end connectivity. Two types of experiments to measure the functionalities, which are to provide end-to-end connectivity and performance on latency and transmission success rate are measured. A new packet format is also proposed with the elimination of the length field from the compressed UDP header. The experiment results showed that end-to-end communication was successfully established by allocating IPv6 address to the node at the gateway. Packet transmission success rate is 100% for 1 hop scenario while latency ranges from 60 and 145 ms and it is comparable with existing prior arts that ranges from 70 ms to few minutes.**

***Index Terms*—6LoWPAN; Wireless Sensor Network; Gateway; IPv6; IEEE802.15.4.**

I. Introduction

This paper is an extension of work originally reported in The Sixth International Conference on Sensor Technologies and Applications (SENSORCOMM 2012) [1].

Wireless Sensor Network (WSN) has been increasingly being used since its introduction by DARPA in 1978. Usage of WSN gained momentum starting from early 2000 and with the cost reduced and better technology in place, more of these devices are being shipped. This is even more prevalent with the implementation of Internet of Things (IoT). Due to its hardware profile, WSN was only used in private and static network without any connectivity with other external devices. This has changed tremendously over the years. From a static type of connectivity to connectivity using web server and mobile network and now using TCP/IP protocol stack. The push for these technology is because the need and the benefits that it provides in various aspect of IoT ecosystems.

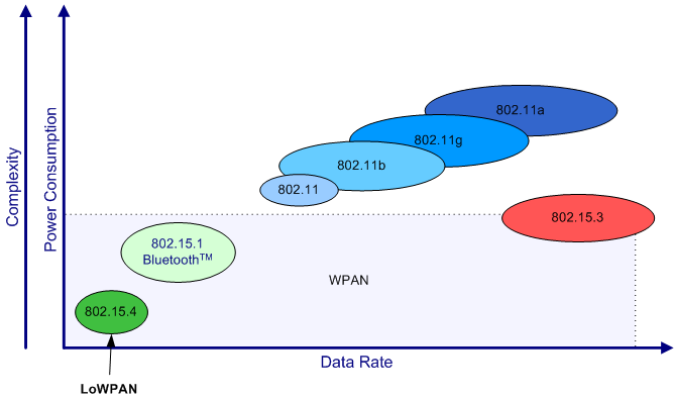


Fig. 1. Comparisons of IEEE802.15.4 with other Wireless Technologies

WSN nodes operate on low power, low processing and low memory hardware profile, which was defined in IEEE802.15.4 [2]. It is the same family of IEEE802.15 that specifies Wireless Personal Area Network (WPAN). Other standards in this family are Bluetooth (IEEE802.15.1) and High Rate WPAN (IEEE802.15.3). IEEE802.15.4 is also referred as Low Rate WPAN and has few revisions. The latest revision being standardized is IEEE802.15.4e and changes proposed in this revision are better channel hopping, which significantly increases robustness against external interference and persistent multi-path fading. IEEE 802.15.4 was designed to operate in three different bands as follows:

- 868.0 to 868.6 MHz → 1 Channel (data rates of 20 kbps, 100 kbps and 250 kbps)
- 902.0 to 928.0 MHz → 10 Channels (data rates of 40 kbps, 250 kbps)
- 2.40 to 2.48 GHz → 16 Channels (data rates of 250 kbps)

Even though there are three sets of bands for IEEE802.15.4, most of WSN implementations operate using 2.4 GHz frequency, which also being used by other standards such as WiFi and WiMAX and this leads to interference. Proper management of packet is required in WSN to reduce packet loss because of this interference. Figure 1 shows the comparison of IEEE802.15.4 standard with other wireless technologies in terms of complexities and power consumptions against data

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

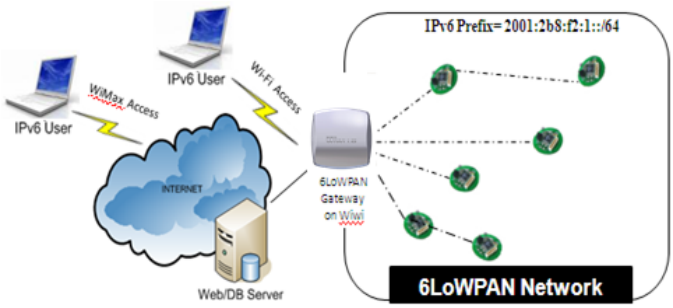


Fig. 2. Interconnection between WSN nodes and external network

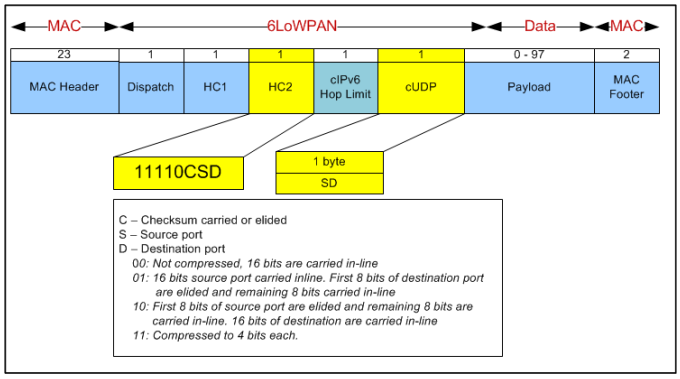


Fig. 3. Changes to 6LoWPAN header

rate.

Knowing the fact that existing TCP/IP is too bulky to be used in WSN nodes, 6LoWPAN [3] working grouping was created to provide a solution. The Working Group (WG) stated that the solution would be "pay as you use" header compression method that removes redundant or unnecessary network level information in the header. Some of the information can be derived from link-level IEEE802.15.4 header. Hence the 40 bytes IPv6 header was reduced to 2 bytes. This is achieved by reusing the link layer header information. The reduction of the header size is necessary as the total header size of IEEE802.15.4 is only 127 bytes, which is too small to accommodate the entire 40 bytes of IPv6 header.

There has been many solutions proposed to use 6LoWPAN to enable end-to-end communications between WSN nodes and external devices. All the communication from WSN nodes have to be through the gateway that interfaces between the WSN and external network. To enable the support for 6LoWPAN type of communication, a new system has to be developed on the gateway so that WSN nodes can be reachable in Internet and at the same time provides better performances. The gateway must be able to read all the three types of addressing format available in 6LoWPAN and also support other features such as routing, mobility, security and others. This paper extends the work provided in [1] and add contributions to [4], by providing detail gateway systems and performance analysis. The communication between the 6LoWPAN nodes and the external network through a gateway is given in Figure 2.

The main contribution of this paper are as follows:

a. Providing a detail 6LoWPAN gateway system that provides end-to-end communication between low power embedded wireless devices and external IPv6 devices.
b. A data management system on the gateway to handle packets that arrives both from external network and from Wireless Sensor Network, which results in increase of successful transmission of packets from WSN nodes and reduction in latency.

The rest of this article is organized as follows: Section II presents a new gateway system to handle communication from 6LoWPAN nodes. Section III provides the implementations and experiments to evaluate the performances, while Section IV discusses the results obtained. Section V presents the existing solutions related to WSN and specifically using 6LoWPAN. This paper concludes in Section VI with some suggestions for further research.

II. 6LoWPAN Gateway System

Architecture for extending the WSN to the Internet is presented by outlining the gateway that interfaces between the WSN and the Internet. By assigning IPv6 addresses and with proper handling of the packets, WSN nodes able to extend their reachability to the Internet and also supports two-way communications. The designed gateway system supports all the three addressing mechanisms available in the 6LoWPAN stack. The three addressing schemes are the short address (16 bits), MAC address (64 bits) and IPv6 address (128 bits). However, in our proposed solution, only 64-bit MAC address is used. This is because the 128-bit IPv6 address is too large to be used in IEEE802.15.4 header and 16-bit short address is not unique for WSN nodes identification. In this paper, only UDP type of packets are considered for experiments. Since the original 6LoWPAN header is not changed, the non-UDP packets will be treated as defined in the standards [20].

A. 6LoWPAN Header

Header Compression 2 (HC2) [19] [20], is a one byte field to define if UDP header need to be compressed or not. Bits 0 through 4 represent the next header ID and '11110' indicates the specific UDP header compression encoding. The 5th bit represent if checksum required or to be elided. Last 2 bits are used to define the source and destination ports. The header format is given in Figure 3. 16 bits of each used for source and destination port can be reduced to 4 bits each by eliminating the first 12 bits. With this, the compressed UDP header is only 1 byte, which is for the 4 bits source and destination ports each.

The 6LoWPAN header format used is given in Figure 4.

B. Extending WSN to Internet

The gateway is designed to support two standards of communication:

- **Pull based communication method** - IPv6 clients request data from sensor nodes in 6LoWPAN network. This

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

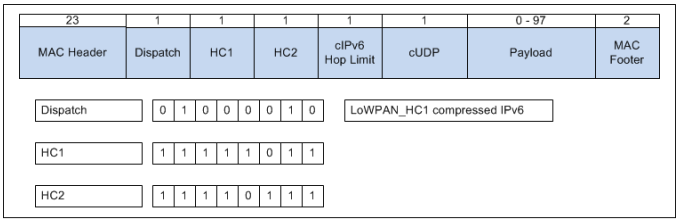


Fig. 4. 6LoWPAN Header Format used

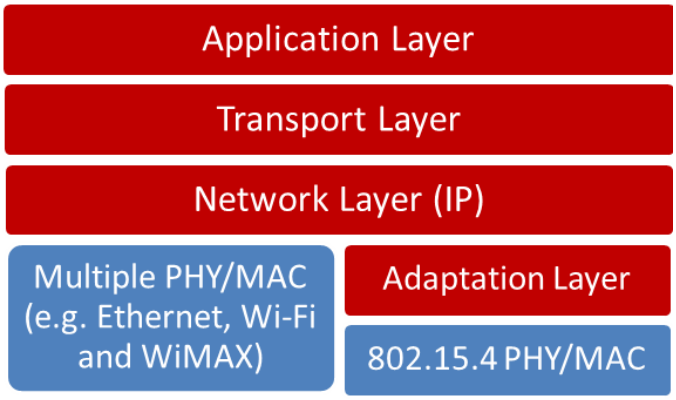


Fig. 5. Dual Stack Protocol in 6LoWPAN Gateway

is two-way communication, between client and sensor node.

- **Push based communication method** - Sensor nodes periodically send sensed data to a particular IPv6 client in IPv6 Network. The IPv6 client in this system is normally just like a remote station or database server. This is one-way communication, from sensor node to a remote station.

The 6LoWPAN gateway system aims at providing communication system and mechanism for ubiquitous wireless sensor network. The system is build by combining IEEE802.15.4 connectivity with standard interface to the Internet such as Wi-Fi, WiMAX, and Ethernet. The gateway must have dual stack protocol as shown in Figure 5 that represents multiple PHY/MAC (e.g., Ethernet, Wi-Fi, and WiMAX) for connecting external IP network and PHY/MAC of 6LoWPAN (IEEE 802.15.4).

Using the dual stack protocol, the gateway is designed to have 3 modules, which are:

- **6LoWPAN (WSN) Module** - This module consists of IEEE802.15.4 compliance hardware, which has the 6LoWPAN stack on it. The module is responsible for handling connectivity and data transmission of 6LoWPAN network using IEEE802.15.4 standard. Packets send by the sensor nodes are captured by this module and forwarded to the service module for further processing. It also forwards packets received from the service module to the sensor nodes.
- **External Interface Module** - This module defines the Physical and MAC layer of any interface that provides connectivity to external IP network. Therefore, the role of this module is to offer functionalities required to ensure connectivity to external IP network. Some of the interfaces may provide connectivity to LAN/Wireless LAN (e.g., Wi-Fi), while others can provide connectivity to back-haul internet (e.g., Ethernet or WiMAX). In case of gateway having multiple accesses, the selection of the interface depends on the priority configured in the service module and it could be changed manually.
- **Service Module** - This module provide services to handle both 6LoWPAN and IPv6 packets. It is a significant module that bridges all the interfaces that connects to different networks. Since most of the main processes occur in this module, the service module has a very important responsibility, which is integrating the 6LoWPAN network with the IP network through other external interfaces. The main purpose of this module is to provide functionalities for handling standard IPv6 packet from external network as well as 6LoWPAN packet. Two sub-modules are created to make this happen. The first sub-module is the node management that collects and stores all the necessary information of the sensor nodes. Some of the information stored are the MAC address of the sensor node, correspondence IPv6 address, and others. The second sub-module is for packet handling and translation. It handles both 6LoWPAN packets and IPv6 packets. The two types of transmission, which are pull-based and push-based are identified by the port number the packets are transmitted. The two sub-modules capture any IPv6 packet as well as 6LoWPAN packet, analyse the source and destination address and process accordingly.

C. 6LoWPAN Gateway System Components

There are many components within gateway that are important for the end-to-end system to work properly. This paper focuses on the packet management within the Gateway. Two main components in the gateway system, which are the focus of this paper that are used so that the packets are properly translated and forwarded are:

- **Node Management** - consists of Node Discovery (ND), Periodical Logger, Mapping Table and Predefined IPv6 Prefix and Address Translation.
- **Packet Handling and Translation** - consists of IPv6 Packet Handler, IPv6 - 6LoWPAN Packet Transformation and Predefined Remote Station Address.

The Node Discovery is a service that discovers the list of node as well as informing the nodes in 6LoWPAN network about their gateway. Both the gateway and the WSN nodes must have the Node Discovery module. The Node Discovery can be active or passive. For the active Node Discovery, the gateway will periodically broadcast Gateway Advertisement (GW_ADV) packet through IEEE802.15.4 interface to 6LoWPAN network. The nodes will response to this GW_ADV message with advertisement response (ADV_RESPONSE). Using

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

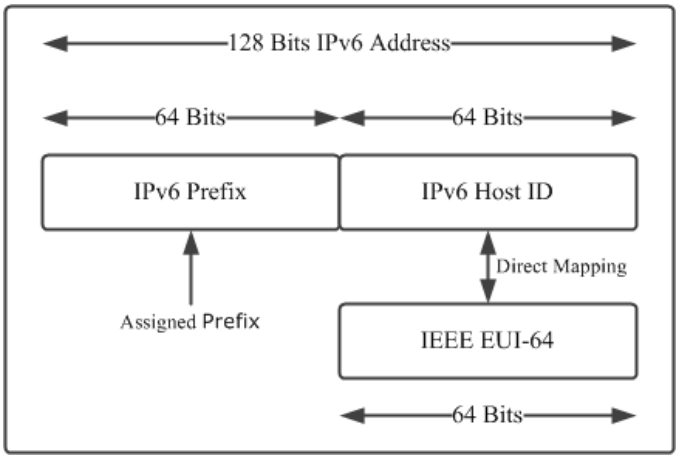


Fig. 6. IPv6 Address Assignment to 6LoWPAN Nodes

this option, the gateway can retrieve information of any sensor nodes available within 6LoWPAN network. Moreover, the nodes will also know their gateway that interfaces with the external IP network. The process of gateway sending GW_ADV message and node response with ADV_RESPONSE message is called Network Join Process. In addition, the MAC addresses of the 6LoWPAN nodes are retrieved from the ADV_RESPONSE message and stored in the Mapping Table. Thus, the Mapping Table for address translation will be generated from the network join process.

The translation is executed after the gateway receives the ADV_RESPONSE by adding a predefine 64 bit IPv6 prefix to a MAC address (EUI-64 bit) of a sensor node, which is retrieved from the MAC header of 6LoWPAN packet. Using this approach, the gateway manages the pseudo IPv6 address of the sensor node. Therefore, the gateway can ignore the process of sending out prefix advertisement to the 6LoWPAN network. This process provides some benefits.

- Message overhead would be reduced as prefix is not sent to the nodes
- Nodes would not process prefix configuration and hence power is not used unnecessarily
- Nodes does not have to allocate memory to configure the IPv6 address

The EUI-64 identifier of a 6LoWPAN device can be used as the interface identifier of the IPv6 address while the predefine IPv6 prefix is used as network identifier. Since the EUI-64 addresses are globally unique and appending it to IPv6 prefix to generate IPv6 addresses are globally unique as well. Figure 6 shows the address translation process and Table I illustrates the mapping table maintained by the gateway after the translation process.

D. Operation and Communication of 6LoWPAN Gateway

To give clear understanding on the practical use of this system using pull based mechanism, Figure 7 provide detail network time diagram.

TABLE I
EXAMPLE OF SENSOR NODE MAPPING TABLE

IPv6 Address	EUI-64 MAC Address
2001:2B8:F2:1:**7E23:1200:20:1200**	**7E:23:12:00:00:20:12:00**
2001:2B8:F2:1:**7D10:400:206:1501**	**7D:10:04:00:02:06:15:01**

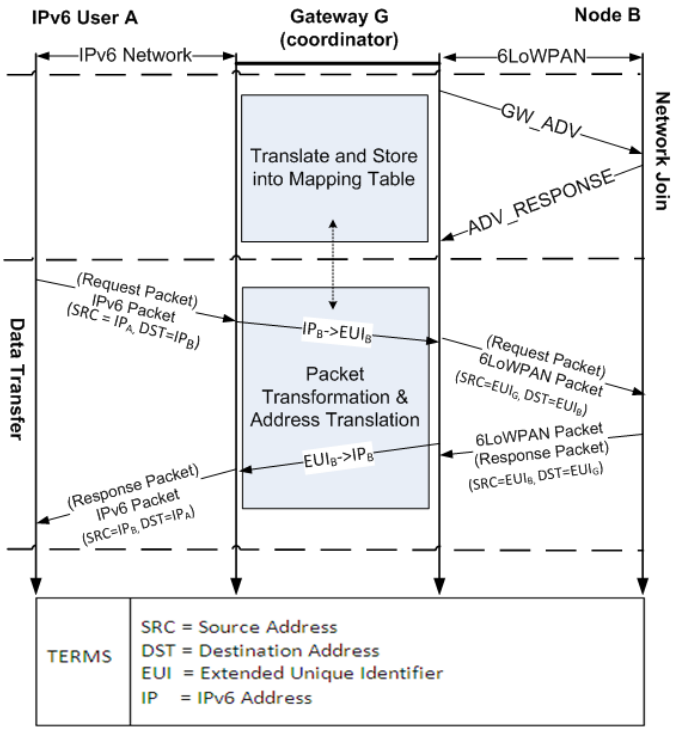


Fig. 7. Pull-Based Communications

Nodes in the wireless sensor network are first need to be registered in the gateway by following network join process explained earlier. This process is executed only at the beginning of network setup and periodically thereafter. This is similar to standard IPv6 neighbor discovery (ND) [18], wherein the advertisements from routers are sent periodically. The periodic time is set at larger intervals to reduce message overhead hence reduces the power consumption for processing the messages. Following are the steps taken in the network join process as given in Figure 7:

- Gateway G conduct node discovery by issuing a GW_ADVERTISEMENT message to the 6LoWPAN network.
- Node B, upon receiving this message, responds with ADV_RESPONSE message indicating that it will join the network.
- Gateway G will update the table with the information of the nodes responded

New nodes that join the network can update their presence using network join message, NET_JOIN message. Nodes can send this message if they did not receive any gateway discovery message from the gateway after a predefined time.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE II
DETAIL INFORMATION IN ADDRESS INFORMATION TABLE

FIELD	LENGTH	DESCRIPTION
ID	1 byte	The requesting packet sequence number
Source Address	16 bytes	IPv6 address of the user (client)
Destination (Sensor Node) MAC Address	8 bytes	The MAC address (EUI-64 bit) of the sensor node. The address is derived by removing the IPv6 prefix from the sensors IPv6 address
Port	2 bytes	Port number allocated (from 61616 - 61630)
Status	1 byte	0: Packet has been forwarded to 6LoWPAN node. 1: Packet has been forwarded to IPv6 client. 2: Pending because the destination address is the as previous packet, which has not been received the response from the node.

The communications for both push based and pull based schemes are maintained through the use of a gateway. Different port numbers are used to differentiate the sensor's traffic for both the schemes. RFC 4944 [19] defines a well-known port range (61616-61631) for UDP packet in 6LoWPAN. In this implementation, the ports used are as follows:

- Port 61616 is used by the gateway to send data to the sensor nodes in pull based mechanism.
- Port 61617 is used by the gateway to receive data from sensor nodes in pull based mechanism.
- Port 61630 is used by the nodes to receive the request from the external node through the gateway and response using the same port.
- Port 61631 is used at the gateway to receive data from sensor nodes in push based method.

For both the communication mechanisms, Gateway maintains an Address Information Table as given in Table II. The gateway can differentiate traffic to the specific nodes that uses the ports defined and traffic from other applications. This is by referring to the table that has been created to store all the nodes that would use these ports. If there are other applications that uses different ports, the system would then operate as defined in the standard.

One of the examples of polling wireless sensor data is using one to one Communication.

This communication scenario occurs whenever different IPv6 clients request data from different sensor nodes. As an example, as shown in Figure 8, 2 IPv6 clients and 2 sensor nodes connected through a Gateway are used. Each IPv6 client requests data from different sensor nodes in 6LoWPAN network.

Based on Figure 8, upon receiving the IPv6 packet requests from an IPv6 client, the gateway will execute Forwarding Process for each packet:

i. Gateway updates the entry for the Address Information Table (Table III) by storing the Destination MAC Address field in the table, which is derived by removing the IPv6 prefix (2003:2b8:f2:1) used for sensor nodes. The gateway do not keep IPv6 destination address (sensor's IPv6 address) since the address can be generated by adding the prefix (e.g., 2003:2b8:f3:1) with EUI-64 address.

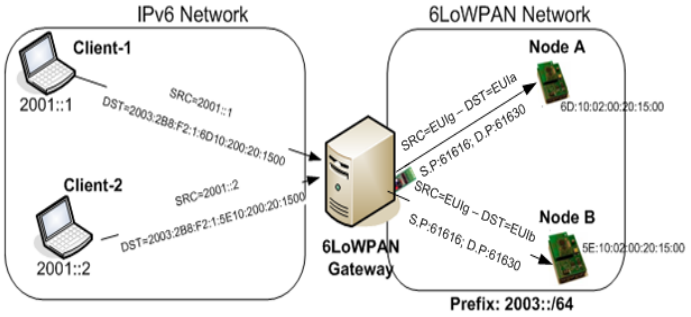


Fig. 8. IPv6 Client requesting data from sensor node

TABLE III
ADDRESS INFORMATION TABLE UPON RECEIVING REQUESTS FROM AN IPV6 CLIENT

ID	IPV6 SOURCE ADDRESS	DESTINATION MAC ADDRESS	PORT	STATUS
1	IP_1 (2001::1)	6D:10:02:00:20:15:00	61616	0
2	IP_2 (2001::2)	5E:10:02:00:20:15:00	61616	0

ii. The gateway checks the destination address (EUI-64 address). If there is an earlier request for data from the same address (status = 0) then the new request is queued by setting the status to 2.

iii. Once the packet is allocated with a source port, the gateway proceed to transform/convert the IPv6 packets to a 6LoWPAN packet:
   a. The gateway uses port number 61616 as source port. Port number 61630 is used for destination port at sensor node.
   b. Use the derived EUI-64 bit MAC address as destination address.

iv. The gateway forwards 6LoWPAN packet to 6LoWPAN network.

While processing any request packets, the gateway is ready for the reply from the sensor node. The Response Process for each response/reply packet from a sensor node is as follows:

i. The reply packet from sensor node will be sent to the port number 61617 of the gateway (Figure 9).

ii. The gateway will wait the reply for a certain amount of time (e.g., 1000 ms); if the gateway does not receive any reply, a second request message would be sent. If the gateway still did not receive any reply after that, it will send the *Time-Out Message* to the IPv6 client.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

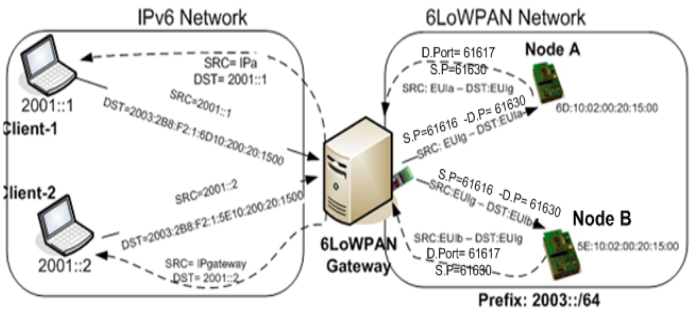


Fig. 9. Communication after receiving response from Sensor Node

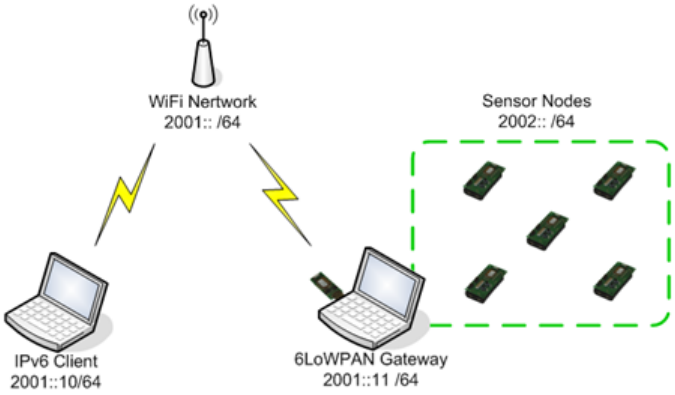


Fig. 10. Testbed for the System

TABLE IV
ADDRESS INFORMATION TABLE AFTER SENDING THE PACKET BACK TO IPV6 CLIENT

ID	IPV6 SOURCE ADDRESS	DESTINATION MAC ADDRESS	PORT	STATUS
1	IP_1 (2001::1)	6D:10:02:00:20:15:00	61617	1
2	IP_2 (2001::2)	5E:10:02:00:20:15:00	61617	1

iii. After the gateway receives a reply from the sensor node, it checks the Address Information Table, and matches the EUI-64 source address of the reply packet in order to retrieve IPv6 address of the client (IPv6 Source Address). The IPv6 source address will be used as the destination address to route back the packet to IPv6 client.

iv. Next, the 6LoWPAN packet is converted to IPv6 packet and route it back to IPv6 client.

v. The Status Field in the table is set 1, meaning the reply packet from sensor node already forwarded to IPv6 client (Table IV) and the entry in the Address Information Table will be deleted.

III. IMPLEMENTATION AND TESTING

A testbed was created to validate the gateway architecture and to measure the end-to-end performance as shown in Figure 10.

The setup consists of nano router and sensor nodes developed by Sensinode Inc. [21] as our hardware platform. Gateway is a laptop computer with Linux OS and has three interfaces; a nano router for the wireless sensor network, WiFi network interface and Ethernet network interface that connects to the IPv6 network. Nano router is a USB device that is attached to one of the available USB port in the gateway. Packet Handler module explained earlier is configured and executed on the gateway. The sensor nodes are installed with the free real-time operating system (FreeRTOS) with the NanoStack software module, which consists of 6LoWPAN stack with added features. Each of the sensor node has 2 AA batteries. The modules were developed using c programming language. The communications for both push based and pull based schemes are maintained through the use of a gateway. The sensor nodes that were deployed provide readings for temperature and light intensity measurements.

TABLE V
PERFORMANCE MEASUREMENT PROPERTIES

Properties	Details
Network Size	4-8 nodes for 1 hop away. 2x2, 2x4 and 2x6 for 2 hops
Distance	3 meters for each hop
Data Sampling intervals	20 seconds
Duration	120 samples (1 hour)
Message size	4, 8, 16, 37 bytes
Measurements	Transmission Success Rate and Latency
Method	Start with 1 node and gradually increase the nodes while sending data simultaneously

A client laptop was also used to retrieve sensor data to verify the bidirectional communication. To validate the performance, tests with different settings were conducted with different data sizes. Furthermore, to test the bidirectional communication, a ping message was sent from the gateway and using the reply, the latency was calculated. Table V provides the properties for the tests.

In 2 hops network environments, the end sensor nodes are configured to forward data through a particular relay node. In the experiments conducted, the sensor nodes are divided equally among the relay nodes. In 2x2 network setup, 1 sensor node forwards data through 1 relay node, in 2x4 network setup, 2 sensor nodes forward data through 1 relay node and in 2x6 network setup, 3 sensor nodes forward data through 1 relay node.

Wireless sensor nodes used in the experiments are configured with the following features:

- The sensor nodes are static (no mobility)
- The nodes are configured without any sleeping schedule hence the nodes will always be active to send and receive data
- Nodes are configured to forward the packets to the gate-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

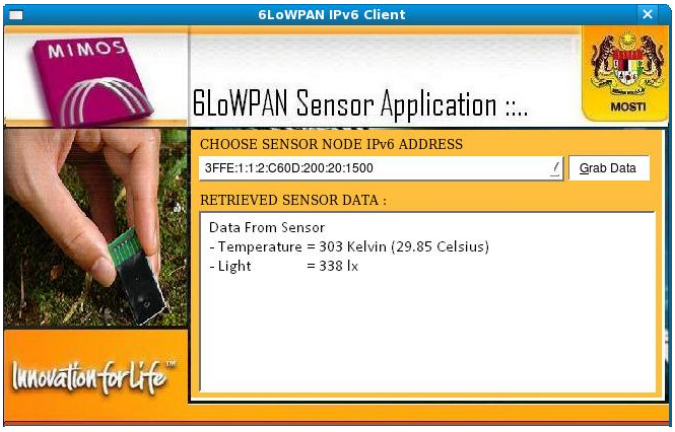


Fig. 11. IPv6 client application to read data directly from sensors. ©2009 MIMOS Bhd. All Rights Reserved

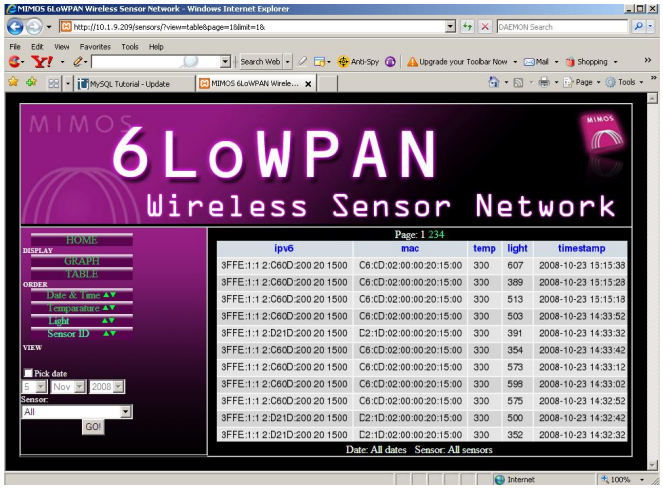


Fig. 12. Display sensor information using web browser. ©2009 MIMOS Bhd. All Rights Reserved

way through a relay node in a 2 hops static deployments

IV. System Performance Evaluation

As described earlier, the request from a client will be forwarded by the gateway using a simple client as shown in Figure 11 [4]. All the sensor nodes' IPv6 addresses are listed in the client and when a particular IPv6 address is selected, a request is forwarded to the gateway, which will then do the necessary actions. The temperature and light reading from the sensor will then displayed on the client. This shows the success of bidirectional communication (Pull based mechanism). In the push based mechanism, the data is periodically sent to a web server and the data is displayed using a web browser as shown in Figure 12.

End-to-end latency usually measured using the ping command by getting the round trip time (RTT). The one way latency is half of the RTT value. There are few components that contributes to the end-to-end latency as given below.

- Processing of the packets - This latency is due to the processing power available at both end nodes. Request packet sent from the application layer has to move to the physical layer so with low processing at the node increases the latency but this is usually minimal.
- Network processing - In a typical network environment, the packets traverse through many routers and processing of packets at the router further increases the latency. The queueing delay is under this category. This happens when a gateway receives multiple packets from different sources heading towards the same destination. This problem is tackled using the Packet Management Module on the gateway.
- Network condition - The network condition is usually unpredictable hence if the network is congested, the packets that travel will get delayed and further increases the latency. Latency value is even more if the packet is sent in a wireless environment. In wireless multi hop environment, inefficient quality of service also effects the latency.

The end-to-end latency when only 1 sensor node is active is measured and average latency is 64.7 milliseconds for 1 hop and 94.1 milliseconds for 2 hops. This average latency is comparable with average latency claimed in the white paper by IPSO-Alliance [22], which is about 125 milliseconds. Total latency is calculated based on the processing latency of packet at the node, processing latency at the network gateway or router and latency due to network condition.

The latency for various data sizes for only 1 node active are given in Figure 13. It can be observed that the increase of data size does not effect much on the latency. This is because the packets are not fragmented and data is sent in one packet. However, the latency increases with the increase of number of hops. This is because of the effect of network condition that is explained earlier. In 2 hop network setup, the packets have to send to the relay node before being sent to the gateway. Processing of packets at the relay node further adds the latency. Figure 14 and Figure 15 show the average for 1 hop and 2 hops. These results are used as a base for the other experiments.

In Figure 16 and Figure 17, it can be observed that as the nodes increased one by one, the latency is also increases. This is because the nodes started sending data every 15 seconds after associating with the gateway. The increase in latency is due to the network condition component explained earlier, which is the increase of active nodes in wireless network increases the latency.

As for the packet delivery rate, 100% success rate obtained for nodes 1 hop away from the gateway for all the scenarios. However, the percentage dropped with the increase of hops and nodes as shown in Figure 18. This is due to the condition of the WSN network and the relay node are not configured with proper packet handling.

To further validate that the system developed is better in terms of packet delivery rate, two experiments, which are without the data management module were executed. The graphs for packet delivery rate for 1 hop and 2 hops without Packet Management Module proposed are given Figure 19 and

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

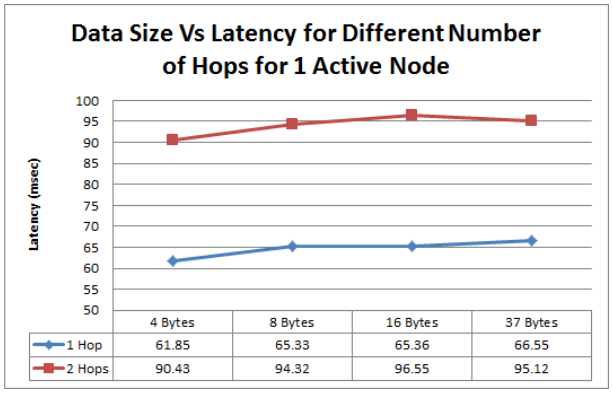


Fig. 13. Data Size Vs Latency for Different Hops for 1 Active Node

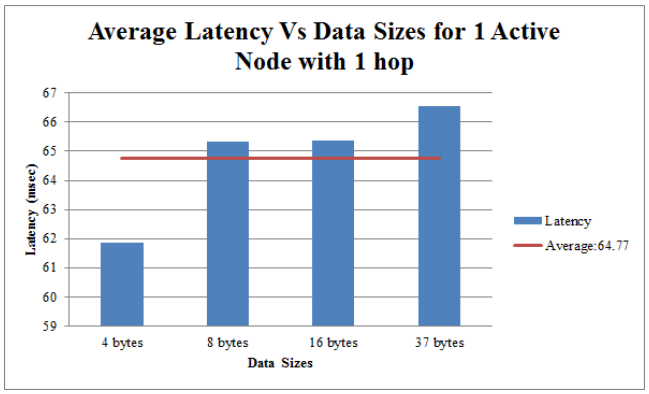


Fig. 14. Average Latency Vs Data Sizes for 1 Active Node with 1 hop

Figure 20.

It can be observed from Figure 19 that there is significant drop of packet delivery rate from 99% for 4 actives nodes and 4 bytes of data to 89% for 8 active nodes and 37 bytes of data. System with the data management module gives 100% packet delivery rate. This is because the gateway without the data management module receives packets from different nodes at the same port and could not handle the packets properly.

Since the same relay node was used, the packet delivery

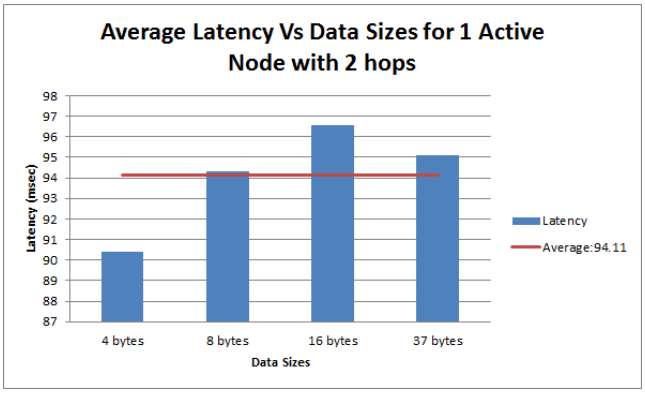


Fig. 15. Average Latency for 1 node in 2 hops with difference data sizes

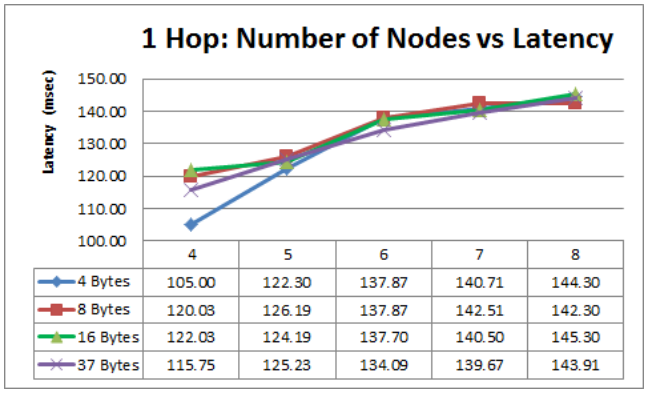


Fig. 16. Number of Nodes Vs Latency in 1 Hop Network Environment

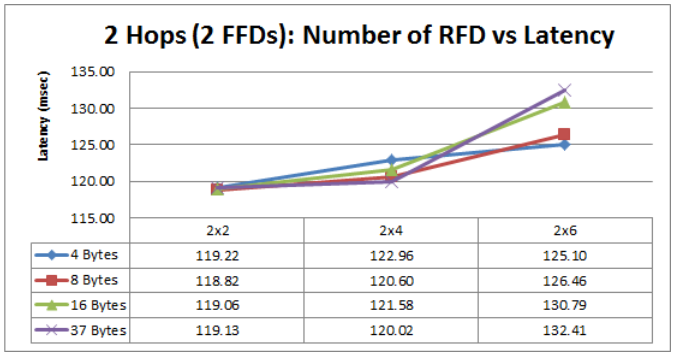


Fig. 17. Number of Nodes Vs Latency in 2 Hops Network Environment

rate dropped with the similar margin as in 1 hop compared to the use of data management module. This is because the relay node is not configured with data management. The drop further increases because no data management at the gateway. The packet drop for 1 hop ranges from 1% to 13% whereas for 2 hops, the packet drop ranges from 2% and 4%.

This shows the importance of packet management at the gateway. With the proper address and data management, packet delivery rate for wireless sensor node can be improved.

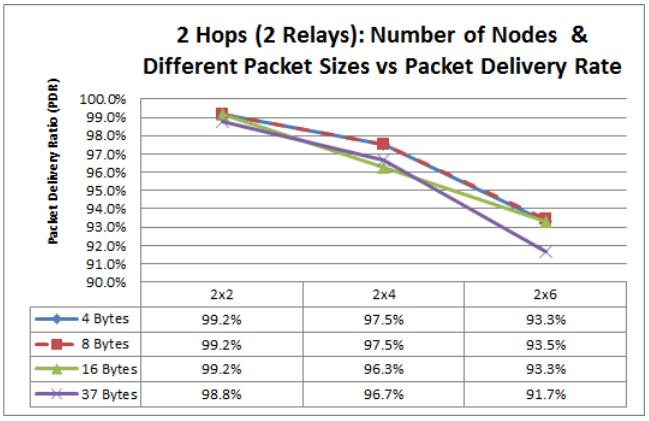
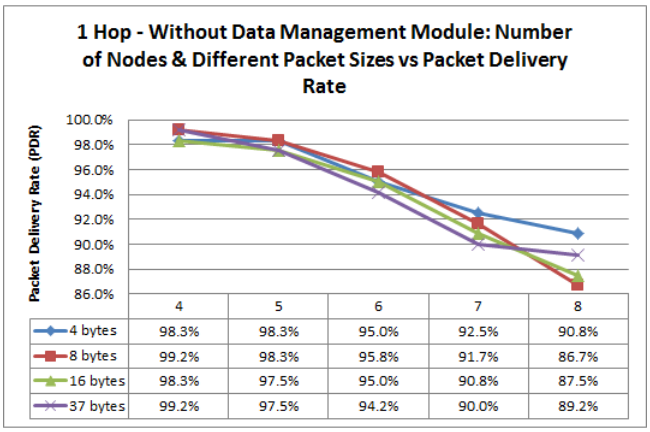


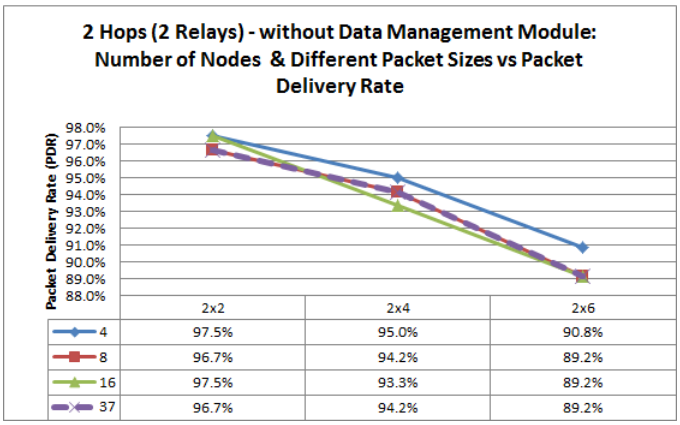
Fig. 18. Number of Nodes Vs Packet Delivery Rate in 2 Hops Network Environment

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



	4	5	6	7	8
4 bytes	98.3%	98.3%	95.0%	92.5%	90.8%
8 bytes	99.2%	98.3%	95.8%	91.7%	86.7%
16 bytes	98.3%	97.5%	95.0%	90.8%	87.5%
37 bytes	99.2%	97.5%	94.2%	90.0%	89.2%

Fig. 19. Number of Nodes Vs Packet Delivery Rate in 1 Hop Network Environment without Data Management Module



	2x2	2x4	2x6
4	97.5%	95.0%	90.8%
8	96.7%	94.2%	89.2%
16	97.5%	93.3%	89.2%
37	96.7%	94.2%	89.2%

Fig. 20. Number of Nodes Vs Packet Delivery Rate in 2 Hops Network Environment without Data Management Module

V. WSN Gateway Related Work

There are several gateway architectures that were proposed for various implementation scenarios. Initially WSN was deployed in isolated network because of the constraint of the devices and technologies. With the progress of technology, data from WSN nodes was collected by the collector and send to a centralized server using GSM network or long range radio. With the introduction of web communication, data can be displayed in the web server but the data is still collected by the collector. Now, this devices have better processing capability and the need for the nodes to be connected to external network are more prevalent so IP connectivity has been suggested. This is possible with the use of gateway for communication to external network.

The systems are grouped based on the trend specified and each system is described by their research contributions and implementations. Three type of connectivity methods are discussed with the emphasis given to the architecture used to implement and method of managing the packets at the gateway. Some systems are deployed using the proprietary protocol such as ZigBee while others using open source such as TinyOS, Contiki and Nanostack.

A. Gateway to server type of connectivity

In this method, data from sensor nodes is sent to a gateway, which then forwards to a server. Gateway may have different type of connectivity to the server such as GSM, GPRS or others. In the system developed by Steenkamp et al. at Cape Penisular University of Technology [5], WSN gateway was developed using TinyOS with AT91RM9200 ARM evaluation kit from Atmel. This gateway enables users to remotely retrieve data from WSN network using GSM network.

A system specifically designed to gather information from the forest was proposed by Wenbin et al. [6]. In this system, gateway is connected to an external server using GPRS module. The gateway collects sensor data and converts it into Comma-Separated Value (CSV) format. After that the CSV file is sent to the server using FTP via GPRS module. Communication between the gateway and the FTP server is established using TCP/IP protocol that was built-in in the Debian Linux for embedded devices. Another system developed using GPRS module was implemented by Tolle et al. [7]. Data from the sensor nodes was collected over Mica2 node attached to RS232 serial, stored in a local database, and then transmitted over a GPRS cellular modem to an off-site database. They implemented the system to capture the microclimate surrounding a coastal redwood tree.

A different approach was introduced by Becher et al. [8] to send health information to a personal computer. In this approach, person's health data such as ECG, pulse rate and body weight are sent to a gateway, which then forwards to a personal computer using Bluetooth technology. ZigBee communication was used between the WSN nodes and the gateway.

The systems and architectures given, uses a point-to-point communication between the gateway and the server using GPRS, Blueetooth, long range wireless or Satellite. The drawbacks with these system are:

- There is no data management at the gateway. Data is collected at the gateway, saved in a file and send to the other end. In some cases, data is forwarded as it arrived at the gateway. These systems are not suitable for critical applications because all the systems have a single point of failure. There is also no mechanism to determine if the data was successfully sent to its destination.
- It has no IP connectivity for the nodes as such end-to-end communication could not be performed. The data from the sensor is send to the gateway, which then forwards it to a data storage server. Information about the node's ID would be added in the data field and this cost more overhead. Besides that, data would be sent to a single end point like the system that uses Bluetooth [8] and not routable in the Internet. IP address given to each node would enable the nodes to be reachable from anywhere as.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

B. Communication using Web Services

Systems developed using this method uses web services to publish the collected data. The web service may be running in a separate server or part of gateway. In a system proposed by Qiu and Choi [9], the information from the sensor nodes are displayed using web server. The approach taken was to setup a web server in the gateway itself using embedded Common Gateway Interface (CGI) technology. Users can check the data from ZigBee sensor network through the web-sensor gateway. Users can get data from a particular sensor by sending a request through the web server at the gateway. The gateway, after receiving the information using the ZigBee protocol, displays the information on the web server for the client to view.

Fan Ye Dun et al. [10] presents a gateway, which connects WSN with external network. In this gateway that uses TinyOS, data is gathered at the gateway and stored in the local storage using embedded database, SQLite3. The information in the database is displayed using a web service so that any external user can access and view the data using existing TCP/IP protocol. Overall architecture presented is similar to system [9] discussed earlier. This system was used for environmental monitoring. The limitation with this approach is similar to the earlier system, which inhibits the end-to-end communication that is important in some applications. Maybe the system is only suitable for environmental monitoring and not for other use cases. Another similar system that displayed data using web services was proposed by Dan et al. [11]. The data are stored in Extensible Markup Language (XML) files according to information types. Web service interface within the gateway encapsulates XML format data in Simple Object Access Packet (SOAP) packet and transmitted to web browser through HTTP protocol. Similar concept was also used in a system developed by Jin et al. [12] for home and building automation and by Malatras et al. [13] for facility management.

Some of the drawbacks with the systems discussed in this category are:

- Nodes cannot be reachable directly from the external network. This limits the goal of providing direct communication and limits the growth of WSN in other aspects such as mobility, etc. The gateway requires extra resources as it also provides web services and data storage. In some systems, IPv4 address was assigned based on the availability. This further inhibits the growth of the network.
- There were no proper data management at the gateway. It is not necessary for this group because most of the time, it is communication between the sensor nodes and the gateway.

C. End-to-end connectivity

In this system, WSN nodes somehow are able to connect to external network using few methods. Zimmermann et al. [14] developed a system using a combination of DNS reverse lookup and address translation method to extend WSN node to external network. In this system, the sensor nodes are configured with IPv6 link local address. Each of the nodes is mapped to a global IPv6 address using the 1 on 1 Network Address Translation (NAT) mapping. Whenever a node wants to communicate with an external device, it is assumed that the node knows the domain name of the external device and sends the query for IPv6 address. The gateway forwards the query while maintaining the requester's information in its database. The Domain Name System (DNS) Application level gateway intercepts the query and replaces the domain name with global IPv6 address of the external device. The global address is mapped to a newly generated link local address using the 1 on 1 NAT mapping at the gateway. The limitations with this system are:

- If the DNS query is not intercepted for some reasons and the DNS server is heavily used, IPv6 address cannot be returned to the sensor node. This will fail the communication between the sensor node and the external device.
- There is no management of the packets at the gateway besides the 1 to 1 mapping. Using link local address adds extra overhead on the node. This can be reduced by reusing the MAC address already available in the header.
- This approach also uses extra overhead which consists of messages being exchanged to retrieve the external device's IPv6 address and this contributes to the increase of transmission latency.
- Both the sensor nodes and the external nodes have link local and global address, which is translated at the gateway using 1 on 1 NAT. The translation of the header increases processing of the packet and it is unnecessary.

An IP address translation mechanism was proposed by Choi et al. [15]. It is assumed that the gateway has records of all the external devices' IPv6 addresses. WSN node request the destination IPv6 address from the gateway by providing the link local address of the external device. Once the node receives the information, the node will then send the packet using EUI64 MAC address of the destination node. The gateway again change the MAC destination address to link local address. Even though the objective for this approach was to provide end-to-end communication, the approach taken was not practical.

- It is not practical for internal node to request address of the external device based on the link local address. In this implementation the gateway has to store all external devices' addresses, which is impossible. This is practical if the node sends data to a known address such as a server.
- Extra overhead and redundant message exchanges between the node and gateway. The node queries the gateway for destination ID by providing the destination link local address address. The destination node ID can actually be retrieved from the link local address used in the query. Furthermore, link local address is not routable in Internet thus it restricts the implementation to a particular local area network.
- There is no method mentioned on the management of data at the gateway. This would be a problem as in some

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

scenario, when nodes continuously and simultaneously transmit data to the gateway and without a proper management mechanism, packet loss will be high.

Since IPv6 is not fully deployed, Chang et al. [16] proposed and implemented a system using 6LoWPAN in IPv4 network. They propose that both public and private IPv4 address be used for the nodes in WSN. Connectivity from gateway to external network could be using Network Address Translator-Protocol Translator (NAT-PT), tunneling service such as ISATAP, Teredo, 6to4 and others.

ZigBee has been widely used in WSN and changing the protocol stack to support IPv6 is not practical hence Chia et al. [17] proposed an architecture using SIP protocol to interconnect ZigBee network with the external network. With this session layer approach, both ZigBee and 6LoWPAN WSN would be supported. For ZigBee nodes, the ZigBee Apps information is translated into SIP while SIP has to be supported in 6LoWPAN node. This extra layer service creates more overhead for 6LoWPAN nodes. End-to-end communication is not supported with this method and the architecture does not provide data management at the gateway.

There was various methods proposed to connect WSN to the Internet but none of it described in detail the method of end-to-end connectivity and does not provide data management at the gateway. Both the end-to-end connectivity and the data management are important features to be incorporated in WSN to ensure that data will be communicated effectively like any other Internet devices.

VI. Conclusion

This paper proposed a gateway system to interconnect wireless sensor network with external network using 6LoWPAN protocol. The gateway provides a mechanism for the end clients to directly communicate with the sensor node, which was assigned with IPv6 address. Besides that, the gateway forwards the periodical data to a web server.

The system is validated with the successful transmission of sensor data, which was displayed using a client and web server. Further tests were conducted to validate the latency and the transmission success rate. The latency for 1 hop with various number of nodes ranges between 60 to 145 milliseconds while the transmission success rate is 100 % for 1 hop. The success rate dropped with the increase of number of hop, which could be because of the relay node (FFD) not forwarding the packets appropriately. Nevertheless, the results are in accordance with the other prior art. It is expected that further increase in the number of hops would reduce the packet transmission success rate.

As for future work, the proposed solution can be further tested in other environments by setting different transmission intervals, less interferences, etc. It is also important for the transmission to be extended with more than 2 hops with minimal packet drops. The performance can also be evaluated with the implementation of other components such as security, routing, dynamic topology and mobility with multi-hop scenarios. Besides that, the gateway can be extended to be used as IoT gateway that will provide seamless connectivity to various standards and devices.

References

[1] Gopinath Rao. S, Zeldi Suryady, Usman Sarwar, Mazlan Abbas, and Sureswaran Ramadass, "IPv6 Wireless Sensor Network Gateway Design and End-to-End Performance Analysis", SENSORCOMM 2012, The Sixth International Conference on Sensor Technologies and Applications, held in Rome, Italy, pp. 67-72, August 19-24, 2012.

[2] Institute of Electrical and Electronics Engineers (IEEE), "IEEE 802.15.4." http://standards.ieee.org/about/get/802/802.15.html

[3] IPv6 over Low Power Personal Area Network (6LoWPAN) IETF Working Group. Retrieved: July, 2012. http://datatracker.ietf.org/wg/6lowpan/

[4] G. R. Sinniah, Z. Suryady, U. Sarwar, and M. Abbas, "A Gateway Solution for IPv6 Wireless Sensor Network", Ultra Modern Telecommunication & Workshops, St. Petersburg, Russia, pp. 1-6, October 2009.

[5] Steenkamp, L. and Kaplan, S. and Wilkinson, R.H., "Wireless sensor network gateway", The 9th IEEE AFRICON 2009, pp. 1-6, September 2009.

[6] Li Wenbin, Cui Dongxu, and Zhang Junguo, "Design and Implementation of Wireless Sensor Network Gateway Faced to Forest Information Monitor", 2010 International Conference on Intelligent System Design and Engineering Application (ISDEA), pp. 524-526, October 2010.

[7] Gilman Tolle, Joseph Polastre, Robert Szewczyk, David Culler, Neil Turner, Kevin Tu, Stephen Burgess, Todd Dawson, Phil Buonadonna, David Gay, and Wei Hong, "A macroscope in the redwoods", Proceedings of the 3rd international conference on Embedded networked sensor systems, SenSys '05, San Diego, USA, pp. 51-63 2005.

[8] K. Becher, C.P. Figueiredo, C. Mu andhle, R. Ruff, P.M. Mendes, and K. Hoffmann, "Design and realization of a wireless sensor gateway for health monitoring", 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Buenos Aires, Argentina, pp. 374-377 August 31 - September 4 2010.

[9] Peng Qiu, Ung Heo, and Jaeho Choi, "The web-sensor gateway architecture for ZigBee", IEEE 13th International Symposium on Consumer Electronics, ISCE '09, Kyoto, Japan, pp. 661-664, May 25-28 2009.

[10] Ye Dun-fan, Min Liang-liang, and Wang Wei, "Design and Implementation of Wireless Sensor Network Gateway Based on Environmental Monitoring", International Conference on Environmental Science and Information Application Technology, ESIAT 2009, Wuhan, China, pp.289-292, 4-5 July 2009.

[11] Dan Hu, Shi-Ning Li, and Zhi-Gang Li, "Design and Implementation of Wireless Sensor Network Gateway Based on Web Services", 4th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM '08, Dalian, China, pp. 1-4, 12-14 October 2008.

[12] Jin Seok Oh, Jeong Il Choi, Hyun Seok Lee, and Jeong Seok Heo, "Web-based real-time sensor monitoring system using Smart Client", International Forum on Strategic Technology, IFOST 2007, Ulaanbaatar, Mongolia, pp. 619-622, 3 - 6 October 2007.

[13] A. Malatras, A. Asgari, and T. Bauge, "Web Enabled Wireless Sensor Networks for Facilities Management", IEEE Systems Journal, pp. 500-512, December 2008.

[14] A. Zimmermann, J. Sa Silva, J.B.M Sobral, and F. Boavida, " 6GLAD: IPv6 Global to Link-layer Address Translation for 6LoWPAN Overhead Reducing", 4th EURO-NGI Conference on Next Generation Internet Networks, NGI 2008, Krakow, Poland, pp. 209-214, 28-30 April 2008.

[15] Dae-In Choi, Jong-tak Park, Su-yoen Kim, and H.K. Kahng, "IPv6 global connectivity for 6LoWPAN using short ID", 2011 International Conference on Information Networking, (ICOIN), Kuala Lumpur, Malaysia, pp. 384-387, 26-28 January 2011.

[16] Chang-Yeol Yum, Yong Sung Beun, Sunmoo Kang, Young Ro Lee, and Joo Seok Song, "Methods to use 6LoWPAN in IPv4 network", The 9th International Conference on Advanced Communication Technology, (ICACT), Gangwon-Do, Republic of Korea, pp. 969-972, 12-14 February, 2007.

[17] Chia-Wen Lu, Shu-Cheng Li, and Q. Wu, "Interconnecting ZigBee and 6LoWPAN wireless sensor networks for smart grid applications", The Fifth International Conference on Sensing Technology (ICST), Palmerston North, New Zealand, pp. 269-272, 28 November- 1 December 2011.

[18] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, RFC 4861: Neighbor Discovery for IP version 6 (IPv6), IETF Standard Document, September 2007.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[19] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, RFC 4944: Transmission of IPv6 Packets over IEEE 802.15.4 Networks, IETF Standard Document, September 2007.

[20] J. Hui and P. Thubert. RFC 6282: Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks, IETF Standard Document, September 2011.

[21] "Sensinode hardware and NanoStack Operating System", 2008. Retrieved: July, 2012. Available: http://www.sensinode.com/

[22] J. Abeill, M. Durvy, J. Hui, and S. Dawson-Haggerty, "Lightweight IPv6 Stacks for Smart Objects: the Experience of Three Independent and Interoperable Implementations", November 2008. Available at: http://www.ipsoalliance.org/white-papers

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Silicon Photomultiplier: Technology Improvement and Performance

Roberto Pagano, Sebania Libertino, Domenico Corso, and Salvatore Lombardo
Istituto per la Microelettronica e Microsistemi
CNR
VIII Strada Z.I. 5, 95121, Catania, Italy
roberto.pagano@imm.cnr.it,
sebania.libertino@imm.cnr.it,
domenico.corso@imm.cnr.it,
salvatore.lombardo@imm.cnr.it

Giuseppina Valvo, Delfo Sanfilippo, Giovanni. Condorelli, Massimo Mazzillo, Angelo Piana, Beatrice Carbone, and Giorgio Fallica
Sensor Design Group
STMicroelectronics
Stradale Primosole, 50 95121 Catania, Italy
giusy.valvo@st.com, delfo.sanfilippo@st.com,
giovanni.condorelli@st.com, massimo.mazzillo@st.com,
angelo.piana@st.com, beatrice.carbone@st.com,
giorgio.fallica@st.com

Abstract— **Our main results on the study of both single pixels and Silicon Photomultiplier arrays produced by STMicroelectronics in Catania are reviewed. Our data, coupled to an extensive simulation study, show that the single pixel technology is close to its ultimate physical limit. The distribution of dark current in large arrays follows a Poissonian law. Cross talk effects are strongly reduced by the presence of optical trenches surrounding each pixel of the array. Finally, we demonstrate that these devices can also be used as single photon counters also without a complex amplification stage.**

Keywords – Silicon Photomultiplier; dark count; trenches.

I. INTRODUCTION

The ability to detect single photons represents the ultimate goal in optical detection. To achieve such sensitivity a number of technologies have been developed and refined to suit particular applications. These include: Photomultiplier Tubes (PMTs), Microchannel Plate Photomultiplier Tubes (MCPMT), Hybrid Photon Detector (HPD), p-i-n photodiodes, linear and Geiger mode Avalanche Photo Diodes (APDs), etc. [1], [2], [3], [4], [5]. The need for ever more sensitive, compact, rugged, and inexpensive optical sensors in the visible region of the spectrum continues today, and it is particularly acute in the fields of the biological sciences, medicine, astronomy, and high energy physics. Applications such as fluorescence and luminescence photometry, absorption spectroscopy, scintillation readout, light detection and ranging, and quantum cryptography require extremely sensitive optical sensors often in adverse environments, such as high magnetic fields, and where space is limited.

In many of these applications, the PMT has become since mid-1930's the detector of choice almost without a convincing alternative. However, PMT presents some disadvantages: it is fragile, it requires high operating voltage (higher than 100 V), and it can not operate without a shielding protection in magnetic environment. Since its inception in the 1980's, the so-called Silicon Photomultiplier (SiPM) has begun to rival the PMT in many of its parameters such as gain, photon detection efficiency, and timing [6], [7], [8], [9], [10], [11]. The SiPM has all the additional benefits of silicon technology such as compactness, reliability, ruggedness, high volume of production, and long term stability. Although all the previous motivations would be sufficient to explore this alternative to PMT, it is the low production cost of silicon technology that attracts the most and has led to the efforts that finally has enabled the realization of this photodetector.

The SiPM major drawback is the relatively large dark current [11], due to the combination of a diffusion current produced at the quasi-neutral regions at the boundaries of the device active region and of generation of carriers due to point defects and/or metallic impurities in the active area depletion layer emitting carriers through the Schockley-Hall-Read (SHR) mechanisms, eventually boosted by the Poole-Frenkel effect [12].

In this paper, after a detailed discussion on the principle of operation of SiPMs presented in Section II, two different pixel design technologies of SiPM developed by STMicroelectronics are discussed. They consist in a n^+ on p silicon structure. The device active part is the same in both pixels. They differ in the doping of the epitaxial layer and in the starting substrate (n-type Si for the first device and p-type Si for the second device). The differences between the two technologies lead to significant differences in the dark count rate (*DC*) measured at temperatures higher than 10°C. In Section III the two device structures, the experimental setup, and the simulation environment used are discussed. The current–voltage characteristics in forward and reverse bias of the two pixels for temperatures ranging from -25°C to 65°C are presented and discussed comparing the measured data with electrical simulations in Section IV. Moreover, electrical and optical performance of SiPM devices suitable for large scale fabrication in a VLSI production line are reviewed in Section V. Finally, the conclusions are outlined in Section VII.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

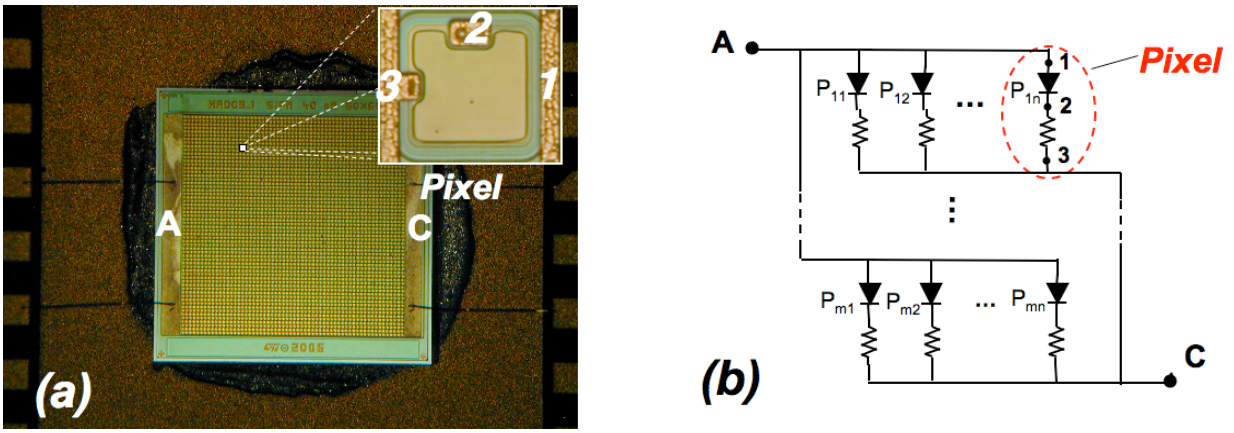


Figure 1. (a) Microphotograph of a 64×64 pixels SiPM with 6.5 mm^2 active area produced by STMicroelectronics.The pixel, shown in the inset, has an active area of 40 µm × 40 µm. (b) Schematic circuit diagram of a SiPM with n×m pixels. The pixel, enclosed by the dashed line, is composed of an SPAD and a quenching resistor. Note that node 1, 2 and 3 are the same in Fig. (a) and (b).

II. PRINCIPLE OF OPERATION

The principle of operation of SiPMs is inspired to the demand of information on the exact determination of the arrival time and density of a very low photon flux. Due to the quantum nature of light, a low photon flux is composed by few photons distributed in time and space. A dense array of space distributed micro-devices (the pixels), individually able to detect the arrival time of a single photon can, in principle, resolve the time and the space distribution of the impinging photons. This is the basic SiPM operation principle. In a SiPM the pixels are electrically connected in parallel forming a matrix of n× m adjacent sensors (see Fig. 1*a*). Each pixel, known in literature as Single Photon Avalanche Diode (SPAD) or Geiger Mode Avalanche Photodiode (GM-APD) [13], [14], consists of a p-n junction suitably doped in order to have avalanche breakdown in a well defined active area with an integrated quenching resistance in series, as shown in the schematic picture of Fig. 1*b*. The active area is formed by creating an enriched well zone, generally doped by ion implantation followed by thermal processing for dopant activation and defect annealing. This dopant local enrichment generates regions where the vertical junction electric field is higher, and these become the pixel active areas for photon detection [15]. The p-n junction devices are operated in Geiger mode, that is, they are biased above the junction breakdown voltage (*BV*). Above *BV* the device can stay in a quiescent state for a relative long time, up to ms, depending on the technology quality process (low defect density) and the operating condition (temperature and voltage) [16]. Then, when the device is in quiescent state, the active volume (active area times the depleted region) is characterized by an electric field well above the breakdown field. However, the p-n junction does not go into avalanche breakdown. In such condition, the absorption of a single photon in the active volume will trigger, through the generation of an electron-hole pair, the onset of the junction avalanche with a probability depending on the operating voltage [17]. A macroscopic current pulse flows through the junction resulting in a strong amplification of the single photon arrival. The amplification value, usually indicated as Gain (*G*), is above 10^6 electrons per pulse. This large gain is the cause of the strong SPAD sensitivity.

The avalanche process is a self-sustaining process and to quench it the integration of a resistor in each pixel is required. In our device the resistor is connected in series to the cathode of the p-n junction (see Fig. 1). The quenching mechanism introduced by the resistor acts as follow: the avalanche following the photon absorption causes a rapid increase of the current flowing through the p-n junction as well as through the external resistor. The voltage drop across the series resistor decreases the voltage applied to the p-n junction below the *BV*, forcing the avalanche quenching and the consequent extinction of the current flux. Once the avalanche is quenched, a recharge time is required to restore the pixel to the original condition of electric field above *BV*, making the pixel ready to the detection of a new photon [18]. Therefore, the detection of a photon by a single pixel results in a current pulse, which can then be easily measured by an external circuit. However, a single pixel works as a digital photon sensor, that is, it can not detect multiple photon arrival. This task is accomplished by the full array. In fact, the current detected by the overall SiPM is simply the sum of the currents produced by the various pixels. Hence, this device compared to the original design of the SPAD has the advantage of having a response, which is in a relatively large dynamic range, proportional to the flux of photons impinging on the detector at the same time [6], [7], [8], [9], [10], [11]. The SiPM Gain is about 10^6, similar to the one of single pixel.

The capability to measure low photon fluxes is limited by the device *DC*. In fact, the single pixel can have a breakdown event even if it does not detect a photon because of thermal generation of electron-hole pairs within the depletion region assisted by defects (SHR mechanism) and / or of minority carrier diffusion from the depletion region boundaries. Such events result in current spikes having the same features of the "real" counts due to photon arrival. This determines a lower limit to the photon count rate. A high performance SiPM

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

must have a very low *DC* rate. Typical value is in the order of 1MHz/mm^2 at room temperature.

Another limiting factor to the device operation is the cross-talk effect. The cross-talk is a noise contribution common to all pixelated devices. A pulse current produced by a pixel, due to a photon detection event or to a primary dark noise event, can induce one or more adjacent pixels to experience the avalanche breakdown. Then, the corresponding output pulse current of the SiPM has an amplitude peak proportional to the sum of the pixels involved in the single photo-detection and in the correlated cross-talk phenomena. This noise contribution is detrimental for all the applications where a single photon resolution is required.

The cross-talk noise has two different physical origins: optical and electrical. The optical cross-talk is due to the photons generation by radiative emission from the hot carriers produced during an avalanche discharge. In an avalanche multiplication process, on average 3 photons, with energy higher than the silicon band gap (1.12eV), are emitted every 10^5 carriers [19]. These emitted photons can travel to a neighboring pixel and trigger a breakdown there.

The electrical cross-talk can occur when carriers, generated during the avalanche breakdown in a pixel, can travel along the epitaxial layer, common to all pixels, and reach the neighboring pixels triggering there a new avalanche breakdown [20]. Some strategies have been studied to reduce the cross-talk between neighboring pixels. The first is to increment the distance between adjacent pixels. This approach has a detrimental effect on the geometrical fill factor of the SiPM. The second strategy consists in fabricating grooves, filled with optical absorbing material, all around each pixel. These grooves, commonly named trenches, prevent from optical and electrical coupling between pixels. The reduction of the geometrical fill factor with such design is mild while the effect on the cross-talk noise is considerable. The devices studied in this paper are fabricated using the second approach and the beneficial effects provided by the trench presence are discussed in Section V-A.

An important feature of SiPM, as one would expect from a semiconductor device, it is the long term stability of its parameters (*BV*, *G*, *DC,* etc.). In many applications, in fact, the variation of such parameters in the course of time may be a problematic issue. As reported by other authors SiPMs have no aging and show stable parameters even if exposed to elevated temperature for long time [10], [21], [22].

III. EXPERIMENTAL

Electrical characterization was performed at wafer level using a Cascade Microtech Probe Station *11000*. The samples were cooled using a Temptronic *TPO 3200A* ThermoChuck that provides a stable temperature, within 0.1°, between -60°C and 200°C. Current *vs.* voltage measurements (I-V) were acquired using an *HP4156B* precision semiconductor parameter analyzer with an integration time of 1s. The *DC* and the gain were measured using a Tektronix *DPO 7104* Digital Oscilloscope (OSC) with 1 GHz bandwidth and 20 Gsa/s measuring the voltage drop through a 50 Ω resistor connected between the cathode of the pixel and the ground. The I-V characteristics have been measured on more than 30 devices, for both single pixels and SiPM arrays of both technologies.

Optical measurements were carried out using a *Cube* laser diode with a wavelength of 659 nm by Coherent working from continuous wave to 6 ns pulses. Modulation was achieved using an external trigger (Agilent *81110A* 165/330 MHz). The device was biased and the signal acquired using the source-meter and oscilloscope already mentioned.

A. Device structures

In this work, two different technologies are compared. They differ for few, but important, characteristics. The full device fabrication details can be found in [23]. In this paper, we want to focus our attention on similarities and

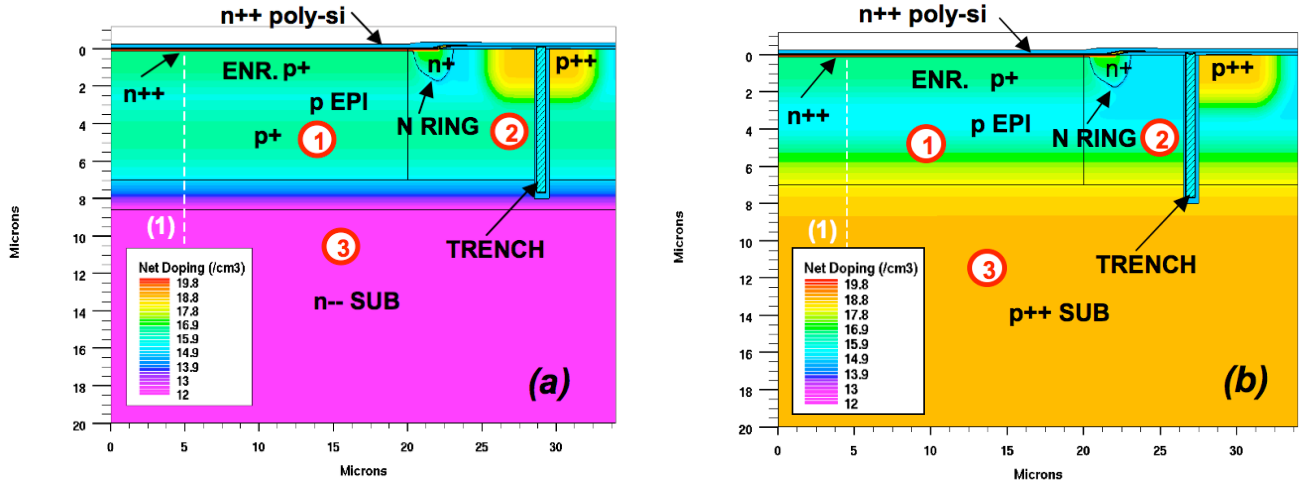


Figure 2. Schematic cross-section of a SiPM pixel. (a) Device 1: double epitaxial layer, n-substrate, trenches crossing sinker diffusion (b) Device 2: single epitaxial layer, p-substrate, isolated sinker diffusion.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

differences between the two technologies. They have the same device active part and guard ring, fabricated as discussed in [23], the same *BV* (-28.2±0.3V at 25°C) and the same active area, of 40×40 µm^2.

Fig. 2 shows a half cross section of the two studied technologies, Fig. 2*a* and 2*b* for device 1 and 2, respectively. The main differences between the two technologies are the doping of the epitaxial layer and the starting substrate. In the first technology, device 1, a double epitaxial layer, first p^+ layer, then followed by a p-type Si, is grown on a low doped n-type (100) oriented Si substrate. In the second technology, device 2, only a single p-type epitaxial Si layer is grown on a highly doped p^+-type (100) Si substrate. In both cases, deep optical trenches are fabricated for the optical and electrical isolation between the pixels [24]. In device 2 the optical trench is closer to the active region than in device 1 (see Fig. 2). As a result, the thermal diffusion of the p^{++} implanted sinker needed for the anode contact, is shielded by the trenches.

In both devices, the same anti-reflecting coating, poly-silicon quenching resistance and metal contact are integrated as discussed in [23]. These elements are not included in the cross sections of Fig 2.

Fig. 2 also shows three regions enumerated as 1, 2 and 3: region 1 is the central epitaxial layer below the active area of the pixel (0 µm < x < 20 µm and 0 µm < y < 7 µm for both devices); region 2 is the border epitaxial region of the pixel (x > 20 µm and 0 µm < y < 7 µm) and region 3 is the substrate (y > 7 µm). They have been identified for simulation purposes. It allows to define trap energy and lifetime separately for each region, to better simulate the real structure.

Fig. 3 shows the comparison between the data (symbols) and the simulated results (lines) of the final net doping along the cut line 1 (dashed with lines in Fig. 2) for device 1 (blue solid line) and device 2 (red dashed line). The experimental doping profile was obtained by the spreading resistance measurements for both device 1 (blue squares) and device 2 (red circles). The simulated profiles follow quite well the experimental data. It is important to stress that the profiles of the two devices overlaps in the full active region. The main differences are in the region below 2 µm. Part of it is highlighted in Fig. 3 (EPI).

The structural differences described so far are at the base of the electrical behavior shown in the following sections.

B. Simulation parameters

Electrical simulations were obtained using a 2-D drift-diffusion solver developed by Silvaco Co. LTD [25].

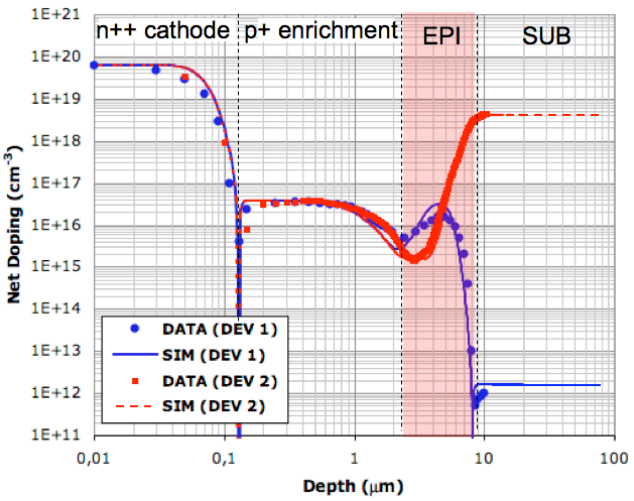


Figure 3. Doping profile of device 1 (blue) and device 2 (red) experimentally measured (symbols) and simulated (lines).

The adopted model is the drift-diffusion approximation including standard SHR generation/recombination, Auger recombination, band gap narrowing, Coulomb scattering, and SHR surface recombination. The parameters of the TCAD simulations are those connected to the minority carrier lifetime in the three device regions above described. The SHR generation (*G*) / recombination (*R*) adopted model consists of the following equations:

$$G - R = \frac{pn - n_{ie}^2}{\tau_p \left[n + n_{ie} \exp\left(\frac{E_T}{kT}\right)\right] + \tau_n \left[p + n_{ie} \exp\left(-\frac{E_T}{kT}\right)\right]} \quad (1)$$

where:

$$\tau_n = \frac{\tau_{n0}}{1 + N_D / N_{\mathrm{Re}f}}, \tau_p = \frac{\tau_{p0}}{1 + N_D / N_{\mathrm{Re}f}} \quad (2)$$

where p, n, and n_{ie} are the hole, electron, and intrinsic carrier concentration, E_T and kT are the trap and the thermal energy, N_D is the local dopant concentration, and N_{Ref}=5×10^{6}cm^{-3}. We have assumed that τ_{n0}=τ_{p0}=τ_0. In the model we assume three different values for τ_0 and E_T in the three above defined device regions, i.e., τ_{01}, τ_{02}, τ_{03}, E_{T1}, E_{T2}, and E_{T3} resumed in Table I. In the same table are also resumed the experimental activation energies discussed in Section IV-B.

TABLE I. SIMULATION PARAMETERS AND EXPERIMENTAL ACTIVATION ENERGIES

Device	Simulation							Experimental	
	τ_{01} (s)	τ_{02} (s)	τ_{03} (s)	E_{T1} (eV)*	E_{T2} (eV)*	E_{T3} (eV)*	SR (cm/s)	E_{a1} (eV)	E_{a2} (eV)
1	10^{-3}	10^{-5}	10^{-5}	0.06	0.2	0	100	0.57	1.18
2	10^{-3}	10^{-5}	10^{-3}	0.06	0.2	0	150	0.59	1.12

* Energies values are with respect to the midgap of the Si energies bandgap.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

IV. SINGLE PIXEL TECHNOLOGY

In this section, the experimental data of the two single pixel technologies described previously are discussed and compared with the electrical simulation for both forward and reverse bias.

A. Forward current

In this paragraph, the pixel forward regime will be discussed. The study of the pixel behavior in forward, even if this is not the regime for photon-detection, is functional to understand the causes leading the differences in the *DC* of the two devices. Moreover, the simulations presented in this paper are the results of the best fit obtained from forward and reverse currents at different temperatures and for different geometries as discussed in the following.

The forward current for both devices has a dominant component at the perimeter of the active area. This effect has been observed in pixels of both types having different active areas (A_{ACT}) and dead areas (A_{DEAD}). The dead area of a pixel is the area surrounding the active region as shown in the inset of Fig. 4. In the same figure, the projection of the measured I-V in the ideal diode regime to the y axis (V=0) (symbol), i.e, the pre-factor I_0 of the Schockley diode equation [26], is reported for pixels with three different A_{ACT} and four different A_{DEAD} compared with simulations (dashed line). The data clearly show that I_0 is almost independent from A_{ACT} and has strong dependence on A_{DEAD}. This geometrical information has been taken into account in the electrical simulation defining the physical parameters discussed in Section III-B. Moreover, a surface recombination model [27], with velocity S_R at the boundary between silicon and oxide, is included in region 2. The final parameters, almost the same for both devices, are summarized in Table I.

Such high difference between the carrier lifetime (electron and hole) in region 1 and in region 2 produces a preferential current path at the perimeter of the p-n junction, as suggested by the experimental data. This effect is clearly visible in Fig. 5 that shows the 2D distribution of the total current density (J_{tot}) at 25°C and for a forward bias of 0.3V

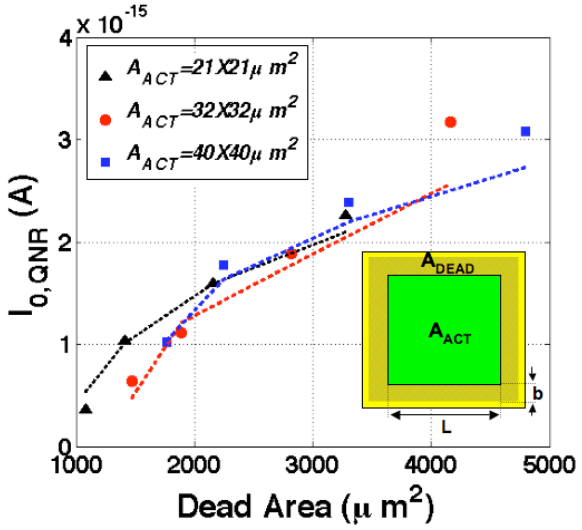


Figure 4. Measured (symbols) and simulated (dotted line) I_0 as function of the active area and the dead area for device 1 @ 25°C. The inset is a plane view of a pixel showing the active area and the dead area.

in both device 1 and 2 (Fig 5 *a* and *b*, respectively). The dashed circle in Fig. 5 highlights the interested region.

Fig. 6 shows the measured I-V (symbols) of device 1 (Fig. 6*a*) and device 2 (Fig. 6*b*) at three different temperatures: -25°C (circles), 25°C (triangles) and 65°C (squares). The measured data are compared with the simulated I-V (dashed line). Two regimes can be clearly observed: the ideal diode following the Schockley law at low voltages (linear region) and the resistive regime due to the integrated quenching resistor R_Q at higher voltages. Actually, the current of device 1 at high voltages deviates from the simulated current (in the range 0.4V - 0.5V depending on temperature). This is due to a parasitic Schotky diode at the anode contact that has been removed in device 2. In the simulation this effect has not been considered. The effect of the R_Q was simulated including an ideal resistor at the cathode contact equal as the measured value in both devices (220 kΩ at 25°C).

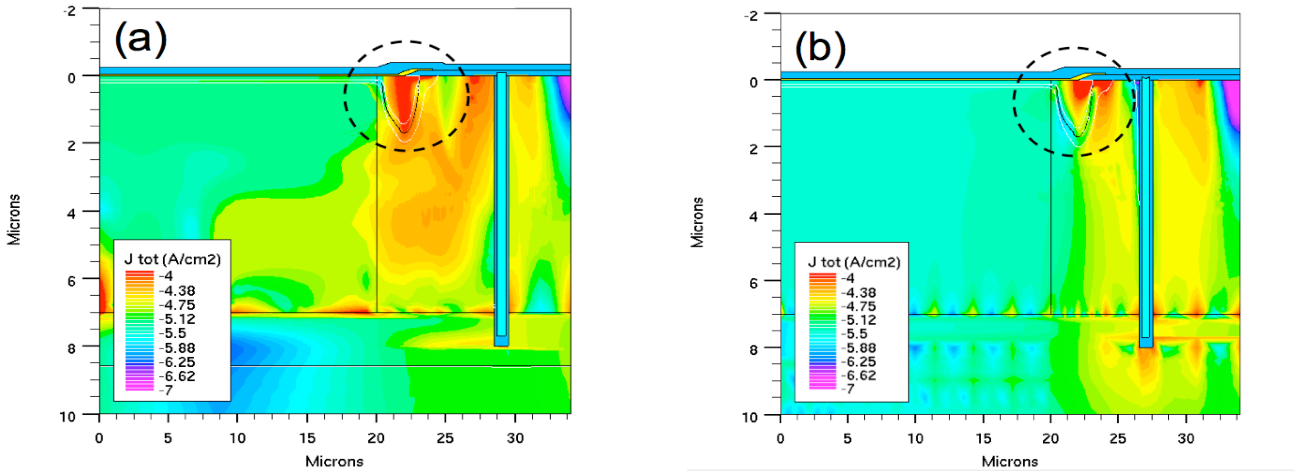


Figure 5. Distribution of the total current density (Jtot) at 25°C and at V=0.3V of (a) device 1 and (b) device 2.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

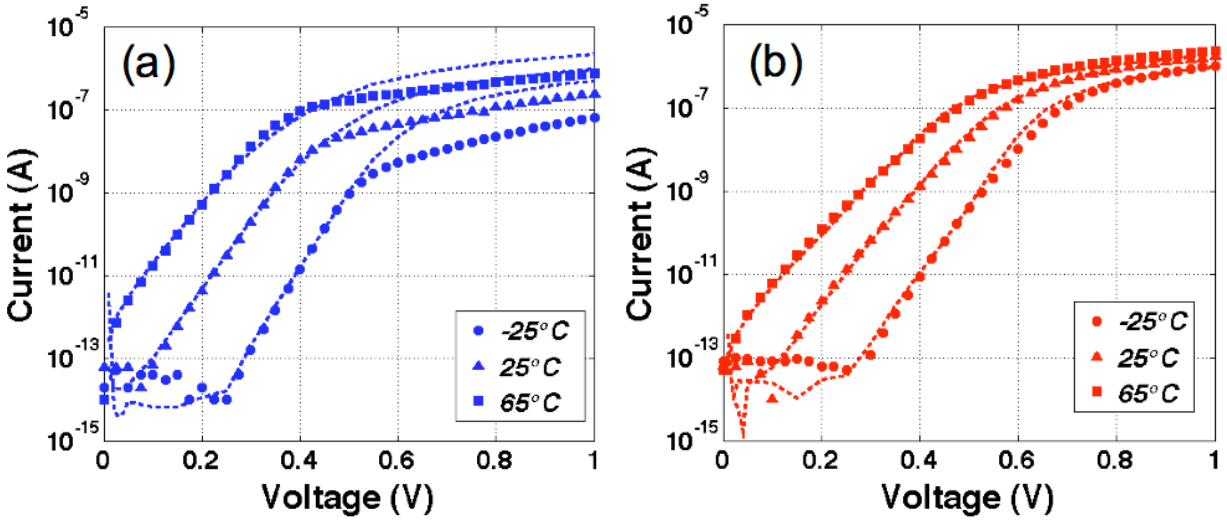


Figure 6. Measured (symbols) and simulated (dotted line) IV in forward polarization at three temperatures of (a) Device 1 and (b) Device 2.

The simulation shows a very good agreement with the experimental data for both devices and deviates only in device 1 as just discussed. The simulations have been carried out using the parameters summarized in Table I.

B. Reverse current and Dark Count

Fig. 7 shows the dark currents as a function of voltage at three different temperatures, -25°C (circles), 25°C (triangles) and 65°C (squares) for a single pixel belonging to device 1 technology (blue symbols) compared to the dark current of a pixel with the structure of device 2 (red symbols). The *BV* of the two pixels is the same, -28.2 V at 25°C, with a temperature coefficient of -29mV/°C.

The leakage currents, i.e., the currents at voltage below the *BV*, are nearly the same for the two kind of devices in the full range of temperature, ~10pA at 25°C. However, the currents at voltage above *BV* increase with a different rate with respect to the temperature. At -25°C the dark currents (circles) are roughly the same while, by increasing the temperature, they show remarkable differences. At 25°C and voltages of -32V (+ 3.8V over-voltage, *OV*) the dark current in the device 1 is one order of magnitude higher than that of the device 2. At 65°C the difference increases approaching two orders of magnitude. When the pixel works as photon detector it is biased above breakdown and the dark currents define the lower limit to the photon rate detectable. The understanding of the physical origin of these currents is an important achievement to improve the device technology. Although the measurements show steady-state I-V curves, the time resolved analysis of the current at the oscilloscope, at a fixed bias above *BV*, reveals that the time averaged current of Fig. 7 is a random sequence of current spikes.

Fig. 8 shows a trace of a single pixel dark current at *OV*=+3.8V, at 25°C in a time window of 1ms acquired with the OSC. Five current spikes with ~90 µA amplitude randomly distributed in time are clearly visible. The

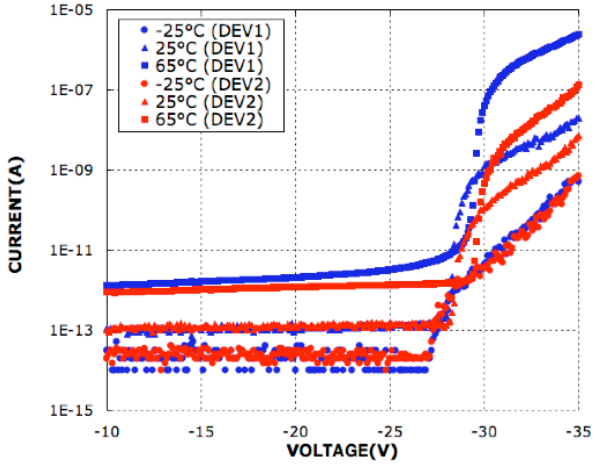


Figure 7. I-V in dark and in reverse bias at -25°C (circle), 25°C (triangle) and 65°C (square) of device 1 (blue) and of device 2 (red).

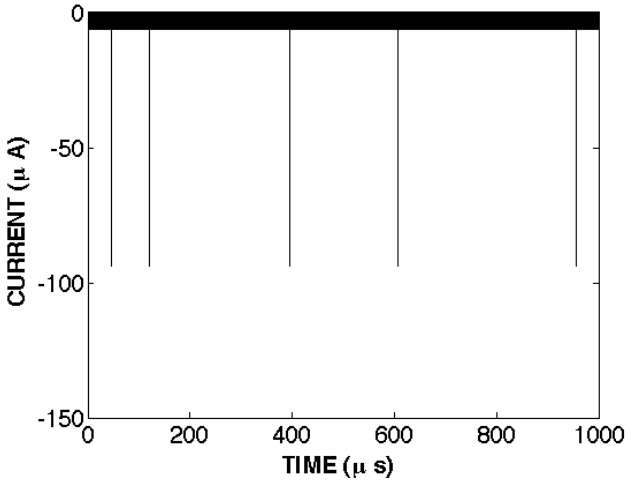


Figure 8. Dark Current v.s time at 25°C and at OV=+3.8V.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

frequency of these spikes is the *DC* of the pixel. These dark counts are attributed to generation inside the depleted region of the junction and / or diffusion from quasi neutral boundaries of a single free carrier which triggers the avalanche. The integrated current signal of a dark count in a short time window, typically 50-100 ns, divided by the electron charge q, is usually referred as the gain of the pixel (G). It was demonstrated in a previous work [28] that the steady-state dark current at any temperature and voltage condition of Fig. 7 is the product of q, G and DC, in symbols:

$$I(V,T) = q \times G(V,T) \times DC(V,T) \quad (3)$$

It is clear that the difference between the dark currents of the two devices at 25°C and 65°C (Fig 7) is necessarily due to a difference or in the G or in the DC.

Fig. 9 shows the measured G of device 1 (blue symbol) and of device 2 (red symbol) at voltages higher than the BV and at three temperatures: -25°C (circles), 25°C (triangles) and 65°C (squares). G was measured as described in [28] integrating the mean dark pulse. As the data show, the gain is nearly the same for both devices at all the investigated temperatures. This is expected because $G = 1/q \times C \times OV$, C is the junction capacitance, and the values of C and OV are the same for both devices.

The DC of the two devices is shown in Fig. 10 for the same voltage and temperature ranges investigated for the G (Fig 9). It was measured counting the dark pulses in a time window of 1s. Blue symbols are used for the DC of device 1 and red symbols for the DC of device 2. At -25°C (circles) both devices have the same DC (~10 hz at V=-31V). At 25°C (triangles) the DC of device 1 is ~ 10 times the DC of device 2 and at 65°C the difference becomes ~ 2 order of magnitudes, roughly the same difference observed in the dark current of Fig. 8.

To better understand the DC behavior with respect to the temperature, the DC of both devices has been measured at fixed OV in a temperature range of 100°C, from -25°C to 85°C.

The Arrhenius plot of the DC at a fixed OV=+3V, i.e. the Napieran logarithm of the DC vs. $1/kT$, where k is the Boltzmann constant and T the temperature in Kelvin, is shown in Fig. 11. The experimental data for device 1 (blue symbols) are compared with those of device 2 (red symbols). Lines are the simulation results, as discussed in the following. The slope of the ln(DC) as a function of $1/kT$ provides the DC activation energy E_a [29]. Two different slopes are recognized from the plot: for temperature lower than ~ 10°C for device 1 and ~ 40°C for device 2 the activation energy is $E_{a1} \sim E_G/2$, where E_G is the silicon forbidden energy bandgap (1.12eV). At higher temperature (>10°C for device 1 and >40°C for device 2) the slope of the Arrhenius plot provides an activation energy $E_{a2} \sim E_G$. The experimental values of E_{a1} and E_{a2} are summarized in Table I. Similar values are found regardless of OV. E_{a1} value indicates that at low temperature the DC of both devices is due to SHR generation-recombination defects located inside the depleted region of the p-n junction. The physical explanation of $E_{a2}=E_G$ is that the diffusion of minority carriers from the boundary of the depleted region is the prevalent effect causing the DC at high temperature. Now it is clear that the larger reduction of the DC of device 2 with respect to the DC of device 1 at temperatures higher than 10°C (Fig. 10) is due to a reduction of the diffusion current in device 2. However, it is still unclear why it happens. At a first glance, one may expect that a reduction of the diffusion current at the perimeter of the active area could cause a reduction in the DC, as already observed in the forward regime [1]. This hypothesis could be suggested also by the different device architecture at the borders, region 2 in Fig. 2, for the two devices. Device 1 shows a large p-type dopant

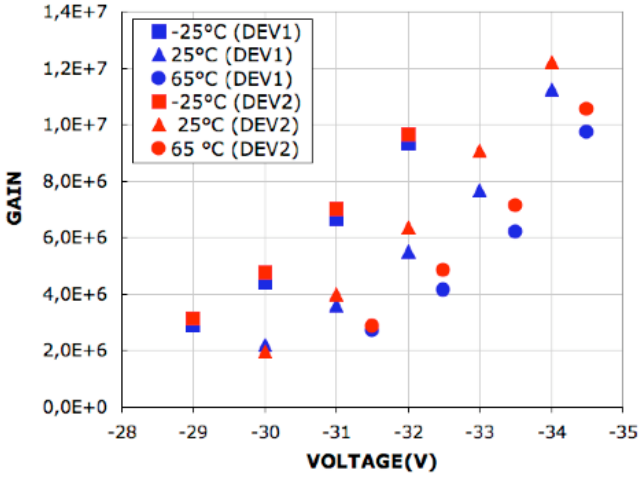


Figure 9. Gain v.s. voltage at -25°C (circle), 25°C (triangle) and 65°C (square) of device 1 (blue symbols) and of device 2 (red symbols).

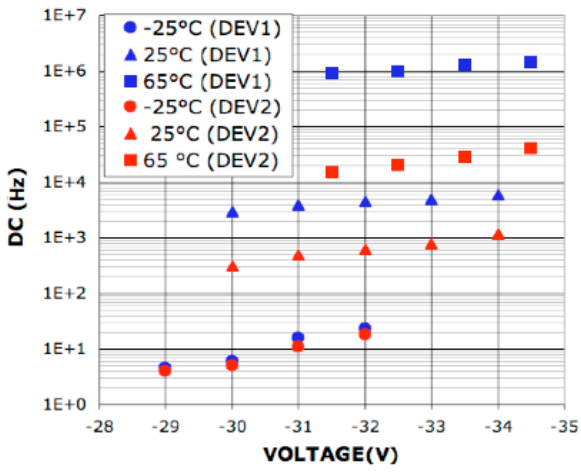


Figure 10. Dark count rate v.s. voltage at -25°C (circle), 25°C (triangle) and 65°C (square) of device 1 (blue) and of device 2 (red).

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

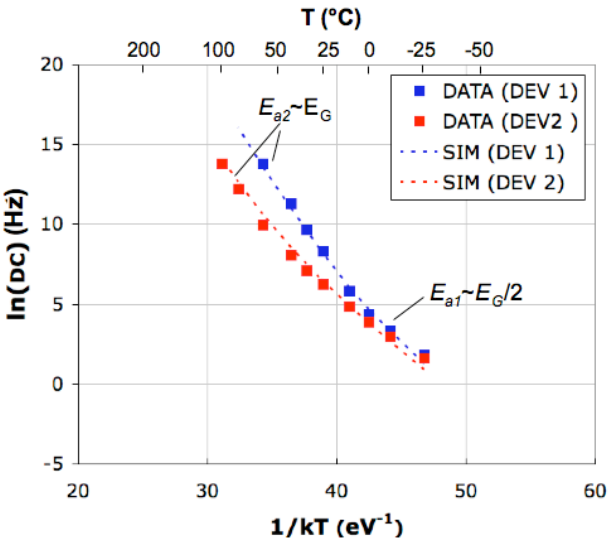


Figure 11. Comparison of the Arrhenius plot of DC measured (symbol) and simulated (dotted lines) at constant OV=+3 V (10%) for device 1 (blue) and device 2 (red).

concentration (2×10^{18}/cm^3), while device 2 has the epitaxial Si concentration value (~1×10^{15}/cm^3). Since the Auger effect [30] is a relevant recombination mechanism at high dopant concentration, a different effective lifetime in the periphery of the two devices could explain the lower diffusion current. A more careful inspection of the results, supported by our electrical simulations, allowed us to obtain a different conclusion. The electrical behavior of both devices was simulated at different temperatures and reverse bias conditions varying the devices physical parameters in the three defined region (see Fig. 2). τ_0 was varied in the range 10^{-7} – 10^{-3} s and E_T in the range 0 – 0.3 eV from to $E_G/2$.

The simulation shown in Fig. 12 refers to 65 °C since at this temperature only the diffusion regime is present. It shows the current density distribution of device 1 at OV=+3V. Even if the 2D drift-diffusion simulator can not reproduce the exact value of J_{TOT} above BV, i.e., it can not simulate the device operation in the Geiger Mode, it gives important information. At voltages above the BV, the current flows preferentially at the center of the device. This behavior was found for all the explored parameter set and for both devices, since it is due only to the electrical field distribution.

Fig. 13 shows the 2D simulation of the electrical field of device 1 at 65°C and +3V of OV (same conditions of Fig. 12). The electrical field at the lateral border is well below the junction breakdown value. It is negligible with respect to the maximum value in the active region, in which, the field is above the value needed for avalanche breakdown. Even if the diffusion current in forward bias or in reverse bias at voltages below BV has it maximum value at the border of the device junction, above BV the probability to trigger an avalanche in this region is too low. The results above described allowed us to conclude that the large reduction of the diffusion current between the two devices, is due to differences in the region 1 physical characteristics. The simulations do not allow us to obtain information on the dark current value above breakdown, but can discriminate between the different components of the leakage current below breakdown. It should be reminded that experimentally, the leakage current below breakdown is due to three components: minority carrier diffusion and SHR generation in region 1, and perimeter current [31]. Since the first two components contribute also to the dark current, the main difference between the two reverse bias regimes (below and above breakdown) is due to the presence of perimeter current below breakdown. In fact, a carrier coming from the perimeter has a probability close to zero to trigger an avalanche [32]. Moreover, the sum of the first two current components is well below the experimentally measured value, demonstrating that the leakage current below breakdown is entirely dominated by the perimeter component. Similar considerations and results hold for device 2.

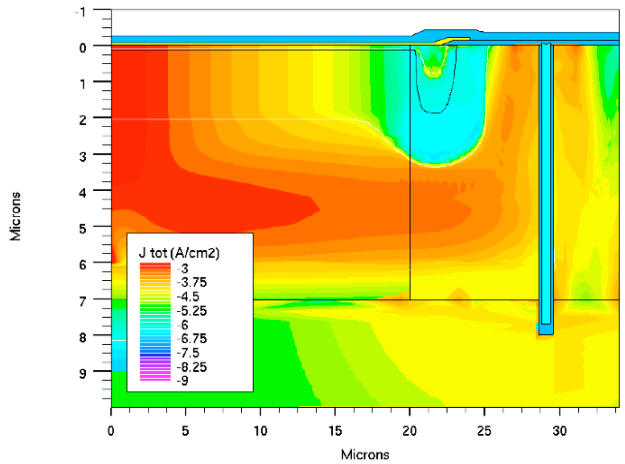


Figure 12. Total current density distribution (Jtot) at 65°C and at OV=+3V of device 1.

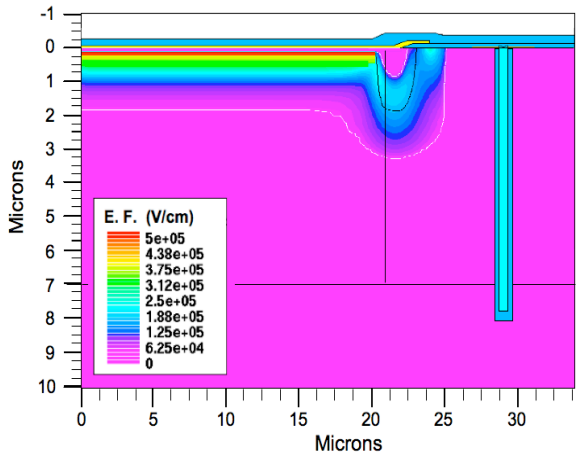


Figure 13. Electrical field distribution at 65°C and at OV=+3V of device 1.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Fig. 14 shows the simulated total current density at 65°C and at -20V of device 1.

The simulated *DC* of Fig. 11 was then obtained considering the J_{TOT} value taken in the center of the depletion layer in region 1 for V=-20V, as shown by the cut line 1 in Fig. 14. Moreover, a uniform triggering probability, P_t, of 0.35 at *OV*=+3V was assumed, as calculated in [33] for a similar device. In symbols:

$$DC=A_{ACT}\times P_t\times J_{TOT}/q. \quad (4)$$

The best fit parameters obtained predict the same τ_0 and E_T for both devices. In region 1, τ_0=1ms and the SHR trap energy is E_T=60 meV below midgap, i.e., 0.54 eV, while in region 2 τ_0=10 µs and E_T=200 meV.

The lower simulated J_{TOT} in the active area of device 1 with respect to that of device 2 (not shown) is due to a large difference in the diffusion component of the *DC* for the two devices at temperatures higher than 10°C. The simulation results are shown in Fig. 11 with the dashed line for device 1 and solid line for device 2.

The differences in the diffusion current passing trough the p-n junction between the two devices in region 1 at 65° must be due to different contribution of the diffusion current components. J_{TOT} is the sum of the diffusion electron current (J_{e-}) and the diffusion hole current (J_{h+}) The first is due to minority electron carriers diffusing from the p bulk to the n^{++} cathode, the latter is due to the diffusion of minority hole carriers from the cathode to the p bulk. The J_{TOT} (squares), J_{e-} (continuous line) and J_{h+} (dashed line) of device 1 (blue) and of device 2 (red) along the cut line 1 of Fig. 14 in the first 2 µm of depth at 65°C and for a voltage polarization of -20V are summarized in Fig. 15. In device 1, J_{TOT}~J_{e-} while in device 2 J_{TOT}~J_{h+} as shown in Fig. 15.

The reduction of the J_{e-} current in device 2 is the cause of the strong reduction of the diffusion current. In fact, J_{h+} is the same in both devices being only due to the doping profile of the n^{++} cathode. As discussed in [34], the J_{e-} decrease in device 2 is mainly due to the doping profile in the epitaxial region, shadowed area in Fig. 3. In fact, minority carriers in this layer (electrons) have a different gradient in their concentration in the substrate direction in the two devices. The gradient is higher in device 2 with respect to device 1, leading to a diffusion of electrons toward the substrate higher than in device 2. As a result, the net diffusion current of electron toward the cathode is reduces below the hole diffusion current flowing in the opposite direction.

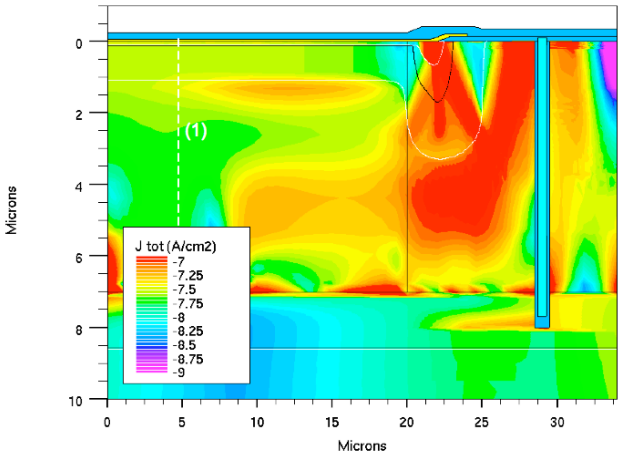


Figure 14. Total current density distribution (Jtot) at 65°C and at V=-20V of device 1. The current inside the depletied region along the cut line 1 is the value cosidered for the DC simulation .

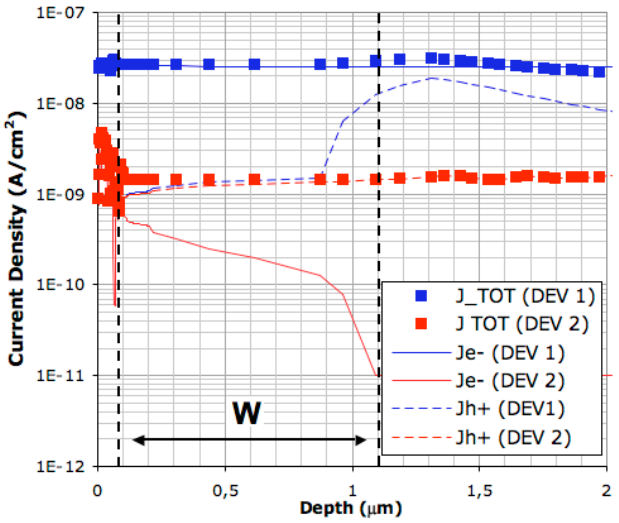


Figure 15. JTOT (squares), Je- (continuous line) and Jh+ (dashed line) at 65°C and at -20V along the cut line 1 of fig. 14 of device 1 (blue) and device 2 (red). W is the length of the depleted region of the p-n junction.

Finally, Fig. 16 shows the comparison of the experimental *DC* (symbols) and the *DC* simulated without SHR generation and recombination (dotted line) at *OV*=+3 V with respect to 1/*kT* for device 2. The comparison shows that at room temperature the experimental *DC* is close to its minimum physical level due to *DC* diffusion component. It is to note that the diffusion process of minority carriers is an intrinsic property of p-n junctions and cannot be avoided. In device 2, diffusion current has been reduced to its minimum value, being dominated by the cathode design. In order to achieve a further reduction of the hole diffusion, a different architecture must be designed. An improvement of the *DC* at room temperature can be reached reducing the defect concentration in the depleted region. We estimated the presence of about 1.6±1.3 defects/cm^{-3} in both devices hence a further reduction is a difficult goal to achieve.

V. SiPM PROPERTIES

In this section, electrical and optical performance of SiPM full array are presented and discussed.

A. Dark count and Cross Talk

The *DC* in a SiPM is conventionally defined as the frequency of dark pulses exciding half of the amplitude of the signal produced by one photo-electron (p.e.) [35].

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

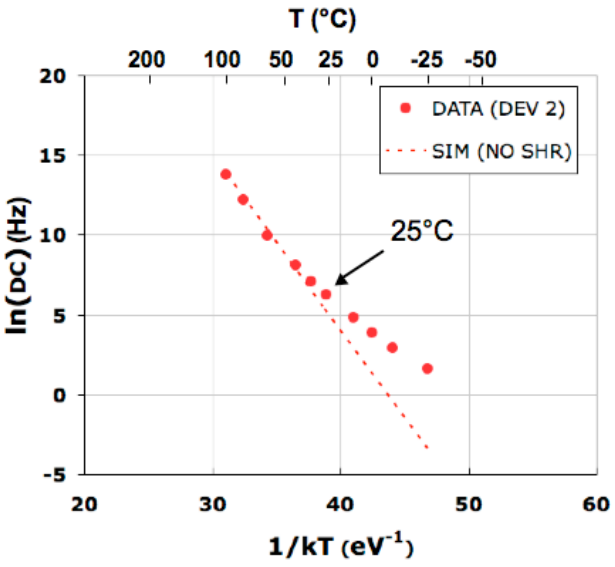


Figure 16. Comparison of the Arrhenius plot of DC measured (symbols) and simulated (dotted lines) without SHR G-R at constant OV=+3 V of device 2.

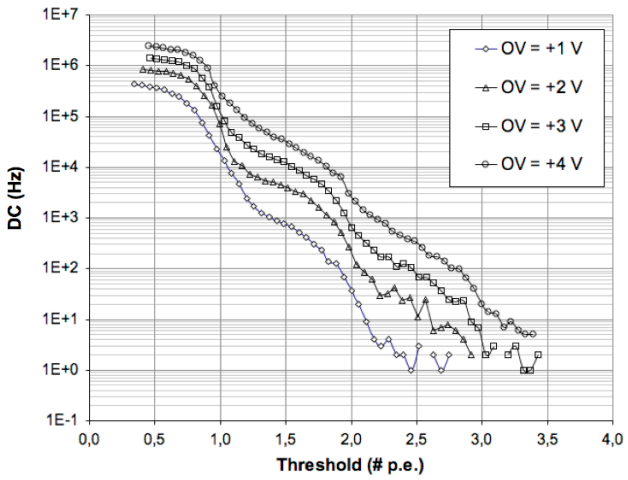


Figure 17. Measured DC at different OV of a 20×20 pixels SiPM as a function of the photoelectron signal amplitude threshold.

Fig. 17 shows the DC measured at room temperature for a 20×20 pixels SiPM (the pixel A_{ACT} is 40×40 µm^2) at different *OV*s, from +1V (diamonds) to +4V (circles) as a function of the normalized photo-electron threshold. The DC at 0.5 p.e. threshold level varies from 400 kHz to 3 MHz in the measured *OV* range. This value is roughly the *DC* rate due to only one pixel in breakdown, being the contribution of two or more pixel in breakdown at the same time to the *DC* rate at least one order of magnitude lower. In fact, at 1.5 p.e. threshold level (two pixels in breakdown simultaneously) the *DC* decreases of about three order of magnitudes at the lowest *OV* (~1 kHz at OV=+1V). The decrease is even more pronounced at 2.5 p.e (three pixels in breakdown simultaneously), being ~2 Hz at +1 *OV*.

The strong reduction in the *DC* value from 0.5 p.e. to 1.5 p.e. is due to different factors. First, the probability of simultaneous avalanche in two different pixels is lower than the probability of a single count. Moreover, the second pixel avalanche may be related to the first pixel one. In fact, there is a finite Cross Talk Probability (*CP*) for each device, strongly related to the array layout. The *CP* can be roughly quantified as:

$$CP = \frac{DC_{1.5}}{DC_{0.5}} \quad (5)$$

where $DC_{0.5}$ and $DC_{1.5}$ are the dark count rate at 0.5 and at 1.5 of the photoelectron signal level threshold. It should be stressed that using this approximation two pixels going in breakdown simultaneously are considered correlated.

The effect of the trench presence in the SiPM array is clearly visible by the inspection of Fig. 18 *a*. In fact, in figure the dark counts of two SiPM both having 20×20 pixels biased to the same *OV* (+2V) are compared. The red triangles are the data obtained from a SiPM with trenches, while the blue squares are from a SiPM without trenches around the pixels. Despite of the fact that the two devices have the same *DC* for single pixel breakdown, they strongly differ in the *DC* for two pixels in avalanche at the same time, being *CP* 0.7% for the device with trenches and 7% for the array without trenches. The trenches presence reduces the two pixels *DC* of one order of magnitude.

The *CP* probability was measured as a function of the *OV* for the two devices described before and the results are summarized in Fig. 18 *b*. The difference is one order of magnitude in the full range of operation. It could be inferred that the *CP* for the array with trenches we measured is actually the probability that two uncorrelated single events occur at the same time. All the devices discussed from now on are arrays with trenches.

B. Dark current in large device

The data so far shown in Section IV refers only to the best single pixels investigated. Since SiPM are an array of pixels, their performances are not exactly the best pixel performances multiplied by the number of pixels in the array. The dark currents can be worsened by the presence of randomly distributed defects that cause a distribution of performances around a mean value. The relationship among the dark currents in single pixels and in complete SiPM arrays can be summarized by the data (points) and simulations (lines) compared in Fig. 19 and already reported in [36]. We modeled the dark current of a single pixel as:

$$I_D = q\left(\frac{N_{Def}}{\tau} + \frac{A_{Pixel}}{\tau_i}\right)G \quad (6)$$

where q is the elementary charge, N_{Def} is the number of carrier generating defects per pixel in the active volume, τ is the average time for carrier generation event by one defect,

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

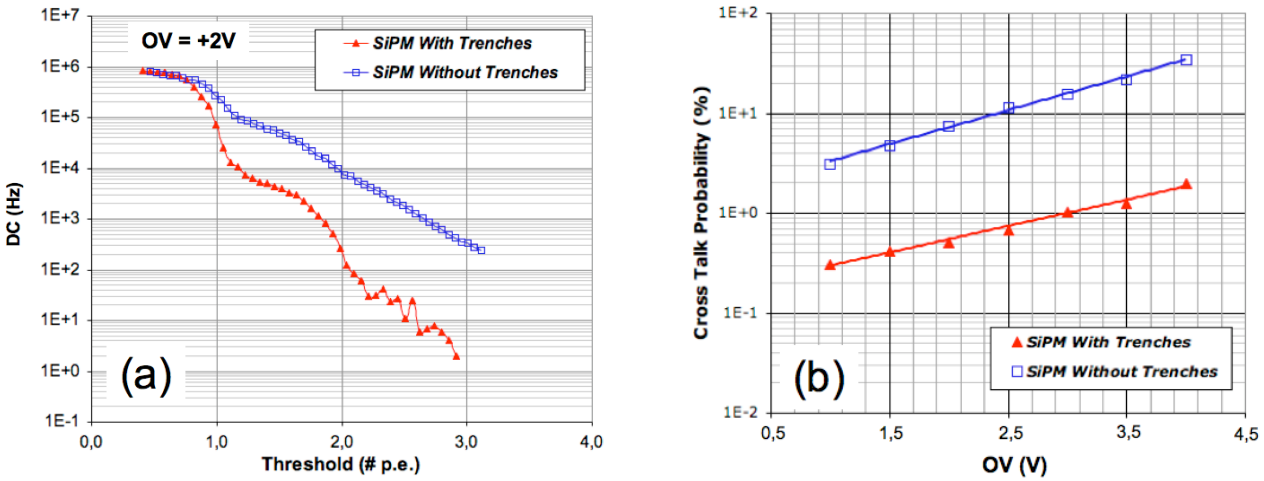


Figure 18. (a) Comparison of the dark count rate as a function of the photoelectron signal amplitude treshold and (b) of the cross talk probability vs. overvoltage of a 20×20 pixels SiPM with trenches (red closed symbols) and a 20×20 pixels SiPM without trenches (blue open symbols).

A_{Pixel} is the single pixel active area, τ_i is the average time per unit area for the intrinsic carrier generation due to diffusion from the quasi neutral regions to the active volume, and G is the gain. The I_D of the overall SiPM devices is simply the sum of the currents of single pixels as above modeled, with no contribution of extrinsic defects providing high leakage paths. In particular, Fig. 19 shows frequency histograms comparing the dark currents measured at room temperature of single pixels and SiPM arrays for a total of 952 devices at OV of 2, 3, and 4 V. The SiPM device contains 4096 pixels, so the respective currents of SiPM to single pixel should stay in ratio of about 4,000, as actually found.

To model I_D in the present devices, we should observe that the term N_{Def}/τ dominates at room temperature [36], hence the I_D statistics should essentially coincide with the N_{Def} statistics. The N_{Def} statistics is a Poisson statistics, hence the probability dP to have a DC between I_D and I_D+dI_D is:

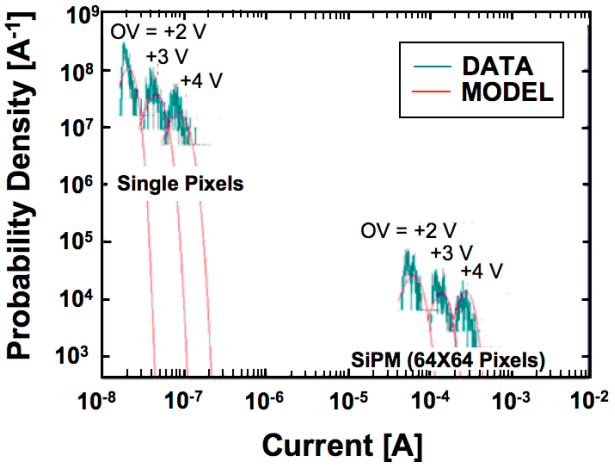


Figure 19. Probability density as a function of the output current at OV of 2V, 3V and 4V, for both single pixels and arrays (having 4096 pixels). The solid red lines are the model results.

$$\frac{dP}{dI_D} = N \exp\left[\frac{m_{I_D}}{\sigma^2_{I_D}}\left(I_D \log(m_{I_D}/I_D) + (I_D - m_{I_D})\right)\right] \quad (7)$$

where N is a normalization constant, m^2_{ID} is the statistical average of the dark current and σ^2_{ID} is the variance. In the case of the SiPM arrays the same expression holds. Fig. 19 reports also the model curves, which show a good match with the experimental data. The model predicts that the combination of statistical parameters σ^2_{ID}/m^2_{ID} should be equal to $q/\tau G$ or 4096× $q/\tau G$, for the single pixel and the SiPM array, respectively.

C. SiPM operation under illumination

Once defined the array performances in dark, measurements under a low illumination were carried out. The device was biased at +2V OV and the pulsed laser (6ns pulses) light was defocused in order to reduce the photon density. The device response is summarized in Fig. 20. In particular, Fig. 20*a* shows the persistence signal acquired on the OSC. The signal is due to the current spikes provided by one to 6 pixels (clearly identified) fired at the same time. More pixels have been fired by with a lower probability during the acquisition time (about 15 min). The signal width is limited by the oscilloscope resolution. This measurement can be made quantitative as shown in Fig. 20 *b*, where counts vs. the signal integrated charge (in 20 ns time range) is reported. The simultaneous firing of up to 8 pixels has been detected. The Gaussian distribution of each peak in Fig. 20 *b* is a clear evidence of the good pixel uniformity in terms of performances. Moreover, the distance between the peaks provide the information on the array gain [37]. It has been estimated as 10^6 at +2V OV. It should be stressed that the light signal, down to one photon count, has been acquired using only a digital oscilloscope, hence this device can be used without an external amplification circuit.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

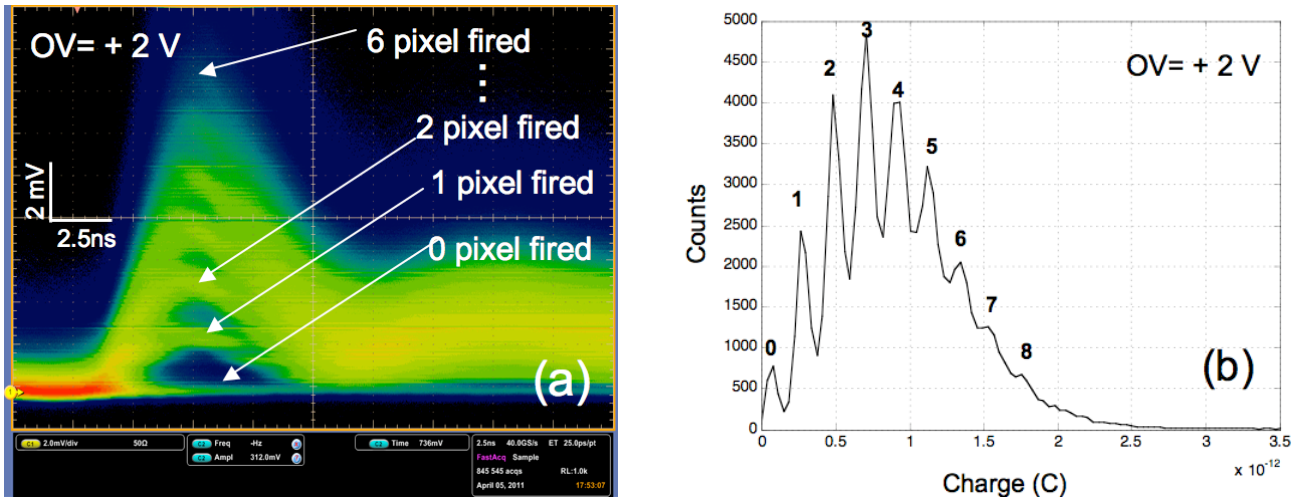


Figure 20. (a) Image of a persistence on a digital OSC and (b) charge distribution of the pulses a 20×20 pixels SiPM for a low intensity nano second laser light at OV=+2V.

VI. CONCLUSIONS

We reviewed our main results on the study of both single pixels and SiPM arrays produced by STMicroelectronics in Catania. Our data coupled to an extensive simulation study, show that the single pixel technology is close to its ultimate physical limit. The *DC* is dominated by diffusion of minority carriers from the cathode for temperatures down to 40°C. At lower temperatures, SHR generation is the main *DC* source. The only improvement in the single pixel technology could be provided by a further reduction in the defect concentration that, up to now, has been estimated to be ~ 1.6±1.3 defects/cm^{-3} in the best device. Obviously, the single pixel performances have a spread, due to the very low defect concentration needed to obtain the "best" device.

Not all the pixels forming the SiPM array are the "best" pixel, their dark current is distributed around a mean value. We found that it follows a Poissonian distribution perfectly mirroring the defect random distribution on the wafer. Hence, SiPM arrays exhibit performances worsened by the presence of defects placed according to a Poissonian distribution.

The presence of optical trenches surrounding each pixel strongly improves the SiPM performances, reducing the cross talk probability of one order of magnitude with respect to arrays without trenches.

The *DC* of SiPM arrays having the latest pixel design technology described at room temperature is in the order of 1 MHz/mm^2 at OV=+3V (~10%), close to that reported by other scientists. The *CP*, thanks to the fabrication of the trenches, is lower than 2% till OV=+4V (~15%), the lowest value, to our knowledge, reported in literature.

Finally, we have shown that these devices can be used as single photon counters also without a complex amplification stage.

ACKNOWLEDGMENT

This work has been partially funded by STMicroelectronics and by the national project MIUR-PON "Hyppocrates – Sviluppo di Micro e Nano-Tecnologie e Sistemi Avanzati per la Salute dell'uomo" (PON02 00355).

REFERENCES

[1] R. Pagano et al., "Improvement of the diffusive component of dark current in SiPM pixels", The Third International Conference on Sensor Device Technologies and Applications, SENSORDEVICES 2012, ISBN: 978-1-61208-208-0.

[2] M. D. Eisaman, J. Fan, A. Migdall, and S. V. Polyakov, "Invited Review Article: Single-photon sources and detectors", Rev. Sci. Instrum., vol. 82, no. 071101, 2011, pp. 1-25.

[3] D. Renker and E. Lorenz, "Advances in solid state state photon detectors", Journ. Instrum., vol. 4, no. P04004, 2009, pp. 1–56.

[4] T. Iijima, "Status and perspectives of vacuum-based photon detectors", Nucl. Instrum. Methods Phys. Res. A, vol. 639, no. 1, 2011, pp. 137–143.

[5] J. C. Campbell et al., "Recent Advances in Avalanche Photodiodes", IEEE Journ. Selec. Topic Quant. Electr., vol. 10, no. 4, 2004, pp. 777-788.

[6] G. Bondarenko, B. Dolgoshein, V. Golovin, Ilyin, R. Klanner, and E. Popova, "Limited Geiger-mode silicon photodiode with very high gain", Nucl. Phys. B Proc. Suppl., vol. 61 B, 1998, pp. 347-352.

[7] P. Buzhan et al., "Silicon photomultiplier and its possible applications," Nucl. Instrum. Meth. Phys. Res. A, vol. 504, no. 1–3, 2003, pp. 48–52.

[8] N. Otte et al., "The Potential of SiPM as Phototn Detector in Astroparticle Physics Experiments like MAGIC and EUSO", Nucl. Phys. B Proc. Suppl., vol. 150, 2006, pp. 144-149.

[9] V. D. Kovaltchouk, G.J. Lolos, Z. Papandreou, and K. Wolbaum, "Comparison of a silicon photomultiplier to a traditional vacuum photomultiplier" Nucl. Instrum. Meth. Phys. Res. A, vol. 538, no. 1–3, 2005, pp. 408–415.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[10] B. Dolgoshein, et al. "Status report on silicon photomultiplier development and its applications", Nucl. Instrum. Meth. Phys. Res. A, vol. 563, 2006, pp. 368-376.

[11] D. Renker, "Geiger-mode avalanche photodiodes, history, properties and problems", Nucl. Instrum. Meth. Phys. Res. A, vol. 567, no.1, 2006, pp. 48-56.

[12] S. Cova, A. Lacaita, and G. Ripamonti, "Trapping Phenomena in Avalanche Photodiodes on Nanosecond Scale" IEEE Electr. Dev. Lett., vol. 12, 1991, pp. 685-687.

[13] M. Ghioni, A. Gulinatti, I. Rech, F. Zappa, and S. Cova, "Progress in Silicon Single-Photon Avalanche Diodes", IEEE Jour. Sel. Top. Quant. Electr.,vol. 13, no. 4, 2007, pp. 852-862.

[14] B. F. Aull et al., "Geiger-Mode Avalanche Photodiode for Three Dimensional Imaging", Lincoln Lab. Jour., vol. 12, no. 2, 2002, pp. 335-350.

[15] E. Sciacca et al., "Silicon Planar Technology for Single-Photon Optical Detectors", IEEE Trans. Electr. Dev., vol. 50, 2003, pp. 918-925.

[16] F. Zappa, S. Tisa, A. Tosi, and S. Cova, "Principles and features of single-photon avalanche diode arrays" Sensors and Actuators A, vol. 140, 2007, pp. 103-112.

[17] W. G. Oldham, R. R. Samuelson, and P. Antognetti, "Triggering Phenomena in Avalanche Diode", ", IEEE Trans. Electr. Dev., vol. 19, no. 9, 1972, pp. 1056-1060.

[18] S. Cova, M. Ghioni, A. Lacaita, C. Samori, and F. Zappa, "Avalanche phototdiodes and quenching circuits for single-photon detection", Appl. Opt., vol. 35, no. 12, 1996, pp. 1956-1976.

[19] A. Lacaita, F. Zappa, S. Bigliardi, and M. Manfredi, "On the bremsstrahlung origin of hot-carrier-induced photons in silicon devices", IEEE Trans. Electr. Dev., vol. 40, no. 3, 1993, pp. 577-582.

[20] J. Briare and K. S. Krisch., "Principles of Substrate Crosstalk generation in CMOS circuits", IEEE Trans. Comp. Aided Des. Integr. Circ. And Sys., vol. 19, no. 6, 2000, pp. 645-653.

[21] O. Mineev et al.," Scintillator counters with multi-pixel avalanche photodiode readout for the ND280 detector of the T2K experiment", Nucl. Instrum. Meth. Phys. Res. A, vol. 577, no. 3, 2007, pp. 540-551.

[22] M. Danilov," Novel photo-detectors and photo-detector systems", Nucl. Instrum. Meth. Phys. Res. A, vol. 604, no. 1-2, 2009, pp. 183-189.

[23] M. Mazzillo et al., "Silicon Photomultiplier Technology st STMicroelectronics", IEEE Trans. Nucl. Sci., vol. 56, 2009, pp. 2434-2442.

[24] E. Sciacca et al., "Arrays of Geiger Mode Avalanche Photodiodes", IEEE Phot. Tech. Lett., vol. 18, no. 15, 2006, pp. 1633-1635.

[25] [Online 13.06.2013]. Available at http://www.silvaco.com.

[26] W. Shockley, "The Theory of p-n Junction in Semiconductors and p-n Junction Transistors", Bell Sys. Tech. Jour., vol. 48, 1949, pp. 435-489.

[27] W. N. Grant, "Electron and Hole Ionization Rates in Epitaxial Silicon at High Electric Fields", Solid-State Electr., vol. 16, 1973, pp. 1189-1203.

[28] R. Pagano et al., "Dark Current in Silicon Photomultiplier Pixels: Data and Model", IEEE Trans. Electr. Dev., vol. 59, no. 9, 2012, pp. 2410-2416.

[29] W. J. Kindt and H. W. van Zeijl, "Modelling and Fabrication of Geiger mode Avalanche Photodiodes", IEEE Trans. Nucl. Sci., vol. 45, no. 3, 1998, pp. 715-719.

[30] L. Passari and E. Susi, "Recombination mechanisms and doping density in silicon", J. Appl. Phys., vol. 54, no. 7, 1983, pp 3935-3937.

[31] A. Poyai, E. Simeon, C. Claeys, and A. Czerwinski, "Silicon substrate effects on the current-voltage characteristics of advanced p-n junction", Mat. Sci. Eng. B, vol. 73, no. 1-3, 2000, pp. 191-196.

[32] J. C. Jackson, P. K. Hurley, B. Lane, and A. Mathewson, "Comparing leakage currents and dark counts rate in Geiger-mode avalanche photodiodes", App. Phys. Lett., vol. 80, no. 22, 2002, pp. 4100-4102.

[33] M. Mazzillo et al., "Quantum Detection Efficiency In Geiger Mode Avalanche Photodiode," IEEE Trans. Nucl. Sci., vol. 55, no. 6, 2008, pp. 3620–3625.

[34] R. Pagano et al., "Silicon photomultiplier device architecture with dark current improved to the ultimate physical limit", App. Phys. Lett., vol. 102, no. 183502, 2013, pp. 1-4.

[35] P. Finocchiaro et al., "Characterization of a novel 100-channel silicon photomultiplier – part I: noise", IEEE Trans. Electr. Dev., vol. 55, no. 10, 2008, pp. 2757-2764.

[36] R. Pagano et al., "Silicon Photomultipliers: Dark Current and its Statistical Spread", Sensors & Transducers Journal, vol. 14, no. 1, 2012, pp. 151-159.

[37] P. Finocchiaro et al., "Characterization of a novel 100-channel silicon photomultiplier – part II: Charge and Time", IEEE Trans. Electr. Dev., vol. 55, no. 10, 2008, pp. 2765-2773.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Application of the Simulation Attack on Entanglement Swapping Based QKD and QSS Protocols

Stefan Schauer and Martin Suda
Safety and Security Department
AIT Austrian Institute of Technology GmbH
Vienna, Austria
stefan.schauer@ait.ac.at, martin.suda.fl@ait.ac.at

***Abstract*—We discuss the security of quantum key distribution protocols based on entanglement swapping against collective attacks. Therefore, we apply a generic version of a collective attack strategy on the most general entanglement swapping scenario used for key distribution. Further, we focus on basis transformations, which are the most common operations performed by the legitimate parties to secure the communication. In this context, we show that the angles, which describe these basis transformations can be optimized compared to an application of the Hadamard operation. As a main result, we show that the adversary's information is reduced to a new minimum of about 0.45, which is about 10% lower than in other protocols. To become a better overview how and on which protocols this generic version of a collective attack is applicable, the security of different quantum key distribution and quantum secret sharing protocols is discussed. Here we show that applying two basis transformations using different angles the security of a particular protocol can be increased by about 25%.**

***Keywords*—quantum key distribution; entanglement swapping; security analysis; optimal basis transformations.**

I. INTRODUCTION

The security of quantum key distribution (QKD) protocols based on entanglement swapping has been discussed on the surface so far. In a recent article [1], a novel attack strategy and its implications on the security of entanglement swapping based protocols was discussed. This attack strategy will be referred to as *simulation attack* since the major idea is to simulate the correlation between Alice's and Bob's measurement results [2]. In this article, we want to take a closer look at the application of the simulation attack on different QKD and quantum secret sharing (QSS) protocols together with the necessary improvements on the security of these protocols.

QKD is an important application of quantum mechanics and QKD protocols have been studied at length in theory and in practical implementations [3], [4], [5], [6], [7], [8], [9], [10]. Most of these protocols focus on prepare and measure schemes, where single qubits are in transit between the communication parties Alice and Bob. The security of these protocols has been discussed in depth and security proofs have been given for example in [11], [12], [13]. In addition to these prepare and measure protocols, several protocols based on the phenomenon of entanglement swapping have been introduced [14], [15], [16], [17], [18]. In these protocols, entanglement swapping is used to obtain correlated measurement results between the legitimate communication parties. In other words, each party performs a Bell state measurement and due to entanglement swapping their results are correlated and further on used to establish a secret key.

Entanglement swapping has been introduced by Bennett et al. [19], Zukowski et al. [20] as well as Yurke and Stolen [21], respectively. It provides the unique possibility to generate entanglement from particles that never interacted in the past. In detail, Alice and Bob exchange two Bell states of the form $|\Phi^+\rangle_{12}$ and $|\Phi^+\rangle_{34}$ such that afterwards Alice is in possession of qubits 1 and 3 and Bob of qubits 2 and 4 (cf. (2) in Figure 1). The overall state can now be written as

$$|\Phi^+\rangle_{12} \otimes |\Phi^+\rangle_{34} = \frac{1}{2}\Big(|\Phi^+\rangle|\Phi^+\rangle + |\Phi^-\rangle|\Phi^-\rangle + |\Psi^+\rangle|\Psi^+\rangle + |\Psi^-\rangle|\Psi^-\rangle\Big)_{1324} \quad (1)$$

Then, Alice performs a complete Bell state measurement on the two qubits 1 and 3 in her possession, and at the same time the qubits 2 and 4 at Bob's side collapse into a Bell state although they originated at completely different sources. Moreover, the state of Bob's qubits depends on Alice's measurement result (cf. (4) in Figure 1). As presented in eq. (1), Bob always obtains the same result as Alice when performing a Bell state measurement on his qubits.

So far, it has only been shown that QKD protocols based on entanglement swapping are secure against intercept-resend attacks and basic collective attacks (cf. for example [14], [15], [17]). Therefore, we analyze a general version of a collective attack where the adversary tries to simulate the correlations between Alice and Bob [2]. A basic technique to secure these protocols is to use a basis transformation, usually a Hadamard operation, similar to the prepare and measure schemes mentioned above, to make it easier to detect an adversary. In [1], the application of general basis transformations about the angles θ_A and θ_B has been discussed and it has been shown that the information of an adversary can be reduced to a minimum of $\simeq 0.45$. Based on these results we analyze the security of three different protocols with respect to the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

simulation attack. In the course of that, we are going to identify, which values for θ_A and θ_B are optimal for these protocols such that an adversary has only a minimum amount of information on the secret key.

In the next section, we are going to shortly review the simulation attack, a generic collective attack strategy where an adversary applies a six-qubit state to eavesdrop Bob's measurement result. A detailed discussion of this attack strategy can be found in [2]. In Section III, we discuss the security of entanglement swapping based QKD protocols against the simulation attack in general. Here, we are focusing on the application of one and two basis transformations and review the optimal angles for these transformations. In the following sections, we discuss the application of the simulation attack on three different protocols: on the prepare & measure QKD protocol by Bennett, Brassard, and Mermin [5] in Section IV, on the entanglement swapping based QKD protocol by Song [17] in Section V and on the QSS protocol by Cabello [16] in Section VI. We will shortly review each of these protocols and provide a detailed security analysis with respect to an application of the simulation attack. At the end, we summarize the results and give a short outlook on the next steps into this topic.

II. The Simulation Attack Strategy

In entanglement swapping based QKD protocols like [14], [15], [16], [17], [18] Alice and Bob rest their security check on the correlations between their respective measurement results coming from the entanglement swapping (cf. eq. (1)). If these correlations are violated to a certain amount, Alice and Bob have to assume that an eavesdropper is present. In 2000, Zhang, Li, and Guo presented an attack strategy, where Eve entangles herself with both parties and manages to obtain full information about the shared key [23]. This collective attack was improved in a previous article [2] to the *simulation attack* and extended to a specific protocol [18] following this basic idea: the adversary Eve tries to find a multi-qubit state, which preserves the correlation between the two legitimate parties. Further, she introduces additional qubits to distinguish between Alice's and Bob's respective measurement results. If she is able to find such a state, Eve stays undetected during her intervention and is able to obtain a certain amount of information about the key. The simulation attack can be generalized to arbitrary entanglement swapping based QKD protocols in a straight forward way, as described in the following paragraphs.

It has been pointed out in detail in [2] that Eve uses four qubits to simulate the correlations between Alice and Bob and she further introduces additional systems, i.e., $|\varphi_i\rangle$, to distinguish between Alice's different measurement results. This leads to the state

$$|\delta\rangle = \frac{1}{2}\Big(|\Phi^+\rangle|\Phi^+\rangle|\varphi_1\rangle + |\Phi^-\rangle|\Phi^-\rangle|\varphi_2\rangle + |\Psi^+\rangle|\Psi^+\rangle|\varphi_3\rangle + |\Psi^-\rangle|\Psi^-\rangle|\varphi_4\rangle\Big)_{PRQSTU} \tag{2}$$

which is a more general version than described in [2]. This state preserves the correlation of Alice's and Bob's measurement results coming from the entanglement swapping (cf. eq. (1)). From eq. (2) it is easy to see that Alice obtains one of the four Bell states when performing a Bell state measurement on qubits P and R. This measurement leaves Bob's qubits Q and S in a Bell state fully correlated to Alice's result. Accordingly, Eve's qubits T and U are in one of the auxiliary states $|\varphi_i\rangle$ she prepared.

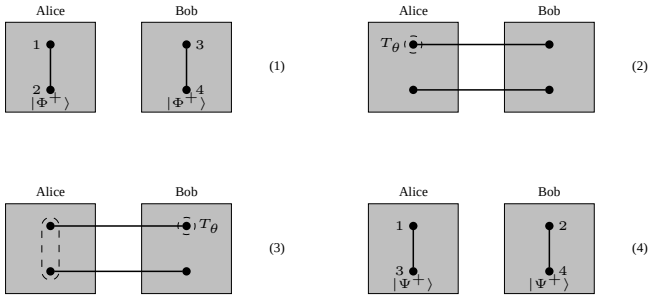


Fig. 1. Illustration of a standard setup for an entanglement swapping based QKD protocol using a basis transformation T_x.

Eve has to choose the auxiliary systems $|\varphi_i\rangle$ such that

$$\langle\varphi_i|\varphi_j\rangle = 0 \qquad i,j \in \{1,...,4\} \;\; i \neq j \tag{3}$$

which allows her to perfectly distinguish between Alice's and Bob's respective measurement results. Thus, she is able to eavesdrop Alice's and Bob's measurement results and obtains full information about the classical raw key generated out of them.

In detail, Eve distributes qubits P, Q, R and S between Alice and Bob such that Alice is in possession of qubits P and R and Bob is in possession of qubits Q and S. When Alice performs a Bell state measurement on qubits P and R the state of qubits Q and S collapses into the same Bell state, which Alice obtained from her measurement (cf. eq. (2)). In particular, if Alice obtains $|\Phi^+\rangle_{PR}$ the state of the remaining qubits is

$$|\Phi^+\rangle_{QS}|\varphi_1\rangle_{TU} \tag{4}$$

and similarly for Alice's other results $|\Phi^-\rangle$ and $|\Psi^\pm\rangle$. This is the exact correlation Alice and Bob would expect from entanglement swapping if no adversary is present (cf. eq. (1) from above). Hence, Eve stays undetected when Alice and Bob compare some of their results in public to check for eavesdroppers. The auxiliary system $|\varphi_i\rangle$ remains at Eve's side and its state is completely determined by Alice's measurement result. Therefore, Eve has full information on Alice's and Bob's measurement results and is able to perfectly eavesdrop the classical raw key.

There are different ways for Eve to distribute the state $|\delta\rangle_{P-U}$ between Alice and Bob. One possibility is that Eve is in possession of Alice's and Bob's source and generates $|\delta\rangle_{P-U}$ instead of Bell states. This is a rather strong assumption because the sources are usually located at Alice's or Bob's laboratory, which should be a secure environment. Eve's second possibility is to intercept the qubits 2 and 3 flying from Alice to Bob and vice versa and to use entanglement swapping to distribute the state $|\delta\rangle$. This is a straight forward method as already described in [2].

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

We want to stress that the state $|\delta\rangle$ is generic for all protocols where 2 qubits are exchanged between Alice and Bob during one round of key generation as, for example, the QKD protocols presented by Song [17], Li et al. [18] or Cabello [14]. As already pointed out in [2], the state $|\delta\rangle$ can also be used for different initial Bell states. Regarding protocols with a higher number of qubits the state $|\delta\rangle$ has to be extended accordingly (cf. Section VI).

III. Security Against Collective Attacks

In the following paragraphs, we discuss Eve's intervention on an entanglement swapping QKD protocol performing a simulation attack, i.e., using the state $|\delta\rangle_{P-U}$. To detect Eve's presence either Alice or Bob or both parties apply a basis transformation as depicted in Figure 1.

A. General Basis Transformations

Similar to the prepare and measure schemes mentioned in the introduction, most of the protocols based on entanglement swapping apply basis transformations to make it easier to detect the presence of an eavesdropper. The basis transformation most commonly used in this case is the Hadamard operation, i.e., a transformation from the Z- into the X-basis. In general, a basis transformation from the Z-Basis into the X-basis can be described as a combination of rotation operations, i.e.,

$$T_x(\theta,\phi) = e^{i\phi} R_z(\phi) R_x(\theta) R_z(\phi) \tag{5}$$

where R_x and R_z are the rotation operations about the X- and Z-axis, respectively. For reasons of simplicity we take $\phi = \pi/2$ in our further discussions and therefore denote the transformation is described solely by the angle θ, i.e., $T_x(\theta)$. From eq. (5) we can directly see that the Hadamard operation equals $T_x(\pi/2)$. To keep the security analysis as generic as possible we discuss a setup where a general basis transformation about an angle θ_A is applied by Alice and a transformation about an angle θ_B is applied by Bob, respectively (cf. Figure 1).

For our further discussions, we will assume that Alice and Bob prepared the initial states $|\Phi^+\rangle_{12}$ and $|\Phi^+\rangle_{34}$ as described above to make calculations easier. As already pointed out above and in more detail in [2] if Alice and Bob choose $\theta_A = \theta_B = 0$, i.e., they perform no transformation, the protocol is completely insecure. Hence, we will focus on the scenarios where either $T_x(\theta_A)$ or $T_x(\theta_B)$ or both transformations are applied. For all scenarios we assume that Alice applies $T_x(\theta_A)$ on qubit 1 and Bob applies $T_x(\theta_B)$ on qubit 4.

B. Application of a Single Transformation

For the first scenario where only Alice applies the basis transformation the overall state of the system after Eve's distribution of the state $|\delta\rangle_{P-U}$ can simply be described as

$$|\delta'\rangle = T_x^{(1)}(\theta_A)|\delta\rangle_{1QR4TU} \tag{6}$$

where the superscript "(1)" indicates that $T_x(\theta_A)$ is applied on qubit 1. When Eve sends qubits R and Q to Alice and Bob, respectively, the state after Alice's Bell state measurement on qubits 1 and R is

$$\cos\frac{\theta_A}{2}\,|\Phi^-\rangle_{Q4}|\varphi_2\rangle_{TU} + \sin\frac{\theta_A}{2}\,|\Psi^+\rangle_{Q4}|\varphi_3\rangle_{TU} \tag{7}$$

assuming Alice obtained $|\Phi^+\rangle_{1R}$ (for Alice's other three possible results the state changes accordingly). This indicates that in this case Bob's transformation back into the Z-basis does not re-establish the correlations between Alice and Bob properly. Performing the calculations we see that Bob's operation $T_x(\theta_A)$ brings qubits Q, 4, T and U into the form

$$\begin{aligned}&\cos^2\frac{\theta_A}{2}\,|\Phi^+\rangle_{Q4}|\varphi_2\rangle_{TU} + \sin^2\frac{\theta_A}{2}\,|\Phi^+\rangle_{Q4}|\varphi_3\rangle_{TU}\\ &-\frac{\sin\theta_A}{2}\,|\Psi^-\rangle_{Q4}|\varphi_2\rangle_{TU} + \frac{\sin\theta_A}{2}\,|\Psi^-\rangle_{Q4}|\varphi_3\rangle_{TU}\end{aligned} \tag{8}$$

When Bob performs a Bell state measurement we can directly see from this expression that Bob obtains either the correlated result $|\Phi^+\rangle_{Q4}$ with probability

$$\left(\cos^2\frac{\theta_A}{2}\right)^2 + \left(\sin^2\frac{\theta_A}{2}\right)^2 = \frac{3+\cos(2\theta_A)}{4} \tag{9}$$

or an error, i.e., the state $|\Psi^-\rangle_{Q4}$, otherwise. In detail, Eve introduces an error with probability $(\sin^2\theta_A)/2$, which yields an expected error probability

$$\langle P_e\rangle = \frac{1}{4}\sin^2\theta_A \tag{10}$$

Nevertheless, as long as the results are correlated, Eve obtains from her Bell state measurement on qubits T and U the state $|\varphi_2\rangle_{TU}$ with probability $(1+\cos(\theta_A))^2/(3+\cos(2\theta_A))$ and knows that Bob obtained $|\Phi^+\rangle_{Q4}$. Consequently, we obtain the expected collision probability

$$\langle P_c\rangle = \frac{1}{8}\Big(7+\cos 2\theta_A\Big). \tag{11}$$

This directly leads to the Shannon entropy

$$H = \frac{1}{2}\,h\Big(\cos^2\frac{\theta_A}{2}\Big) \tag{12}$$

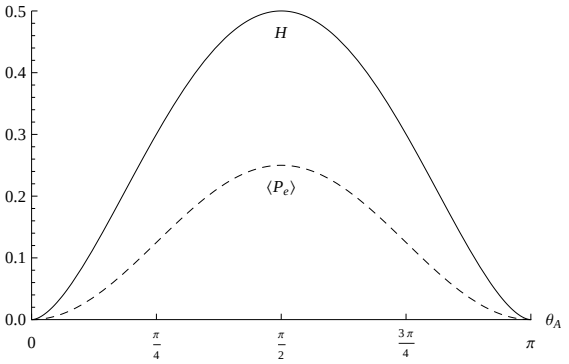


Fig. 2. Alice's and Bob's Shannon entropy H and the according average error probability $\langle P_e\rangle$ if either Alice or Bob applies a basis transformation.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

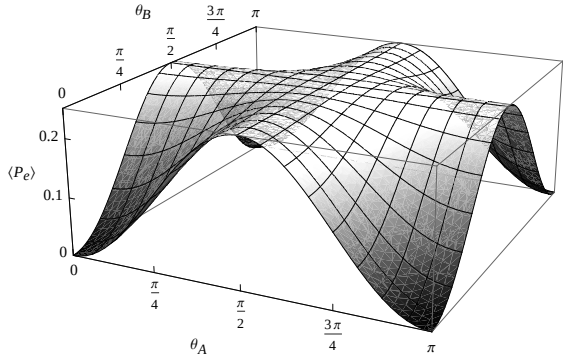


Fig. 3. Eve's expected error probability $\langle P_e \rangle$ if both parties apply a basis transformation with the respective angles θ_A and θ_B.

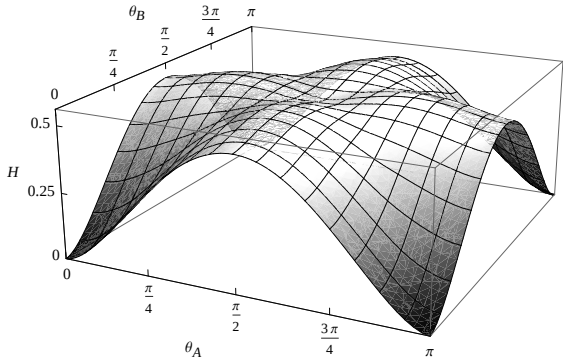


Fig. 4. Alice's and Bob's Shannon entropy H if both parties apply a basis transformation with the respective angles θ_A and θ_B.

where $h(x) = -x \log_2 x - (1-x)\log_2(1-x)$ is the binary entropy. Looking at $\langle P_e \rangle$ and H in Figure 2 we see that the optimal angle for a single basis transformation is $\pi/2$, i.e., the Hadamard operation, for protocols using only one basis transformation, as it is already known from literature [15], [2], [1]. In this case, the average error probability as well as the Shannon entropy are maximal at $\langle P_e \rangle = 0.25$ and $H = 0.5$ (cf. Figure 2). If only Bob applies the basis transformation, the calculations run analogous and therefore provide the same results. Further, Eve's information on the bits of the secret key is given by the mutual information

$$I_{AE} = 1 - H = 1 - \frac{1}{2} = \frac{1}{2} \tag{13}$$

which means that Eve has 0.5 bits of information on every bit of the secret key. Using error correction and privacy amplification Eve's information can be brought below 1 bit of the whole secret key as long as the error rate is below $\sim 11\%$ [13]. This is more or less the standard threshold value for the prepare and measure QKD protocols.

C. Application of Combined Transformations

When both Alice and Bob apply their respective basis transformation, the overall state changes to

$$|\delta'\rangle = T_x^{(1)}(\theta_A) T_x^{(4)}(\theta_B) |\delta\rangle_{1QR4TU} \tag{14}$$

and after Alice's Bell state measurement on qubits 1 and R and Bob's application of $T_x(\theta_B)$ on qubit Q the state of the remaining qubits is

$$\begin{aligned} &\cos^2 \frac{\theta_A - \theta_B}{2} \, |\Phi^+\rangle_{Q4} |\varphi_1\rangle_{TU} \\ &+ \sin^2 \frac{\theta_A - \theta_B}{2} \, |\Phi^+\rangle_{Q4} |\varphi_4\rangle_{TU} \\ &- \frac{\sin\big(\theta_A - \theta_B\big)}{2} \, |\Psi^-\rangle_{Q4} \Big(|\varphi_1\rangle_{TU} - |\varphi_4\rangle_{TU} \Big) \end{aligned} \tag{15}$$

Consequently, Bob obtains a correlated result with probability $(3 + \cos(2\theta_A - 2\theta_B))/4$ and, following the argumentation from scenario described in Section III-B above, this yields an average error probability (cf. Figure 3 for a plot of this function)

$$\begin{aligned} \langle P_e \rangle = &\frac{1}{8} \sin^2 \theta_A + \frac{1}{8} \sin^2 \theta_B \\ &+ \frac{1}{16} \sin^2\big(\theta_A + \theta_B\big) + \frac{1}{16} \sin^2\big(\theta_A - \theta_B\big) \end{aligned} \tag{16}$$

When the results are correlated Eve obtains either $|\varphi_1\rangle_{TU}$ or $|\varphi_4\rangle_{TU}$, as it is easy to see from eq. (15). Hence, Eve's information on the Alice's and Bob's result is lower compared to the first scenario, i.e., Alice's and Bob's Shannon entropy is higher:

$$\begin{aligned} H = &\frac{1}{4} \, h\Big(\cos^2 \frac{\theta_A}{2}\Big) + \frac{1}{4} \, h\Big(\cos^2 \frac{\theta_B}{2}\Big) \\ &+ \frac{1}{8} \, h\Big(\cos^2 \frac{\theta_A + \theta_B}{2}\Big) + \frac{1}{8} \, h\Big(\cos^2 \frac{\theta_A - \theta_B}{2}\Big) \end{aligned} \tag{17}$$

This is due to the fact that it is more difficult for Eve to react on two separate basis transformations with different angles θ_A and θ_B. Taking the optimal choice for only one basis transformation, i.e., the Hadamard operation, we see that if both parties apply the Hadamard operation at the same time the operations cancel out each other. Hence, the angles θ_A and θ_B have to be different. As we can further see from Figure 4, the Shannon entropy for a combined application of basis transformations is much higher than 0.5 for some regions. In detail, the maximum of the function plotted in Figure 4 is

$$H \sim 0.55 \quad \text{and thus} \quad I_{AE} \sim 0.45 \tag{18}$$

for $\theta_A = \pi/4$ and $\theta_B = \pi/2$ or vice versa. Hence, if just one of the parties applies a Hadamard operation and the other one a transformation about an angle of $\pi/4$, Eve's mutual information is about 10% lower compared to the application of a single basis transformation (cf. eq. (13)). At the same time we see from Figure 3 that for these two values of θ_A and θ_B the error probability is still maximal with $\langle P_e \rangle = 0.25$. This means Alice and Bob are able to further increase the security by the combined application of two basis transformations, one about $\theta = \pi/2$ and the other about $\theta = \pi/4$.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

IV. Application on the BBM Protocol

In 1992, Bennett, Brassard, and Mermin presented a variant of the Ekert protocol [4], where they show that a test of the CHSH-inequalities [22] is not necessary for the security of the protocol [5]. Instead of the CHSH-inequalities, Alice and Bob use two complementary measurement bases as in the BB84 protocol [3] and randomly apply them on the received qubits. Due to the entangled state Alice and Bob obtain perfectly correlated results from their measurement if no adversary is present.

A. Protocol Description

In detail, Alice and Bob use a source emitting maximally entangled qubit pairs, e.g., in the Bell-state $|\Psi^-\rangle_{12}$. This source is located between Alice and Bob and one qubit of the state is flying to Alice and the other one to Bob. When looking at physical implementations of the BBM protocol the source is usually located at the laboratory of one of the communication parties. Hence, we will assume that the source is located at Alice's lab and she sends the second qubit of each pair to Bob (cf. Figure 5). After receiving the qubit, both communication parties randomly and independently choose either the Z- or the X-basis to measure their qubit. Due to the entanglement of the qubits in the state $|\Psi^-\rangle_{12}$ Alice's measurement completely determines the state of Bob's qubit, i.e., if Alice measures a $|1\rangle$, Bob's qubit is in the state $|0\rangle$, and vice versa. If he measures in a different basis than Alice, Bob destroys the information carried by the qubit and thus will not obtain a correlated result. To identify where they used different bases both parties publicly compare all of their measurement bases and discard the results where they had chosen differently. The remaining results should be perfectly correlated and the communication parties compare a randomly selected fraction in public. If there is too much discrepancy between their results they have to assume that an adversary is present and they start over the protocol. It has also been shown by Bennett et al. in this paper that the security of this version of the protocol is equal to the security of the BB84 scheme [5].

The random measurement in either the Z- and X-basis can also be interpreted as a random application of the Hadamard operation by Alice. As pointed out above, the Hadamard operation is a complete basis transformation from the Z- into the X-basis, i.e., by an angle $\theta_A = \pi/2$. Therefore, it can be said that both Alice and Bob randomly apply the Hadamard operation on the qubits they receive and measure it in the Z-basis afterwards. In the end, both parties compare in public where they used the Hadamard operation and similar to the original protocol they discard the results where only one of them applied the Hadamard operation.

B. Security Analysis

Looking at this interpretation we want to discuss whether the Hadamard operation is optimal in this scenario. Therefore, we will discuss the information an eavesdropper Eve is able to obtain when performing a simulation attack. Further, we assume that Alice and Bob are not limited to the Hadamard operation but they use a general basis transformation $T_x(\theta_A)$.

Fig. 5. Illustration of the BBM protocol [5]. Here, Alice performs a measurement in the Z-basis.

To fit to the setting of the BBM protocol the adversary Eve has to prepare a slightly different $|\delta\rangle$ for the simulation attack, i.e.,

$$|\delta\rangle_{RST} = \frac{1}{\sqrt{2}}\Big(|0\rangle|1\rangle|\varphi_1\rangle + |1\rangle|0\rangle|\varphi_2\rangle\Big)_{RST} \tag{19}$$

This state perfectly simulates the correlation between Alice's and Bob's result in case they do not apply any operation. As described above, the auxiliary states $|\varphi_1\rangle$ and $|\varphi_2\rangle$ have to be orthogonal (cf. eq. (3)) such that they can be distinguished by Eve. For reasons of simplicity, we will assume that Eve intercepts the qubits coming from Alice and uses entanglement swapping on qubits 2 and R to establish the state $|\delta\rangle_{1ST}$ between Alice and Bob, where Bob is now in possession of qubit S.

Following the protocol Alice and Bob randomly perform the basis transformation $T_x(\theta_A)$ on their respective qubits 1 and S. Since they discard all results where just one of them applies $T_x(\theta_A)$ we are only interested in two scenarios: either none or both of them perform $T_x(\theta_A)$. In scenario one, it is easy to see from the structure of the state $|\delta\rangle_{1ST}$ that Eve's qubits are in the state $|\varphi_1\rangle_T$ whenever Alice obtains $|0\rangle$ and in the state $|\varphi_2\rangle_T$ whenever Alice obtains $|1\rangle$. In this case Eve is able to perfectly eavesdrop the respective raw key bits.

In the second scenario, the application of the basis transformation $T_x(\theta_A)$ on qubits 1 and S changes the overall state to

$$|\delta'\rangle = T_x(\theta_A)^{(1)}|\delta\rangle_{1ST}, \tag{20}$$

where the superscript "(1)" denotes an application on qubit 1. This results in the state

$$\begin{aligned}\frac{1}{\sqrt{2}}\Bigg(&\sin\frac{\theta_A}{2}\Big(|00\rangle|\varphi_2\rangle + |11\rangle|\varphi_1\rangle\Big)\\ &+\cos\frac{\theta_A}{2}\Big(|01\rangle|\varphi_1\rangle - |10\rangle|\varphi_2\rangle\Big)\Bigg)\end{aligned} \tag{21}$$

before Alice performs her measurement on qubit 1. Assuming Alice obtains $|0\rangle_1$ from her measurement and Bob applies $T_x(\theta_A)$ on qubit S this changes the state described in the previous equation into

$$\begin{aligned}&\frac{\sin\theta_A}{2}\,|0\rangle_S|\varphi_1\rangle_T + \frac{\sin\theta_A}{2}\,|0\rangle_S|\varphi_2\rangle_T\\ &-\cos^2\frac{\theta_A}{2}\,|1\rangle_S|\varphi_1\rangle_T + \sin^2\frac{\theta_A}{2}\,|1\rangle_S|\varphi_2\rangle_T\end{aligned} \tag{22}$$

From this expression we can directly see that Bob obtains from his Bell state measurement either the correlated result

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$|1\rangle_S$ with probability

$$\left(\cos^2\frac{\theta_A}{2}\right)^2 + \left(\sin^2\frac{\theta_A}{2}\right)^2 = \frac{3+\cos(2\theta_A)}{4} \qquad (23)$$

or an error, i.e., the state $|0\rangle_S$, otherwise. Hence, Eve introduces an error with probability $(\sin^2\theta_A)/2$, which yields an expected error probability

$$\langle P_e\rangle = \frac{\sin^2\theta_A}{4} \qquad (24)$$

These are the same results as described in Section III-B above (cf. eq. (10)). Accordingly, performing the same computations as above, we obtain the mutual information I_{AE}, i.e., the information Eve is able to obtain about the raw key, as

$$I_{AE} = 1 - H = 1 - \frac{1}{2}\,h\Big(\cos^2\frac{\theta_A}{2}\Big) \qquad (25)$$

which is equal to the general result in eq. (13) from Section III-B. Hence, we can conclude that for the BBM protocol the optimal choice is a basis transformation about an angle $\theta_A = \frac{\pi}{2}$, i.e., the Hadamard operation.

V. Application on Song's QKD Protocol

In 2004, Song published a QKD scheme based on entanglement swapping, which is supposed to spare alternative measurements [17]. In this scheme Song uses a rather unusual basis transformation (compared to the Hadamard operation most commonly used in other protocols) with $\theta = 2\pi/3$. Hence, based on the discussions in the previous sections it is indicated that the security of the protocol can be further increased by using a different angle θ.

A. Protocol Description

In each round of the protocol, Alice and Bob prepare two qubits in their laboratories, which are either in the Bell basis or in a transformed basis. The transformation is done by the operation $T = T_x(2\pi/3)$, which is denoted in matrix form as

$$T = \frac{1}{2}\begin{pmatrix} 1 & \sqrt{3} \\ \sqrt{3} & -1 \end{pmatrix} \qquad (26)$$

Alice and Bob prepare random Bell states and then randomly choose between applying $\mathbb{1}$ or T onto qubit 2 and 4, respectively, in their possession. The application of T changes $|\Phi^\pm\rangle$ to $|\eta^\pm\rangle$ and $|\Psi^\pm\rangle$ to $|\nu^\pm\rangle$, where the state in the alternative basis are denoted as

$$\begin{aligned} |\eta^\pm\rangle &= \frac{1}{2}|\Phi^\mp\rangle + \frac{\sqrt{3}}{2}|\Psi^\pm\rangle \\ |\nu^\pm\rangle &= \frac{\sqrt{3}}{2}|\Phi^\pm\rangle - \frac{1}{2}|\Psi^\mp\rangle. \end{aligned} \qquad (27)$$

For our further discussion suppose that Alice prepares $|\Psi^+\rangle_{12}$ and Bob prepares $|\Phi^-\rangle_{34}$. Additionally, Bob applies T onto qubit 4 such that $|\Phi^-\rangle_{34}$ is changed into $|\eta^-\rangle_{34}$ (cf. (1) and (2) in Figure 6). The two parties exchange qubits 2 and 4 and publicly confirm the arrival of the respective qubit.

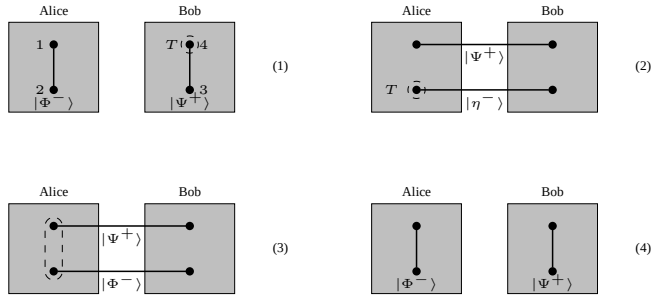


Fig. 6. Illustration of the protocol presented by Song [17]. Here, only Bob applies the basis transformation onto his qubit.

Before measuring, Alice and Bob announce publicly whether they applied the basis transformation T or not. If one party performed the basis transformation, the other party reverses the transformation by applying T onto the received qubit. In our case Alice applies T on qubit 4 (cf. (2) in Figure 6). Then, both parties perform Bell state measurements on the qubits in their possession. Based on their own outcome of the Bell state measurement both parties can compute each other's result. Following our example, if Alice obtains $|\Phi^-\rangle_{14}$, Bob obtains $|\Psi^+\rangle_{23}$.

B. Security Analysis

Song discussed a basic version of an intercept-resend attack as well as the ZLG attack [23] in his article [17] and showed in principle that the protocol is secure against this kind of attack. Nevertheless, he gave no expected error rate or mutual information for Eve, which would be of great interest since the operation T is an unusual basis transformation by an angle of $2\pi/3$ and is different from the more common choice of the Hadamard operation. Hence, we are going to look at these values in detail in the next paragraphs.

Due to arguments discussed in Section III above, we can immediately show that Song's protocol is completely open to the simulation attack when Alice does not apply the transformation T. In this case, Alice and Bob just perform the entanglement swapping and Eve can intercept qubits 2 and 4 in transit. As it is described in detail above, Eve distributes the state $|\delta\rangle$ from eq. (2) between Alice, Bob and herself using entanglement swapping and sends qubits Q to Bob and S to Alice, respectively (cf. (1) in Figure 7). When Alice and Bob perform their Bell state measurements, the correlation between their results is preserved due to the structure of the state $|\delta\rangle$. After Alice and Bob are finished Eve is able to obtain full information about Alice's and Bob's secret measurement based on the state of qubits T and U in her possession.

If either Alice or Bob performs the transformation T, we have the scenario described in Section III. Eve is not able to compensate the random application of the transformation while still preserving the correlation when T is not applied. Hence, Eve's intervention introduces an error, i.e., the parties do not obtain correlated results all the time. Taking the example from Section III above, Bob applies T onto qubit 4 and therefore Alice also applies T onto qubit S she receives from

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Eve (cf. (2) in Figure 7). When Alice obtains $|\Phi^-\rangle_{1S}$ from her measurement Bob obtains the correlated result $|\Psi^+\rangle_{23}$ only with probability $5/8$. In other words, Eve introduces an error with probability $3/8$, which leads to an expected error probability for this scenario of

$$\langle P_e \rangle = \frac{1}{4} \sin^2 \frac{2\pi}{3} = \frac{3}{16} \tag{28}$$

which is significantly lower than $1/4$. Hence, Eve has a better opportunity to eavesdrop the key in this protocol than, for example, in the revised version of the Cabello protocol [15] or the protocol by Li et al [18]. Due to the fact that the transformation T maps onto an unbiased superposition of states (cf. eq. (27) above) Eve is able to extract more information than usual from her attack strategy. The Shannon entropy for the simulation attack on Song's protocol is

$$H = \frac{1}{2}\, h\Big(\cos^2 \frac{\pi}{3}\Big) = \frac{1}{8}\Big(2 + 3 \log \frac{4}{3}\Big) \tag{29}$$

which further leads to Eve's mutual information

$$I_{AE} = 1 - H\big(S|M\big) \simeq 0.594 \tag{30}$$

Assuming that both parties perform the basis transformation T the protocol becomes insecure again. Due to Eve's entanglement swapping the operation T is brought from qubits 2 and 4 onto qubits 1 and 3, which leads to the state

$$T^{(1)} T^{(3)} |\delta\rangle_{1Q3STU} \tag{31}$$

When Alice and Bob apply the basis transformation T on qubits Q and S they receive from Eve, the state changes again into

$$T^{(1)} T^{(Q)} T^{(3)} T^{(S)} |\delta\rangle_{1Q3STU} \tag{32}$$

When Alice performs her Bell state measurement onto qubits 1 and S, it has the effect that the operations $T^{(1)}$ and $T^{(S)}$ are swapped onto qubits Q and 3 thus reverting the effect of T at Bob's side and re-establishing the state $|\delta\rangle$. Hence, Bob's measurement on qubits Q and 3 results into a state completely correlated to Alice's result. Further, Eve's qubits T and U are also correlated to Bob's result such that she has full information about the key when Alice and Bob announce their initial states.

The expected error probability from eq. (28) as well as the mutual information from eq. (30) indicate that the choice of $T = T_x(2\pi/3)$ is not optimal. Looking at Section III-B and eq. (10) and eq. (13) therein, we see that a basis rotation about an angle $\pi/2$, i.e., the Hadamard, instead of the operation T increases the expected error probability by $\simeq 33\%$ to $\langle P_e \rangle = 0.25$ and at the same time decreases the mutual information by $\simeq 16\%$ to $I_{AE} = 0.5$. Alternatively, a combined application of two basis transformations $T_x(\pi/2)$ and $T_x(\pi/4)$ by Alice and Bob further decreases the mutual information I_{AE}. As described in Section III-C two different basis rotations, randomly applied by Alice and Bob, leave the expected error probability $\langle P_e \rangle = 0.25$ but reduces Eve's information about the raw key by almost 25% to $I_{AE} = \simeq 0.45$ compared to the single application of T.

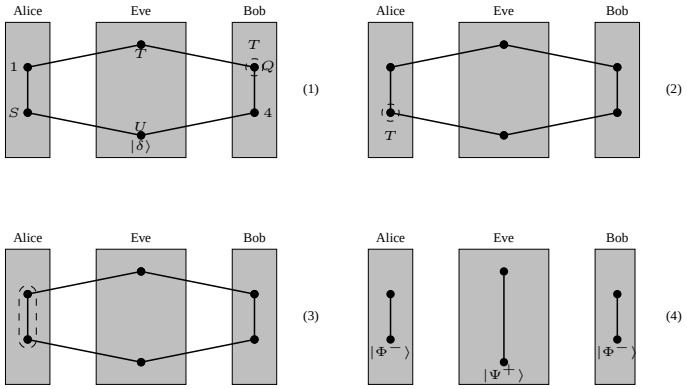


Fig. 7. Illustration of the simulation attack strategy on the protocol presented in [17]. Here, only Bob applies the basis transformation T onto qubit Q in his possession.

VI. Application on Cabello's QSS Protocol

In the year 2000, Cabello described a QSS protocol based on entanglement swapping [16]. The idea is to share a classical key between two parties, Bob and Charlie, such that they can communicate with Alice only if they collaborate and bring their shares together. The entanglement between the three parties is realized using a maximally entangled 3-qubit state, i.e., a GHZ state [24]. In our further discussions we will denote the GHZ states as

$$\begin{aligned}
|P_{00}^{\pm}\rangle &= \frac{1}{\sqrt{2}}\Big(|000\rangle \pm |111\rangle\Big) \\
|P_{01}^{\pm}\rangle &= \frac{1}{\sqrt{2}}\Big(|001\rangle \pm |110\rangle\Big) \\
|P_{10}^{\pm}\rangle &= \frac{1}{\sqrt{2}}\Big(|010\rangle \pm |110\rangle\Big) \\
|P_{11}^{\pm}\rangle &= \frac{1}{\sqrt{2}}\Big(|011\rangle \pm |100\rangle\Big)
\end{aligned} \tag{33}$$

The security of this protocol against the ZLG attack [23] has already been discussed by Lee, Lee, Kim, and Oh in [25]. They presented an adaption of the ZLG attack strategy, where the adversary Eve entangles herself with both Bob and Charlie using two Bell states. By intercepting the qubits coming from Alice and forwarding qubits from her Bell states, Eve is able to obtain Bob's and Charlie's secret measurement results. According to these results Eve is able to alter Alice's intercepted qubits such that her intervention is not detected.

In addition to their security analysis, Lee, Lee, Kim, and Oh presented a revised version of the protocol in [25], which includes the random application of Hadamard operation at Bob's and Charlie's laboratory. In the following paragraphs we are going to describe, how the simulation attack works on this protocol and whether the Hadamard operation is optimal in this context. We are going to show that using the simulation attack strategy the protocol is open to an attack to stress the fact that it is also applicable on QSS protocols.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE I. ALICE'S GHZ STATE AFTER BOB'S AND CHARLIE'S MEASUREMENT.

	$\|\Phi^+\rangle_{4A}$	$\|\Phi^-\rangle_{4A}$	$\|\Psi^+\rangle_{4A}$	$\|\Psi^-\rangle_{4A}$
$\|\Phi^+\rangle_{5B}$	$\|P^+_{00}\rangle_{1CD}$	$\|P^-_{00}\rangle_{1CD}$	$\|P^+_{10}\rangle_{1CD}$	$\|P^-_{10}\rangle_{1CD}$
$\|\Phi^-\rangle_{5B}$	$\|P^-_{00}\rangle_{1CD}$	$\|P^+_{00}\rangle_{1CD}$	$\|P^-_{10}\rangle_{1CD}$	$\|P^+_{10}\rangle_{1CD}$
$\|\Psi^+\rangle_{5B}$	$\|P^+_{01}\rangle_{1CD}$	$\|P^-_{01}\rangle_{1CD}$	$\|P^+_{11}\rangle_{1CD}$	$\|P^-_{11}\rangle_{1CD}$
$\|\Psi^-\rangle_{5B}$	$\|P^-_{01}\rangle_{1CD}$	$\|P^+_{01}\rangle_{1CD}$	$\|P^-_{11}\rangle_{1CD}$	$\|P^+_{11}\rangle_{1CD}$

A. Protocol Description

As already pointed out in the previous paragraph the original protocol by Cabello [16] is not secure and thus we will discuss the revised version given in [25] here. The revised version in general uses the Quantum Fourier Transformation (QFT) defined as

$$|j\rangle \overset{QFT}{\longmapsto} \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\Pi ijk/N} |k\rangle \tag{34}$$

to secure the qubits in transit (cf. for example [26] for details on the QFT). Since we are using qubits, the dimension $N = 2$ and the QFT reduces to the Hadamard operation for this special case. Therefore, we will use the Hadamard operation in the following considerations.

In this protocol, three parties are involved, which are able to distribute a key among them or share a secret between two of them. The aim is to use the 3-qubit entanglement of the GHZ state to achieve these tasks. Therefore, Alice, Bob, and Charlie are in possession of an entangled pair, i.e., $|\Phi^+\rangle_{12}$, $|\Phi^+\rangle_{4C}$, and $|\Phi^+\rangle_{5D}$, respectively. Further, Alice generates the GHZ state $|P^+_{00}\rangle_{3AB}$ at her side. She keeps qubit 3 of the GHZ state and sends qubits A and B to Bob and Charlie, respectively. At the same time, Bob and Charlie send their respective qubits C and D to Alice (cf. (1) in Figure 8). Additionally, Bob and Charlie randomly apply the Hadamard operation on qubits 4 and 5 still in their possession. After Alice received the qubits from Bob and Charlie she performs a Bell state measurement on qubits 2 and 3 and Bob and Charlie act similarly on their qubits 4 and A as well as 5 and B, respectively (cf. (2) in Figure 8). If both Bob and Charlie do not apply the Hadamard operation, the protocol is the same as in the original version by Cabello [16]. If either of them applies the Hadamard operation onto his qubit the GHZ state after Bob's measurement is altered as

$$\begin{aligned}\frac{1}{2}\Big(&|\Phi^+\rangle_{4A}\frac{1}{\sqrt{2}}\big(|P^-_{00}\rangle + |P^+_{10}\rangle\big)_{1CB}\\ &+|\Phi^-\rangle_{4A}\frac{1}{\sqrt{2}}\big(|P^+_{00}\rangle + |P^-_{10}\rangle\big)_{1CB}\\ &+|\Psi^+\rangle_{4A}\frac{1}{\sqrt{2}}\big(|P^-_{00}\rangle - |P^+_{10}\rangle\big)_{1CB}\\ &-|\Psi^-\rangle_{4A}\frac{1}{\sqrt{2}}\big(|P^+_{00}\rangle - |P^-_{10}\rangle\big)_{1CB}\Big)\end{aligned} \tag{35}$$

and similarly for Charlie's measurement (in this case the GHZ state changes to either $|P^{\pm}_{00}\rangle$ or $|P^{\pm}_{01}\rangle$). In case both parties

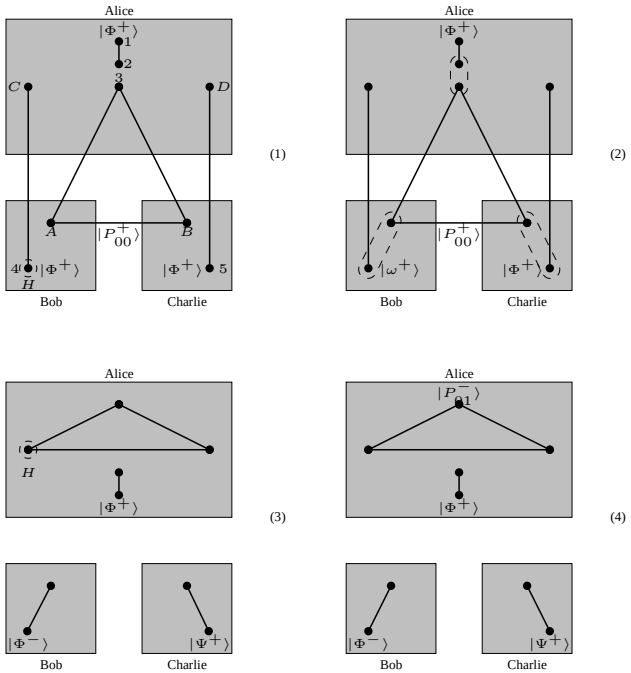


Fig. 8. Illustration of the QSS scheme described in [25].

apply the Hadamard operation the GHZ state changes into

$$\begin{aligned}\frac{1}{2}\Big(&|\Phi^+\rangle_{5B}\frac{1}{2}\big(|P^+_{00}\rangle + |P^-_{01}\rangle + |P^-_{10}\rangle + |P^+_{11}\rangle\big)_{1CD}\\ &+|\Phi^-\rangle_{5B}\frac{1}{2}\big(|P^-_{00}\rangle + |P^+_{01}\rangle + |P^+_{10}\rangle + |P^-_{11}\rangle\big)_{1CD}\\ &+|\Psi^+\rangle_{5B}\frac{1}{2}\big(|P^-_{00}\rangle - |P^+_{01}\rangle + |P^+_{10}\rangle - |P^-_{11}\rangle\big)_{1CD}\\ &-|\Psi^-\rangle_{5B}\frac{1}{2}\big(|P^+_{00}\rangle + |P^-_{01}\rangle - |P^-_{10}\rangle + |P^+_{11}\rangle\big)_{1CD}\Big)\end{aligned} \tag{36}$$

if Bob obtained $|\Phi^+\rangle_{4A}$ and equivalently for $|\Phi^-\rangle_{4A}$ and $|\Psi^{\pm}\rangle_{4A}$. Then, Bob and Charlie publicly announce their decision and Alice performs the Hadamard operation on the qubits she received from Bob and Charlie according to their decision (cf. (3) and (4) in Figure 8). Alice's Hadamard operation brings the GHZ state back to the state corresponding to the correlation described in Table I.

B. Security Analysis

Also in this case the strategy of the simulation attack is to find a state, which simulates the correlations given in Table I and provides Eve with additional information about Bob's and Charlie's measurement results. The version of the state $|\delta\rangle$ given in eq. (2) would be a possible choice, but not a very good one. A better version for $|\delta\rangle$ is

$$\begin{aligned}|\delta\rangle = \frac{1}{2}\Big(&|\Phi^+\rangle|\varphi_1\rangle|\delta_1\rangle\big) + |\Phi^-\rangle|\varphi_2\rangle|\delta_2\rangle\big)\\ &+|\Psi^+\rangle|\varphi_3\rangle|\delta_3\rangle\big) + |\Psi^-\rangle|\varphi_4\rangle|\delta_4\rangle\big)\Big)_{E_1-E_{11}}\end{aligned} \tag{37}$$

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

where $|\delta_1\rangle$ - $|\delta_4\rangle$ are defined as

$$\begin{aligned}
|\delta_1\rangle &= \frac{1}{2}\big(|\Phi^+\rangle|\varphi_5\rangle|P_{00}^+\rangle + |\Phi^-\rangle|\varphi_6\rangle|P_{00}^-\rangle \\
&\quad |\Psi^+\rangle|\varphi_7\rangle|P_{01}^+\rangle + |\Psi^-\rangle|\varphi_8\rangle|P_{01}^-\rangle\big) \\
|\delta_2\rangle &= \frac{1}{2}\big(|\Phi^+\rangle|\varphi_5\rangle|P_{00}^-\rangle + |\Phi^-\rangle|\varphi_6\rangle|P_{00}^+\rangle \\
&\quad +|\Psi^+\rangle|\varphi_7\rangle|P_{01}^-\rangle + |\Psi^-\rangle|\varphi_8\rangle|P_{01}^+\rangle\big) \\
|\delta_3\rangle &= \frac{1}{2}\big(|\Phi^+\rangle|\varphi_5\rangle|P_{10}^+\rangle + |\Phi^-\rangle|\varphi_6\rangle|P_{10}^-\rangle \\
&\quad +|\Psi^+\rangle|\varphi_7\rangle|P_{11}^+\rangle + |\Psi^-\rangle|\varphi_8\rangle|P_{11}^-\rangle\big) \\
|\delta_4\rangle &= \frac{1}{2}\big(|\Phi^+\rangle|\varphi_5\rangle|P_{10}^-\rangle + |\Phi^-\rangle|\varphi_6\rangle|P_{10}^+\rangle \\
&\quad +|\Psi^+\rangle|\varphi_7\rangle|P_{11}^-\rangle + |\Psi^-\rangle|\varphi_8\rangle|P_{11}^+\rangle\big)
\end{aligned} \tag{38}$$

Similarly to the auxiliary systems defined in Section II the states $|\varphi_1\rangle$ to $|\varphi_8\rangle$ have to fulfill

$$\begin{aligned}
\langle\varphi_i|\varphi_j\rangle &= 0 \qquad i,j \in \{1,...,4\} \;\; i \neq j \qquad \text{and} \\
\langle\varphi_i|\varphi_j\rangle &= 0 \qquad i,j \in \{5,...,8\} \;\; i \neq j
\end{aligned} \tag{39}$$

For reasons of simplicity we will assume that the states $|\varphi_i\rangle$ are 2-qubit states, since they are the smallest states fulfilling the equation above. Based on that, it can be immediately verified that this state simulates all possible correlations from Table I and that the qubit pairs E_3, E_4 and E_7, E_8 can be used to obtain full information about Bob's and Charlie's measurement results.

Focusing on an external adversary Eve, we assume again that she is able to distribute the state $|\delta\rangle$ between Alice, Bob, and Charlie using entanglement swapping. This means, Eve prepares the state $|\delta\rangle$ from eq. (37) and intercepts qubits A and B coming from Alice and performs a GHZ state measurement on them together with qubit E_9 (cf. (1) in Figure 9). Further, she intercepts qubits C and D coming from Bob and Charlie, respectively, and performs a Bell state measurement on the pairs E_1, C as well as E_5, D. Eve sends qubits E_2 to Bob, E_6 to Charlie and qubits E_{10} and E_{11} to Alice such that the state $|\delta\rangle$ is now distributed between all 4 parties. The definition of $|\delta\rangle$ indicates that Bob's and Charlie's measurements on the qubits in their possession yield random results but the respective qubits still in Eve's possession are in the same state, afterwards (cf. (3) in Figure 9). Additionally, the three qubits 3, E_{10} and E_{11} at Alice's laboratory are always in a correlated state to Bob's and Charlie's results. Assuming again that Bob obtained $|\Psi^+\rangle_{4E_2}$ and Charlie obtained $|\Phi^-\rangle_{5E_6}$, qubits 3, E_{10} and E_{11} are in the state $|P_{10}^-\rangle$, which corresponds to the state Alice expects to find if she obtains $|\Phi^+\rangle_{23}$ (cf. (4) in Figure 9 and also Table I). Also Alice's secret measurement on qubits 2 and 3 does not leave these three qubits in a state violating the expected correlation since her measurement changes the GHZ state accordingly.

In the revised version of Cabello's protocol, Bob and Charlie randomly apply a Hadamard operation on one qubit in their possession, which is not taken into account in the considerations above. If Bob applies the Hadamard operation on his qubit 4, the overall state $|\delta\rangle$ introduced by Eve described in eq. (37) above changes into

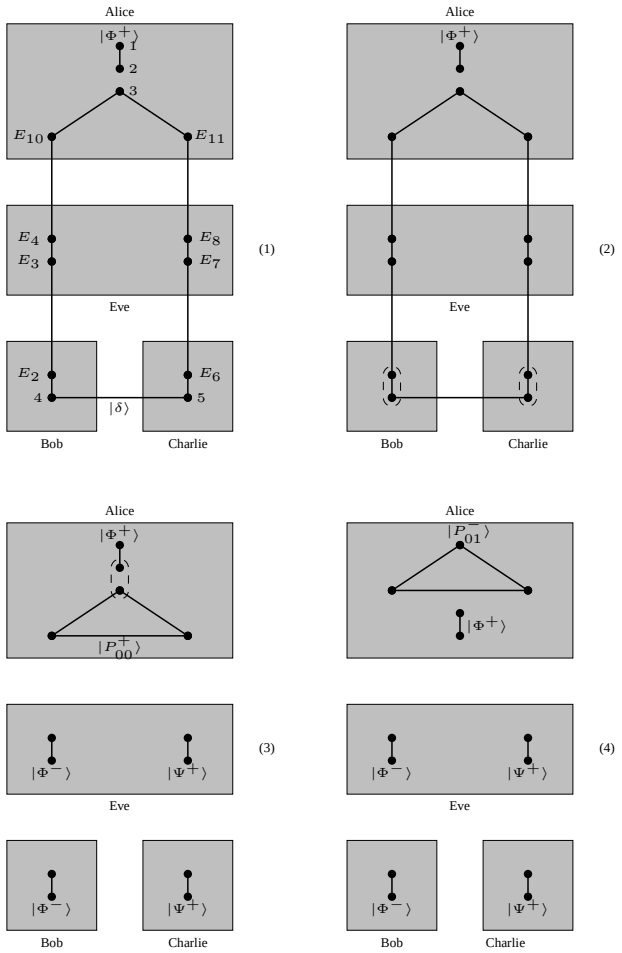


Fig. 9. Illustration of the simulation attack on the QSS scheme described in [25]. Here, no basis transformation is applied.

$$\begin{aligned}
\frac{1}{2\sqrt{2}}\Big(&|\Phi^+\rangle\Big(|\varphi_2\rangle|\delta_2\rangle + |\varphi_3\rangle|\delta_3\rangle\Big) \\
&+|\Phi^-\rangle\Big(|\varphi_1\rangle|\delta_1\rangle - |\varphi_4\rangle|\delta_4\rangle\Big) \\
&+|\Psi^+\rangle\Big(|\varphi_1\rangle|\delta_1\rangle + |\varphi_4\rangle|\delta_4\rangle\Big) \\
&-|\Psi^-\rangle\Big(|\varphi_2\rangle|\delta_2\rangle - |\varphi_3\rangle|\delta_3\rangle\Big)\Big)_{E_1-E_{11}}
\end{aligned} \tag{40}$$

and similarly for Charlie's Hadamard operation on qubit 5. This affects Eve's as well as Alice's measurement results such that Eve is not able to stay undetected any more.

To have a more general view on the revised protocol, we assume that Bob and Charlie are not restricted to the Hadamard operation but apply a basis transformation $T_x(\theta_B)$ and $T_x(\theta_C)$. First, assuming that only Bob applied $T_x(\theta_B)$ operation the overall state changes into

$$\sin\frac{\theta_B}{2}|\varphi_1\rangle \otimes |\delta_1\rangle + \cos\frac{\theta_B}{2}|\varphi_4\rangle \otimes |\delta_4\rangle \tag{41}$$

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

if Bob's result is $|\Psi^+\rangle_{4E_2}$. Hence, at this time Eve obtains from a measurement on qubits E_3 and E_4 either $|\varphi_1\rangle_{E_3E_4}$ or $|\varphi_4\rangle_{E_3E_4}$ but both do not correspond to Bob's result. Thus, the best strategy for Eve is to delay her measurement until she knows whether Bob applied the basis transformation $T_x(\theta_B)$ or not, as described below. Similarly, if just Charlie applies $T_x(\theta_C)$ the overall state after Bob's result $|\Psi^+\rangle_{4E_2}$ is

$$|\varphi_3\rangle \otimes T_x(\theta_C)|\delta_3\rangle \tag{42}$$

with

$$\begin{aligned} &T_x(\theta_C)|\delta_3\rangle = \\ &\frac{1}{2}\Big[|\Phi^+\rangle\Big(\cos\frac{\theta_C}{2}|\varphi_6\rangle|P_{10}^-\rangle + \sin\frac{\theta_C}{2}|\varphi_7\rangle|P_{11}^+\rangle\Big) \\ &+|\Phi^-\rangle\Big(\cos\frac{\theta_C}{2}|\varphi_5\rangle|P_{10}^+\rangle + \sin\frac{\theta_C}{2}|\varphi_8\rangle|P_{11}^-\rangle\Big) \\ &+|\Psi^+\rangle\Big(\cos\frac{\theta_C}{2}|\varphi_8\rangle|P_{11}^-\rangle + \sin\frac{\theta_C}{2}|\varphi_5\rangle|P_{10}^+\rangle\Big) \\ &+|\Psi^-\rangle\Big(\cos\frac{\theta_C}{2}|\varphi_7\rangle|P_{11}^+\rangle + \sin\frac{\theta_C}{2}|\varphi_6\rangle|P_{10}^-\rangle\Big)\Big] \end{aligned} \tag{43}$$

In this case, Eve obtains the same result as Bob but further on her measurement on qubits E_7E_8 yields a result uncorrelated to Charlie's measurement outcome due to his basis transformation. In the last case where both Bob and Charlie apply their basis transformations $T_x(\theta_B)$ and $T_x(\theta_C)$, respectively, the overall state changes to

$$\sin\frac{\theta_B}{2}|\varphi_1\rangle \otimes T_x(\theta_C)|\delta_1\rangle + \cos\frac{\theta_B}{2}|\varphi_4\rangle \otimes T_x(\theta_C)|\delta_4\rangle \tag{44}$$

in case Bob obtains $|\Psi^+\rangle_{4E_2}$ from his measurement. From eq. (43) above we can see that after Charlie's measurement the state of the remaining qubits is

$$\begin{aligned} &\sin\frac{\theta_B}{2}|\varphi_1\rangle\Big(\cos\frac{\theta_C}{2}|\varphi_5\rangle|P_{00}^+\rangle + \sin\frac{\theta_C}{2}|\varphi_8\rangle|P_{01}^-\rangle\Big) \\ &+\cos\frac{\theta_B}{2}|\varphi_4\rangle\Big(\cos\frac{\theta_C}{2}|\varphi_5\rangle|P_{10}^-\rangle + \sin\frac{\theta_C}{2}|\varphi_8\rangle|P_{11}^+\rangle\Big) \end{aligned} \tag{45}$$

assuming Charlie obtains $|\Phi^-\rangle_{5E_6}$. It is described in eq. (45) that Eve's results are completely uncorrelated to the two secret results of Bob and Charlie. Thus, the optimal strategy for Eve is to delay her measurements on qubits E_3E_4 and E_7E_8 until Bob and Charlie finished their measurements and publicly announce their choice regarding the application of the Hadamard operation. Eve performs the measurement on her qubit pairs afterwards, obtaining Bob's and Charlie's result only with a certain probability.

In all three cases discussed in the previous paragraphs, Alice applies the operation $T_x(\theta_B)$ on qubits E_{10} and operation $T_x(\theta_C)$ on E_{11}, respectively, to reverse the effect of Bob's and Charlie's operations. This changes the GHZ state into a superposition of GHZ states. Hence, Alice obtains a GHZ state corresponding to Bob's and Charlie's secrets only to a certain amount. Following our example where only Bob used the Hadamard operation as described in eq. (41) we see after a little calculation that for Charlie's result $|\Phi^-\rangle_{5E_6}$ the state of the remaining qubits is

$$\begin{aligned} &\sin\frac{\theta_B}{2}|\varphi_1\rangle_{E_3E_4}|\varphi_6\rangle_{E_7E_8}|P_{00}^-\rangle_{1E_{10}E_{11}} \\ &+\cos\frac{\theta_B}{2}|\varphi_4\rangle_{E_3E_4}|\varphi_6\rangle_{E_7E_8}|P_{10}^+\rangle_{1E_{10}E_{11}} \end{aligned} \tag{46}$$

Therefore, Alice obtains the GHZ state correlated to Bob's and Charlie's result only with a certain probability. Hence, Eve's intervention introduces on average an error rate of

$$\big\langle P_e\big\rangle = \frac{1}{4}\sin^2\theta_B + \frac{1}{4}\sin^2\theta_C - \frac{1}{16}\sin^2\theta_B\sin^2\theta_C \tag{47}$$

Furthermore, Eve's results are correlated to Bob's and Charlie's results only with a certain probability such that she is not able to obtain much information about Alice's secret. In detail, the Shannon entropy for Alice, Bob, and Charlie is

$$H = \frac{7}{16}\Big(h\big(\cos^2\theta_B\big) + h\big(\cos^2\theta_C\big)\Big) \tag{48}$$

When looking at Figure 10 and Figure 11 we see that the average error probability $\big\langle P_e\big\rangle$ as well as the Shannon entropy H have their maximum when $\theta_B = \theta_C = \pi/2$, i.e., the optimal choice for the basis transformation is the Hadamard operation. In this case,

$$\big\langle P_e\big\rangle = \frac{1}{4} + \frac{1}{4} - \frac{1}{16} = \frac{7}{16} \tag{49}$$

and

$$H = \frac{7}{16}\Big(h\Big(\frac{1}{2}\Big) + h\Big(\frac{1}{2}\Big)\Big) = \frac{7}{8} \tag{50}$$

and thus both values are much larger compared to the results from previous sections. Accordingly, Eve's mutual information is rather low at

$$I_{AE} = 1 - H = \frac{1}{8} \tag{51}$$

compared to the results from above.

A scenario dealing with an adversary from the inside, i.e., Charlie as malicious party who wants to obtain Alice's secret without the help of Bob, is a more severe threat for a QSS protocol. Here, Charlie also prepares the state $|\delta\rangle$ from eq. (37) instead of his Bell state and intercepts the qubits coming from Alice and Bob. He performs a GHZ state measurement on A, B and E_9 as well as a Bell state measurement on E_1 and C to entangle himself with Alice and Bob. Then, he forwards qubits E_{10}, E_{11} to Alice and E_2 to Bob and jointly measures his qubits E_5 and E_6. We have to remark that in this case with the adversary coming from the inside, qubits E_7 and E_8 of the state $|\delta\rangle$ can be ignored since Charlie is, of course, fully aware of his own secret measurement result. Whenever Bob does not use the basis transformation $T_x(\theta_B)$ we have already seen that qubits E_3 and E_4 in Charlie's possession are perfectly correlated to Bob's result giving Charlie full information about Bob's result. We already showed that based on the structure of the state $|\delta\rangle$ the three qubits in Alice's possession are always in a GHZ state corresponding to Bob's and Charlie's secret results.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

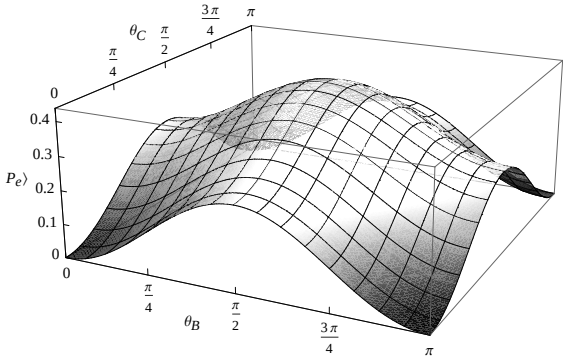


Fig. 10. Eve's expected error probability $\langle P_e \rangle$ if both parties apply a basis transformation with the respective angles θ_B and θ_C.

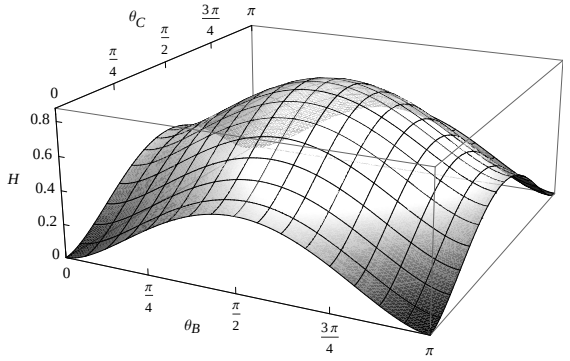


Fig. 11. Eve's Shannon entropy $\langle P_e \rangle$ if both parties apply a basis transformation with the respective angles θ_B and θ_C.

Whenever Bob chooses to use the basis transformation $T_x(\theta_B)$ the exact state of the remaining qubits is of the form described in eq. (41), if he obtained $|\Psi^+\rangle_{4E_2}$. Since Charlie is fully aware of his measurement results the scenario is equal to the attack of an external adversary if only Bob applies the basis transformation. Therefore, based on the calculations above, we see that Eve introduces an average error rate

$$\langle P_e \rangle = \frac{1}{4} \sin^2 \theta_B \tag{52}$$

similar to the probability in eq. (10) above. Hence, $\langle P_e \rangle$ becomes maximal with $\theta_B = \pi/2$ such that

$$\langle P_e \rangle = \frac{1}{4} \tag{53}$$

Accordingly, the Shannon entropy for Alice and Bob is

$$H = \frac{1}{2}\, h\Big(\cos^2 \frac{\theta_B}{2}\Big) \tag{54}$$

also taking its maximum with $\theta_B = \pi/2$ such that

$$H = \frac{1}{2}\, h\Big(\frac{1}{2}\Big) = \frac{1}{2} \tag{55}$$

leaving Eve's mutual information at

$$I_{AE} = 1 - H = 1 - \frac{1}{2} = \frac{1}{2} \tag{56}$$

which is equal to the results from the previous sections.

VII. Conclusion and Further Research

In this article, we discussed the optimality of basis transformations to secure entanglement swapping based QKD protocols. Starting from a generic entanglement swapping scenario, we used a collective attack strategy to analyze the amount of information an adversary is able to obtain. We showed that in case only one party applies a basis transformation, the operation $T_x(\theta_A)$ reduces to the Hadamard operation, i.e., the angle $\theta_A = \pi/2$ allows a maximal mutual information of $I_{AE} = 0.5$. Whereas, the main result of this article is the fact that if both parties apply a transformation, the optimal choice for the angles θ_A and θ_B describing the basis transformations is $\theta_A = \pi/4$ and $\theta_B = \pi/2$. As a consequence, this decreases the mutual information of an adversary further to $I_{AE} \sim 0.45$, which improves the security.

Additionally, we discussed 3 different protocols, the BBM protocol [5], Song's QKD protocol [17] and Cabello's QSS protocol [16] to show how the simulation attack is applied on various kinds of protocols. We showed that for the BBM protocol the optimal angle for the basis transformation is $\pi/2$, i.e., the Hadamard operation, due to the fact that no entanglement swapping is performed and a measurement on only one entangled state is applied. Nevertheless, the simulation attack describes the most general collective attack strategy on this kind of protocol.

Regarding Song's QKD protocol we were able to show that the basis transformation by an angle $2\pi/3$ is by no means optimal. Using the results from the simulation attack, the optimal choice for a basis rotation is to use two different angles $\pi/2$ and $\pi/4$ to reduce Eve's mutual information about the raw key by about 25% from 0.594 to $\simeq 0.45$ and thus increasing the security.

Looking at a QSS protocol instead of a key distribution protocol we examined the application of the simulation attack on Cabello's QSS protocol. In this case, the optimal angle for the basis transformation is again $\pi/2$, i.e., the Hadamard operation. This is true for Bob's and Charlie's basis transformation since both operations act separately on the GHZ state in Alice's possession. Nevertheless, the average error probability and Alice's, Bob's, and Charlie's Shannon entropy are rather high with $\langle P_e \rangle = 7/16$ and $H = 7/8$, respectively, for an adversary from the outside. Dealing with an adversary form the inside, i.e., a malicious Charlie, $\pi/2$ is still optimal. This reduces the average error probability and the Shannon entropy to the more common $\langle P_e \rangle = 1/4$ and $H = 1/2$, respectively, because Charlie has to cope with Bob's basis transformation alone.

The next questions arising directly from these results are how, if at all, the results change if basis transformations from the Z- into the Y-basis are applied. A first inspection shows that such basis transformations can not be plugged in directly into this framework. Hence, besides the transformation from the Z- into the Y- basis, the effects of the simpler rotation

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

operations on the results have to be inspected during further research. Since basis transformations can be described in terms of rotation operations it could be easier to apply rotation operations in this framework. Nevertheless, due to the similar nature of basis transformations and rotation operations it can be assumed that the results will be comparable to the results presented here.

To keep the setting as general as possible, a further main goal is to allow Alice and Bob to use arbitrary unitary operations instead of just basis transformations to secure the protocol. This should make it even more difficult for Eve to gain information about the raw key.

ACKNOWLEDGMENTS

We would like to thank Christian Kollmitzer, Oliver Maurhart as well as Beatrix Hiesmayr and Marcus Huber for fruitful discussions and interesting comments.

REFERENCES

[1] S. Schauer and M. Suda, "Security of Entanglement Swapping QKD Protocols against Collective Attacks," in *ICQNM 2012 , The Sixth International Conference on Quantum, Nano and Micro Technologies*. IARIA, 2012, pp. 60–64.

[2] ——, "A Novel Attack Strategy on Entanglement Swapping QKD Protocols," *Int. J. of Quant. Inf.*, vol. 6, no. 4, pp. 841–858, 2008.

[3] C. H. Bennett and G. Brassard, "Public Key Distribution and Coin Tossing," in *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*. IEEE Press, 1984, pp. 175–179.

[4] A. Ekert, "Quantum Cryptography Based on Bell's Theorem," *Phys. Rev. Lett.*, vol. 67, no. 6, pp. 661–663, 1991.

[5] C. H. Bennett, G. Brassard, and N. D. Mermin, "Quantum Cryptography without Bell's Theorem," *Phys. Rev. Lett.*, vol. 68, no. 5, pp. 557–559, 1992.

[6] D. Bruss, "Optimal Eavesdropping in Quantum Cryptography with Six States," *Phys. Rev. Lett*, vol. 81, no. 14, pp. 3018–3021, 1998.

[7] A. Muller, H. Zbinden, and N. Gisin, "Quantum Cryptography over 23 km in Installed Under-Lake Telecom Fibre," *Europhys. Lett.*, vol. 33, no. 5, pp. 335–339, 1996.

[8] A. Poppe, A. Fedrizzi, R. Usin, H. R. Böhm, T. Lorünser, O. Maurhardt, M. Peev, M. Suda, C. Kurtsiefer, H. Weinfurter, T. Jennewein, and A. Zeilinger, "Practical Quantum Key Distribution with Polarization Entangled Photons," *Optics Express*, vol. 12, no. 16, pp. 3865–3871, 2004.

[9] A. Poppe, M. Peev, and O. Maurhart, "Outline of the SECOQC Quantum-Key-Distribution Network in Vienna," *Int. J. of Quant. Inf.*, vol. 6, no. 2, pp. 209–218, 2008.

[10] M. Peev, C. Pacher, R. Alléaume, C. Barreiro, J. Bouda, W. Boxleitner, T. Debuisschert, E. Diamanti, M. Dianati, J. F. Dynes, S. Fasel, S. Fossier, M. Fürst, J.-D. Gautier, O. Gay, N. Gisin, P. Grangier, A. Happe, Y. Hasani, M. Hentschel, H. Hübel, G. Humer, T. Länger, M. Legré, R. Lieger, J. Lodewyck, T. Lorünser, N. Lütkenhaus, A. Marhold, T. Matyus, O. Maurhart, L. Monat, S. Nauerth, J.-B. Page, A. Poppe, E. Querasser, G. Ribordy, S. Robyr, L. Salvail, A. W. Sharpe, A. J. Shields, D. Stucki, M. Suda, C. Tamas, T. Themel, R. T. Thew, Y. Thoma, A. Treiber, P. Trinkler, R. Tualle-Brouri, F. Vannel, N. Walenta, H. Weier, H. Weinfurter, I. Wimberger, Z. L. Yuan, H. Zbinden, and A. Zeilinger, "The SECOQC Quantum Key Distribution Network in Vienna," *New Journal of Physics*, vol. 11, no. 7, p. 075001, 2009.

[11] N. Lütkenhaus, "Security Against Eavesdropping Attacks in Quantum Cryptography," *Phys. Rev. A*, vol. 54, no. 1, pp. 97–111, 1996.

[12] ——, "Security Against Individual Attacks for Realistic Quantum Key Distribution," *Phys. Rev. A*, vol. 61, no. 5, p. 052304, 2000.

[13] P. Shor and J. Preskill, "Simple Proof of Security of the BB84 Quantum Key Distribution Protocol," *Phys. Rev. Lett.*, vol. 85, no. 2, pp. 441–444, 2000.

[14] A. Cabello, "Quantum Key Distribution without Alternative Measurements," *Phys. Rev. A*, vol. 61, no. 5, p. 052312, 2000.

[15] ——, "Reply to "Comment on "Quantum Key Distribution without Alternative Measurements"","," *Phys. Rev. A*, vol. 63, no. 3, p. 036302, 2001.

[16] ——, "Multiparty Key Distribution and Secret Sharing Based on Entanglement Swapping," *quant-ph/0009025 v1*, 2000.

[17] D. Song, "Secure Key Distribution by Swapping Quantum Entanglement," *Phys. Rev. A*, vol. 69, no. 3, p. 034301, 2004.

[18] C. Li, Z. Wang, C.-F. Wu, H.-S. Song, and L. Zhou, "Certain Quantum Key Distribution achieved by using Bell States," *International Journal of Quantum Information*, vol. 4, no. 6, pp. 899–906, 2006.

[19] C. H. Bennett, G. Brassard, C. Crepeau, R. Jozsa, A. Peres, and W. K. Wootters, "Teleporting an Unknown Quantum State via Dual Classical and EPR Channels," *Phys. Rev. Lett.*, vol. 70, no. 13, pp. 1895–1899, 1993.

[20] M. Zukowski, A. Zeilinger, M. A. Horne, and A. K. Ekert, ""Event-Ready-Detectors" Bell State Measurement via Entanglement Swapping," *Phys. Rev. Lett.*, vol. 71, no. 26, pp. 4287–4290, 1993.

[21] B. Yurke and D. Stolen, "Einstein-Podolsky-Rosen Effects from Independent Particle Sources," *Phys. Rev. Lett.*, vol. 68, no. 9, pp. 1251–1254, 1992.

[22] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt, "Proposed Experiment to Test Local Hidden-Variable Theories," *Phys. Rev. Lett.*, vol. 23, no. 15, pp. 880–884, 1969.

[23] Y.-S. Zhang, C.-F. Li, and G.-C. Guo, "Comment on "Quantum Key Distribution without Alternative Measurements"," *Phys. Rev. A*, vol. 63, no. 3, p. 036301, 2001.

[24] D. Greenberger, M. A. Horne, and A. Zeilinger, "Going beyond Bell's Theorem," in *Bell's Theorem, Quantum Theory and Conceptions of the Universe*, M. Kafatos, Ed. Kluwer, 1989, pp. 69–72.

[25] J. Lee, S. Lee, J. Kim, and S. D. Oh, "Entanglement Swapping Secures Multiparty Quantum Communication," *Phys. Rev. A*, vol. 70, no. 3, p. 032305, 2004.

[26] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Maximizing Utilization in Private IaaS Clouds with Heterogenous Load through Time Series Forecasting

Tomáš Vondra and Jan Šedivý
Dept. of Cybernetics, Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27
Prague, Czech Republic
vondrto6@fel.cvut.cz, sedivja2@fel.cvut.cz

***Abstract*—This document presents ongoing work on creating a computing system that can run two types of workloads on a private cloud computing cluster, namely web servers and batch computing jobs, in a way that would maximize utilization of the computing infrastructure. To this end, a queue engine called Cloud Gunther has been developed. This application improves upon current practices of running batch computations in the cloud by integrating control of virtual machine provisioning within the job scheduler. For managing web server workloads, we present ScaleGuru, which has been modeled after Amazon Auto Scaler for easier transition from public to private cloud. Both these tools are tested to run over the Eucalyptus cloud system. Further research has been done in the area of Time Series Forecasting, which enables to predict the load of a system based on past observations. Due to the periodic nature of the interactive load, predictions can be made in the horizon of days with reasonable accuracy. Two forecasting models (Holt-Winters exponential smoothing and Box-Jenkins autoregressive) have been studied and evaluated on six server load time series. The autoscaler and queue engine are not yet integrated. Meanwhile, the prediction can be used to decide how many servers to turn off at night or as an internal component for the autoscaling system.**

Keywords - Cloud Computing; Automatic Scaling; Job Scheduling; Real-time Infrastucture; Time Series Forecasting.

I. INTRODUCTION

This paper is an extension of conference article [1].

According to Gartner [2], private cloud computing is currently at the top of the technology hype; but, its popularity is bound to fall due to general disillusionment.

Why? While the theoretical advantages of cloud computing are widely known – private clouds build on the foundations of virtualization technology and add automation, which should result in savings on administration while improving availability. They provide elasticity, which means that an application deployed to the cloud can dynamically change the amount of resources it uses. Another connected term is agility, meaning that the infrastructure can be used for multiple purposes depending on current needs. Lastly, the cloud should provide self-service, so that the customer can provision his infrastructure at will, and pay-per-use, so he will pay exactly for what he consumed.

The problem is that not all of these features are present in current products that are advertised as private clouds. Specifically, this document will deal with the problem of infrastructure agility.

A private cloud can be used for multiple tasks, which all draw resources from a common pool. This heterogenous load can basically be broken down into two parts, interactive processes and batch processes. An example of the first are web applications, which are probably the major way of interactive remote computer use nowadays, the second could be related to scientific computations or, in the corporate world, data mining.

This division was chosen because of different service level measures used in both the fields. While web servers need to be running all the time and have response times in seconds, in batch job scheduling, the task deadlines are generally in units ranging from tens of minutes to days. This allows a much higher amount of flexibility in allocating resources to these kinds of workloads. In other words, while resources for interactive workloads need to always be provisioned in at least the amount required by the offered load, a job scheduler can decide on when and where to run tasks that are in its queue.

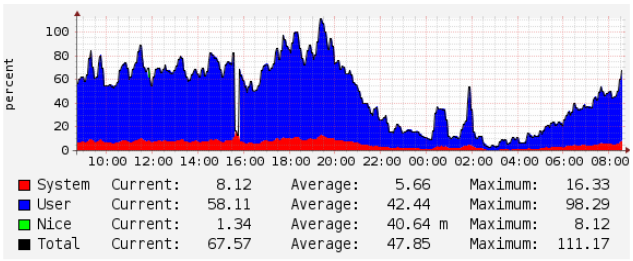


Figure 1. Daily load graph of an e-business website [3]

When building a data center, which of course includes private clouds, the investor will probably want to ensure that it is utilized as much as possible. The private cloud can help achieve that, but not when the entire load is interactive. This is due to the fact that interactive load depends on user activity, which varies throughout the day, as seen in Figure 1.

In our opinion, the only way to increase the utilization of a private cloud is to introduce non-interactive tasks that will fill in the white parts of the graph, i.e., capacity left unused by interactive traffic (which of course needs to have priority over batch jobs).

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

HPC (High Performance Computing) tasks are traditionally the domain of grid computing. Lately, however, they also began to find their way into the cloud. Examples may be Google's data mining efforts in their private cloud or Amazon's Elastic MapReduce public service [4]. The grid also has the disadvantage that it is only usable for batch and parallel jobs, not interactive use.

Currently, there is not much support for running of batch jobs on private clouds. The well-known scheduling engines Condor [5] and SGE (Sun Grid engine) [6] both claim Amazon EC2 (Elastic Compute Cloud) [7] compatibility, they however cannot control the cloud directly, they only use resources provisioned by other means (See Section II.). (SGE seems to be able to control cloud instances in a commercial fork by Univa, though [8].)

That is why the Cloud Gunther project was started. It is a web application that can run batch parallel and pseudoparallel jobs on the Eucalyptus private cloud [9]. The program does not only run tasks from its queue; it can also manage the VM (virtual machine) instances the tasks are to be run on.

What the application currently lacks is support for advanced queuing schemes (only Priority FCFS (First Come First Served) has been implemented). Further work will include integration of a better queuing discipline, which will be capable of maximizing utilization of the cloud computing cluster by reordering the tasks as to reduce the likelihood of one task waiting for others to complete, while there are unused resources in the cluster, effectively creating a workflow of tasks (see Section V).

The goal is that the scheduler will be fed with data about the likely amount of free resources left on the cluster by interactive processes several hours into the future by a predictor. This will ensure that the cluster is always fully loaded, but the interactive load is never starved for resources.

Prediction of load or any other quantity in time is studied in a branch of statistics called Time Series Analysis and Forecasting. This discipline has also been studied as part of this project and first results are presented in this paper.

This document has five sections. After Section I, Introduction, comes Section II, Related Work, which will present the state of the art in the area of grid schedulers and similar cloud systems. Section III, Cloud Technology, summarizes progress done in cloud research at the Dept. of Cybernetics, mainly on the ScaleGuru autoscaler and the Cloud Gunther job scheduler. Section IV, Time Series, deals with the possibilities for load prediction and evaluates two forecasting methods on server load data. Section V, Future Work, outlines the plans for expansion of the scheduler, mainly to accommodate heterogenous load on the cloud computing cluster. Section VI, Conclusion, ends the paper.

II. Related Work

As already stated, the most notable job control engines in use nowadays are probably SGE [6] and Condor [5]. These were developed for clusters and thus lack the support of dynamic allocation and deallocation of resources in cloud environments.

There are tools that can allocate a complete cluster for these engines, for example StarCluster for SGE [10]. The drawback of this solution is that the management of the cloud is split in two parts – the job scheduler, which manages the instances currently made available to it (in an optimal fashion, due to the experience in the grid computing field), and the tool for provisioning the instances, which is mostly manually controlled.

This is well illustrated in an article on Pandemic Influenza Simulation on Condor [11]. The authors have written a web application, which would provision computing resources from the Amazon cloud and add them to the Condor resource pool. The job scheduler could then run tasks on them. The decision on the number of instances was however left to the users.

A similar approach is used in the SciCumulus workflow management engine, which features adaptive cloud-aware scheduling [12]. The scheduler can react to the dynamic environment of the cloud, in which instances can be randomly terminated or started, but does not regulate their count by itself.

The Cloud Gunther does not have this drawback, as it integrates job scheduling with instance provisioning. This should guarantee that there is no unused time between the provisioning of a compute resource and its utilization by a task, and that the instances are terminated immediately when they are no longer needed.

A direct competitor to Cloud Gunther is Cloud Scheduler [13]. From the website, it seems to be a plug-in for Condor, which can manage VM provisioning for it. Similar to Cloud Gunther, it is fairly new and only features FCFS queuing.

An older project of this sort is Nephele [14], which focuses on real-time transfers of data streams between jobs that form a workflow. It provisions different-sized instances for each phase of the workflow. In this system, the number and type of machines in a job are defined upfront and all instances involved in a step must run at once, so there is little space for optimization in the area of resource availability and utilization.

Aside from cluster-oriented tools, desktop grid systems are also reaching into the area of clouds. For example, the Aneka platform [15] can combine resources from statically allocated servers, unused desktop computers and Amazon Spot instances. It can provision the cloud instances when they are needed to satisfy job deadlines. Certainly, this system seems more mature than Cloud Gunther and has reached commercial availability.

None of these systems deals with the issue of resource availability in private clouds and fully enjoys the benefits of the illusion of infinite supply. To the best of our knowledge, no one has yet dealt with the problem of maximizing utilization of a cloud environment that is not fully dedicated to HPC and where batch jobs would have the status of "filler traffic."

As to time series forecasting, there are efforts to use it on Grids, such as the Network Weather Service (NWS) referenced in a paper by Yang, Foster, and Schopf [16], who describe a better forecasting method for it. The method seems much simpler than the ones being applied in this

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

article. The NWS project seems to be no longer active, though.

The problems on grids are different from those in clouds. In clouds, we discuss automatic scaling of web servers on identical hardware and data center utilization, whereas in grids, the main problems are prediction of task execution times on heterogenous machines, as described by Iverson, Özgüner, and Potter in [17], and queue wait times and job interarrival times, discussed by Li in [18].

III. CLOUD TECHNOLOGY

A. Eucalyptus

Eucalyptus [9] is the cloud platform that is used for experiments at the Dept. of Cybernetics. It is an open-source implementation of the Amazon EC2 industry standard API (Application Programming Interface) [7]. It started as a research project at the University of California and evolved to a commercial product.

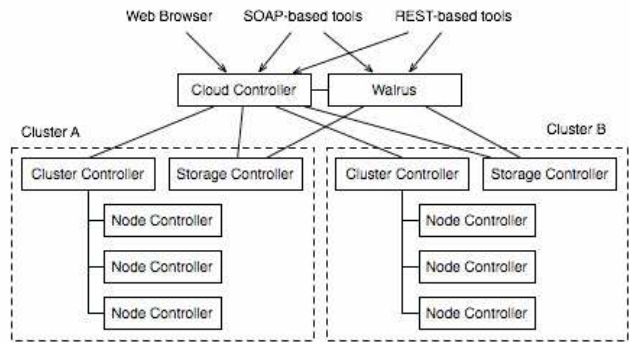


Figure 2. Eucalyptus architecture [9]

It is a distributed system consisting of five components. Those are the Node Controller (NC), which is responsible of running virtual machines from images obtained from the Walrus (Amazon S3 (Simple Storage Service) implementation). Networking for several NCs is managed by a Cluster Controller (CC), and the Cloud Controller (CLC) exports all external APIs and manages the cloud's operations. The last component is the Storage Controller (SC), which exports network volumes, emulating the Amazon EBS (Elastic Block Store) service. The architecture can be seen in Figure 2.

Our Eucalyptus setup consists of a server that hosts the CLC, SC and Walrus components and is dedicated to cloud experiments. The server manages 20 8-core Xeon workstations, which are installed in two labs and 1/4 of their capacity can be used for running VM instances through Eucalyptus NCs. A second server, which is primarily used to provide login and file services to students and is physically closer to the labs, is used to host Eucalyptus CC.

The cloud is used for several research projects at the Cloud Computing Center research group [19]. Those are:

- Automatic deployment to PaaS (Platform as a Service), a web application capable of automatic deployment of popular CMS (Content Management Systems) to PaaS. [20]
- ScaleGuru, an add-on for private clouds, which adds automatic scaling and load balancing support for web applications. [21]
- Cloud Gunther, a web application that manages a queue of batch computational jobs and runs them on Amazon EC2 compatible clouds.

Aside from this installation of Eucalyptus, we also have experience deploying the system in a corporate environment. An evaluation has been carried out in cooperation with the Czech company Centrum. The project validated the possibility of deploying one of their production applications as a machine image and scaling the number of instances of this image depending on current demand. A hardware load-balancer appliance from A10 Networks was used in the experiment and the number of instances was controlled manually as private infrastructure clouds generally lack the autoscaling capabilities of public clouds.

B. ScaleGuru

The removal of this shortcoming is the target of the ScaleGuru project [21], an autoscaling system that can be deployed in a virtual machine in a private IaaS cloud and is able to automatically manage instances of other applications on it.

The software is written in Node.JS with the MongoDB database. It is closely modeled after Amazon Auto Scaling [22], so that users familiar with its structure will easily learn to use ScaleGuru. Therefore, its data model contains Autoscaling Groups, which place lower and upper limits on the number of started instances. Launch Configs then specify the image of the managed application and its parameters. Load Balancers manage the hostnames of the managed services and balanced ports. Lastly, there are Autoscaling Policies and Autoscaling Alarms, which together form the scaling rules such as: "If the CPU Utilization was over 80% for 2 minutes, launch 1 more instance." Using multiple rules, it is possible to create a dynamic response curve.

The program consists of four parts, which are easily replaceable. The Application Core implements the autoscaling logic. It uses the Monitoring component to provide input. Currently it supports collection of CPU utilization, disk and network throughput using an agent on the managed instances. This has the advantage that it is not hypervisor-dependent, but requires the user's cloud API key so that the agent can be injected. If implemented as a service on the private cloud, this is not a problem and has the advantage that the user can sign in to the autoscaler using these keys.

The scaling decisions are implemented through the Cloud Controller component, which supports Amazon EC2 compatible clouds and was tested on Eucalyptus. It can track the state of launched instances and can retry launching on failure. All errors are logged to the web interface. Launched instances are added to Nginx configuration through the Load Balancer Controller.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

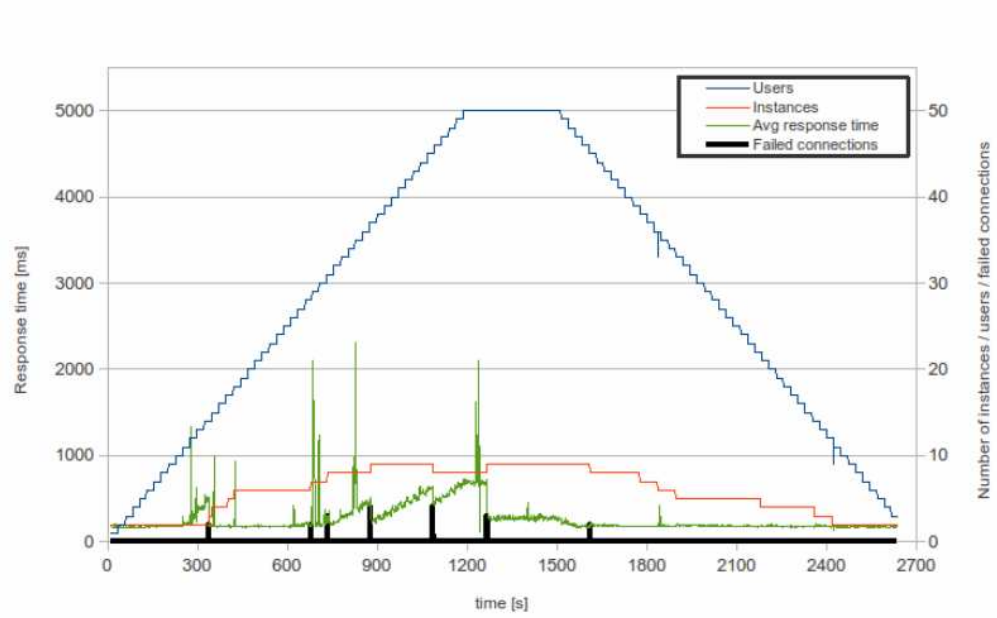


Figure 3. ScaleGuru evaluation [21]

The software was evaluated in a lab setup with Wordpress as the managed application. The PHP version of RUBiS [23], which is a web application created as a benchmarking etalon, was also tried, but it proved to be ill suited for a cloud scaling experiment, as the design of the system is 10 years old and is, contrary to Wordpress, not prepared for horizontal scaling.

A graph from the benchmarking scenario is on Figure 3. In blue is the number of simulated users, who are alternating between thinking (0.5 - 2 s) and waiting for server response. The peak load was about 100 requests per second. In red is the number of instances single CPU and 512 MB RAM on an Intel(R) Core(TM) i3-2100T CPU @ 2.50GHz (2 cores, 4 threads) machine). In green are the response times at the load tester. A drawback of the software load balancer can be seen on the failed connection count (black), which spikes for several hundred milliseconds every time the balancer configuration is reloaded. HAProxy was also tried but had the same problem. The x axis is in milliseconds, y in units of instances and percents of failed connections.

The ScaleGuru application has a modern looking web interface created using Twitter Bootstrap. The monitoring panel, shown on Figure 5, has the number of running instances in green, pending in orange and the red line is average CPU utilization across the autoscaling group. Machine access using a query interface is also possible, it is however currently not Amazon-compatible.

What is important in the context of this paper is that all historical performance data on all autoscaling groups are saved in the database, which enables later analysis using time series methods.

Therefore, the autoscaler will provide input for further experiments on the level of particular applications and will create non-static load in the context of the whole private cloud. A next version of the system could also use the output of the predictor as input for its autoscaling decisions and thus be able to provision capacity for a spike (of a predictable daily or weekly nature), before an actual overload happens.

As far as we know, it is the only piece of autoscaling software, which is installable on a private cloud and fairly universal, and, therefore, suitable for experiments. All other solutions we found were either offered as remotely as Software as a Service or were simple scripts created for a particular project.

C. *Cloud Gunther*

While the ScaleGuru project will also be instrumental for further research, the Cloud Gunther and possibilities for its further development are the main topic of this article.

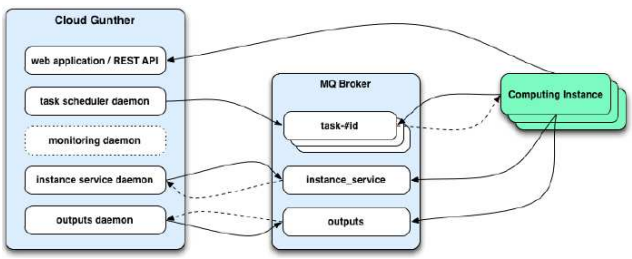


Figure 4. Communication scheme in Cloud Gunther [24]

The application is written in the Ruby on Rails framework and offers both interactive and REST (Representational State Transfer) access. It depends on Apache with mod_passenger, MySQL and RabbitMQ for operation. It can control multiple Amazon EC2 [20] compatible clouds. The queuing logic resides outside the MVC (Model, View, Controller) scheme of Rails, but shares database access with it. The communication scheme is on Figure 4.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

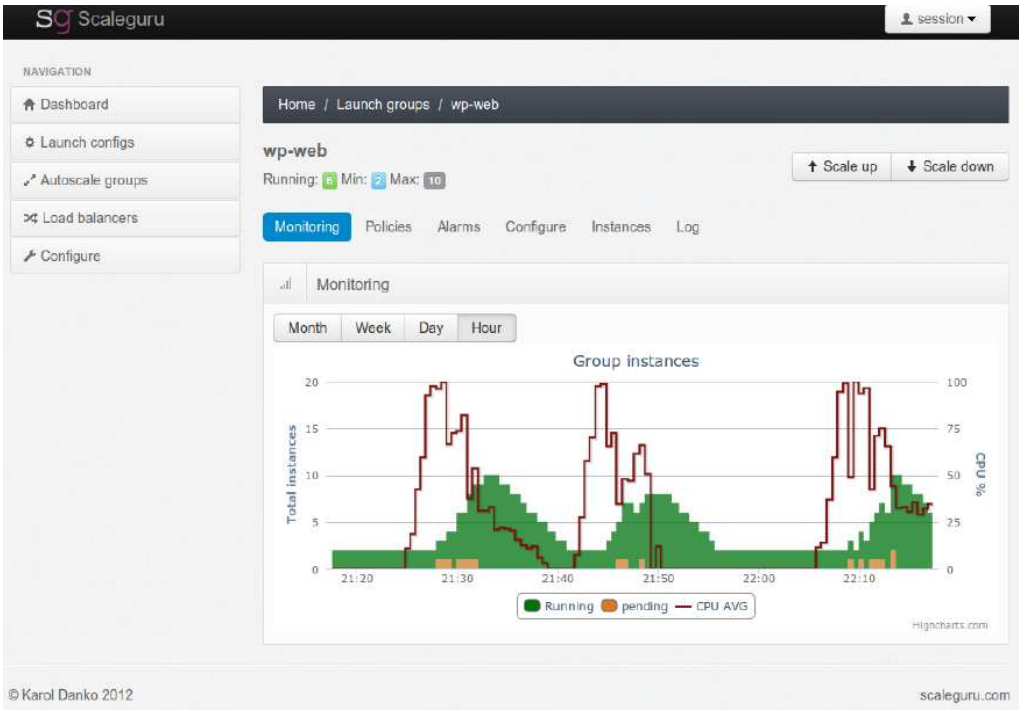


Figure 5. ScaleGuru web interface [21]

The Scheduler daemon contains the Priority FCFS queuing discipline and is responsible for launching instances and submitting their job details to the message broker. The Agent on the instance then retrieves these messages and launches the specified user algorithm with the right parameters. It is capable of running multiple jobs of the same type from the same user, thus saving the overhead of instance setup and teardown.

The two other daemons are responsible for collecting messages from the queue, which are sent by the instances. The Instance Service serves to terminate instances, which have run out of jobs to execute; the Outputs daemon collects standard and error outputs of user programs captured by the launching Agent. A Monitoring daemon is yet to be implemented.

The web application itself fulfills the requirement of multitenancy by providing standard user login capabilities. The users can also be categorized into groups, which have different priorities in the scheduler.

The cloud engine credentials are shared for each cloud (for simpler cloud access via API and instance management via SSH (Secure Shell)).

Each cloud engine has associated images for different tasks, e.g., image for Ruby algorithms, image for Java, etc. The images are available to all users, however when launched, each user will get his own instance.

The users can define their algorithm's requirements, i.e., which image the algorithm runs on and what instance size it needs. There is also support for management of different versions of the same algorithm. They may only differ in command line parameters, or each of them may have a binary program attached to it, which will be uploaded to the instance before execution.

Individual computing tasks are then defined on top of the algorithms. The task consists of input for the algorithm, which is interpolated into its command line with the use of macros, as well as the instance index and total count of instances requested. These values are used by pseudoparallel algorithms to identify the portion of input data to operate on, and by parallel algorithms for directing communication in message passing systems.

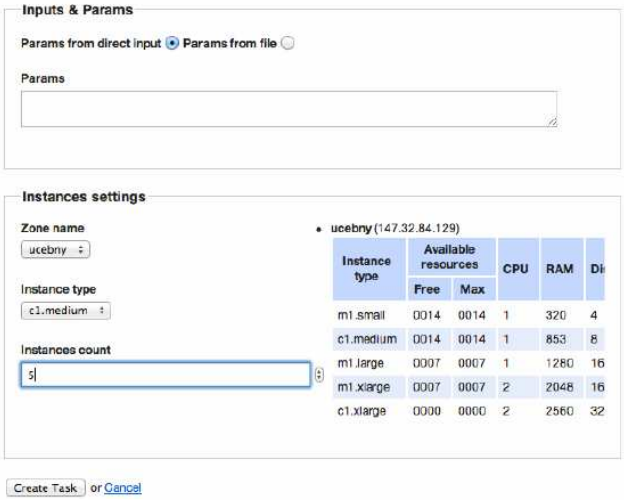


Figure 6. Cloud Gunther – part of the New Task screen [24]

As one can see in Figure 6, the system is ready for private clouds. It can extract the amount of free resources

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

from Eucalyptus and the scheduler takes it into account when launching new instances.

The Cloud Gunther has been tested on several real workloads from other scientists. Those were production planning optimization, recognition of patterns in images and a multiagent simulation. They represented a parameter sweep workflow, a pseudoparallel task and a parallel task, respectively.

VM images for running the tasks were prepared in cooperation with the users. Usability was verified by having the users set up algorithm descriptions in the web interface. The program then successfully provisioned the desired number of VM instances, executed the algorithms on them, collected the results and terminated the instances.

The main drawback, from our point of view, is that when there are jobs in the queue, the program consumes all resources on the cluster.

This is not a problem in the experimental setting, but in a production environment, which would be primarily used for interactive traffic and would attempt to exploit the agility of cloud infrastructure to run batch jobs as well, this would be unacceptable.

In such a setting, the interactive traffic needs to have absolute priority. For example, if there was a need to increase the number of web servers due to a spike in demand, then in the current state, the capacity would be blocked by Cloud Gunther until some of its tasks finished. It would be possible to terminate them, but that would cause loss of hours of work. A proactive solution to the heterogenous load situation is needed.

IV. TIME SERIES

The sought solution will deal with estimation of the amount of interactive load in time. The interactive traffic needs to have priority over the batch jobs. Therefore, the autoscaler will record the histogram of the number of instances that it is managing. From this histogram, data on daily, weekly and monthly usage patterns of the web servers may be extracted and used to set the amount of free resources for Cloud Gunther.

A similar problem exists in desktop grids. Ramachandran, in article [25], demonstrates the collection of availability data from a cluster of desktop machines and presents a simulation of predictive scheduling using this data. The abstraction of the cloud will shield away the availability of particular machines or their groups, the only measured quantity will be the amount of available VM slots of a certain size.

With a predictor, instead of seeing only the current amount of free resources in the cloud, the batch job scheduler could be able to ask: "May I allocate 10 large instances to a parallel job for the next 4 hours with 80% probability of it not being killed?"

A solution to this question exists in statistics, in a discipline called Time Series Analysis. A good tutorial is written by Keogh [26]. It has very wide coverage, mainly on filtering, similarity measures, Dynamic Time Warping and lower bounds on similarity. However, the solution was found elsewhere, although clustering on particular days and offering the next day after the best match as forecast is also a valid approach and was evaluated as better than the two others presented here in the bachelor thesis of Babka [27] on photovoltaic power plant output prediction.

A. Holt-Winters exponential smoothing

Due to the fact that the ScaleGuru autoscaler was not yet tested in a real environment, it was decided to obtain experimental data from single servers of a web hosting company. These are monitored by Collectd and time series data stored in RRDTool's Round Robin Databases. While examining the documentation for export possibilities, a function by Brutlag [28] was discovered, which uses Holt-Winters exponential smoothing to predict the time series one step ahead and then raise an alarm if the real value is too different from the prediction. This allows to automatically detect spikes in server of network activity.

A good description of exponential smoothing methods including mathematical notation is written by Kalekar [29]. Simple exponential smoothing is similar to moving average. It has a single parameter, α, which controls the weight of the current observation versus the historical value and a single memory that holds the average. It is good for time series that do not exhibit trend or seasonality, and its prediction is a straight line in the mean.

Double or Holt's exponential smoothing takes trend into account. It has 2 parameters, α and β, and a memory of 2, the mean and the slope. The slope is calculated as an exponentially smoothed difference between the current value and predicted mean. Predictions from this model are a straight line from the mean under the average slope.

Lastly, Triple or Holt-Winters exponential smoothing takes seasonality into account. It has 3 parameters, α, β and γ and a memory of 2 plus the number of observations per period. The seasonal memory array holds the factor or addend (depending on whether multiplicative or additive seasonality is used) of each observation point in the season to the exponentially smoothed value, and is itself updated through exponential smoothing. The prediction from this model looks like the average season repeated over in time, starting at the average value and "stair-stepping" with the trend.

Estimation of the parameters can either be done by hand and evaluated using MSE (Mean Squared Error) or MAPE (Mean Average Percentage Error) on the training data (a quick explanation of their significance is in Hyndman [30]), or it can be left to statistics software, which can do fitting by least square error. For the experiments in this paper, the R statistics package [31] was used, particularly the forecast package by Hyndman [32]. The RRDTool implementation is not suitable as it only forecasts one point into the future for spike detection.

An introduction to time series in R, including loading of data, creating time series objects, extracting subsets, performing lags and differences, fitting linear models, and using the zoo library is written by Lundholm [33]. A summary of all available time series functions is in the time series task view [34], while a more mathematical view of the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

capabilities including citations of the authors of particular packages is in McLeod, Yu and Mahdi [35].

B. Experiments

1) Loading of data

The evaluation of the method was done on six time series from servers running different kinds of load. The data was first extracted from RRDTool and pushed into MySQL by a bash script, which was being run every day to get data at the desired resolution. The RRD format automatically aggregates data points using maximum, minimum and average, after they overflow the configured age boundaries. Those were (in files created by Collectd) 10 hours in 30 second intervals, 24 h in 60 s, 8 days in 8 minutes, 1 month in 37 min, and 1 year in 7.3 hours.

The chosen initial resolution for experiments was 15 minutes, as the aim is to forecast a) for IaaS clouds, where instance start-up takes about 5 minutes, plus user initialization, and accounting is done in hours, and b) for batch jobs, where the user will probably give task durations in hours or their fractions. Later, it will be evident that this resolution is appropriate for forecasts with the horizon of days, which was the goal of the selection.

The data was then loaded into R (using manual [36]). There was a total of 8159 observations or 2.8 months of data. Time series objects (ts) were created. Their drawback is that observations need to be strictly periodic and the x axis is indexed only by numbers. Any missing values have been interpolated (there was no larger consecutive missing interval). For uneven observation intervals, the "zoo" library may be used, which indexes observations with time stamps [37]. It was not used here, so for clarification: The measurement interval starts with time stamp 1128, which was November 28, and then the count increases every day by 1 irrespective of the calendar as the seasonal frequency was set to 1 day. Therefore, the interval contains Christmas at about 1/3, and it ends on Thursday.

2) Time series diagnostics

The servers included in the experiments have code names oe, bender, lm, real, wn, gaff. In the next paragraph follow their designations and the result of examinations of the time plots of their CPU load time series. This series was also filtered by simple moving average (SMA) with window set to 1 day to obtain deseasonalized trend. The time plots of the series along with best forecasts from both methods are attached in Appendix A.

- oe is a large web shop. It has a clear and predictable daily curve with one weekday higher and weekend and holidays lower (incl. Christmas). Trend is stationary (except Christmas).
- bender is shared PHP webhosting. It has a visible daily curve with occasional spikes. First month shows a decreasing trend, and then it stabilizes.
- lm is a discount server. The low user traffic creates a noisy background load that is dominated by spikes of periodic updates. Trend alternates irregularly between two levels; the duration is on the scale of weeks.
- real is a map overlay service, not much used but CPU intensive (as one map display operation fetches many objects in separate requests). The time plot is a collection of spikes, more frequent during day than night. There are 2 stationary levels, where the first month the load was higher, and then the site was optimized so it went lower.
- wn is PHP hosting of web shops. It has low traffic with a visible daily curve. There is a slow linear additive trend after the first month.
- gaff is a web shop aggregator and search engine. Its daily curve is inverted with users creating background load in the day and a period of high activity due to batch imports during the night. Trend is stationary.

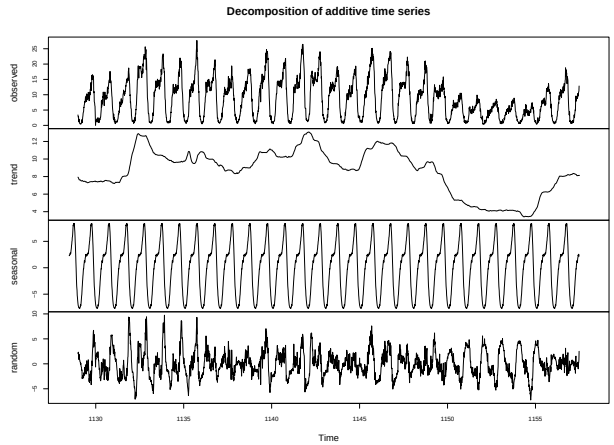


Figure 7. oe series decomposition, from top to bottom: overall time plot, trend, seasonal and random compoment

As suggested in the tutorial by Coghlan [38], which also covers installation of R and packages, as well as Holt-Winters and ARIMA models, the time series were run through seasonal decomposition. For oe, bender and wn, the daily curve was as expected; with gaff, the nightly spike also showed nicely. lm and real surprisingly also show daily seasonality as the spikes are apparently due to periodic jobs. Decomposition of the first month of oe is in Figure 7. We can clearly see the repeated daily curve and a change in trend during Christmas.

Another tool to diagnose time series is the seasonal subseries plot. When applied to the test data, only oe shows clean seasonal behavior. In the bender series, noise may be more dominant than seasonality. The lm series seasonal subseries is also not clearly visible. real clearly shows that traffic on certain hours is higher. For wn, the upward trend is visible in each hourly subseries. gaff shows that the duration of the batch jobs is not always the same so there are large spikes in the morning hours, mainly at the start of the measurement interval. This plot is in Figure 8. It contains 96 subseries because of the 15-min frequency, index 0 is midnight.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

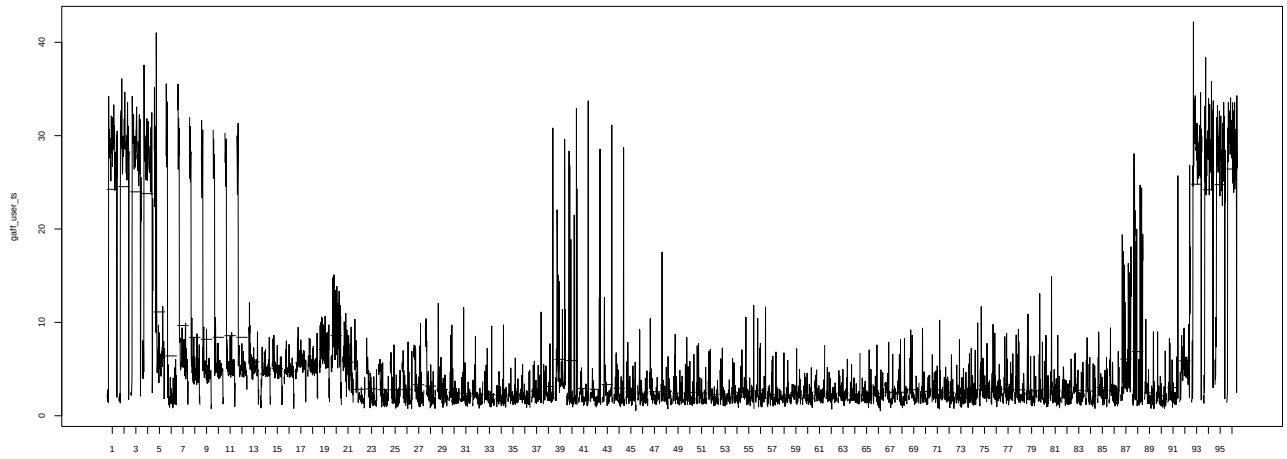


Figure 8. gaff series seasonal subseries plot

3) Model fitting and evaluation

A modified script from Hyndman and Athanasopoulos [39] was used for model fitting and validation. The algorithm first shortens the time series by 3 days at the end and fits a model on it. Then forecasts are created for 6, 24, and 96 hour horizons and compared with the withheld validation data. The result is a table of standard model efficiency measures for each series and interval ("in" meaning in-sample). One more measure was defined in accordance with the goal specified at the beginning of this section – how many validation data points missed the computed 80% prediction intervals in the 3-day forecast (that is 288 points in total).

As to the forecast error measures, the following ones are used: The Mean Error (ME) is a measure of error in absolute scale; it is signed, so it can be used to see a bias in forecasts, but cannot be used for comparison of time series with different scale.

The Root Mean Squared Error (RMSE) measures squared error and is thus more sensitive to outliers. It is best used when the scale of errors is significant. The square root operation returns the dimension to that of the original data.

Mean Absolute Error (MAE) is similar to ME, but ignores the direction of the error by using absolute values.

Mean Percentage Error (MPE) removes the influence of scale from ME by dividing error by the value,

Mean Absolute Percentage Error (MAPE) does the same to MPE. It is probably the best measure for human evaluation.

The Mean Absolute Scaled Error (MASE) is different from the others in that it does not compare the error to the original data, but to the error of the naïve "copy the previous value" forecast method.

For one-step-ahead forecasts, MASE values below one indicate that the evaluated method is better. For larger horizons, this is not true, as the naïve method has more information than the one under evaluation (i.e., always the previous data point). Normally, ME, RMSE, and MAE have the dimension of the original data, MPE and MAPE are in

TABLE I. EVALUATION OF THE HOLT-WINTERS MODEL ON OUT-OF-SAMPLE DATA

	ME	RMSE	MAE	MPE	MAPE	MASE	miss		ME	RMSE	MAE	MPE	MAPE	MASE	miss
oe in	0.003	1.109	0.798	2.776	17.91	1.036		**rea1 in**	-0.03	4.836	3.029	-18.2	38.47	0.398	
oe 6	0.691	1.110	0.829	6.111	7.623	1.076		**rea1 6**	-1.54	7.206	4.878	-68.3	86.06	0.641	
oe 24	0.500	2.461	1.985	-30.4	62.34	2.575		**rea1 24**	-0.28	6.843	4.770	-50.2	71.75	0.627	
oe 96	1.843	4.238	3.223	-26.1	75.49	4.181	2	**rea1 96**	-0.31	7.004	4.916	-56.3	77.77	0.646	84
bend in	-0.06	1.699	1.176	-7.38	23.45	1.110		**rea2 in**	-0.11	7.515	5.973	-68.4	95.76	0.785	
bend 6	0.015	1.280	1.068	-2.11	14.47	1.009		**rea2 6**	-1.78	6.866	5.485	-105	122.1	0.721	
bend 24	-0.36	1.436	1.200	-17.9	27.39	1.133		**rea2 24**	-0.28	8.304	6.619	-88.9	115.6	0.870	
bend 96	-1.33	2.385	1.934	-35.7	41.42	1.826	2	**rea2 96**	-0.30	8.387	6.713	-95.1	122.0	0.883	44
lm1 in	-0.35	5.408	3.832	-10.3	31.63	0.801		**wn in**	-0.01	2.469	1.600	-15.2	43.31	1.047	
lm1 6	3.408	4.839	3.713	18.09	20.46	0.777		**wn 6**	-0.35	1.880	1.553	-11.4	24.44	1.016	
lm1 24	-12.9	17.78	14.81	-119	129.4	3.099		**wn 24**	-1.42	3.617	2.980	-74.8	87.18	1.950	
lm1 96	-27.2	32.23	27.86	-248	251.4	5.830	97	**wn 96**	-1.29	5.151	3.995	-86.4	102.8	2.614	0
lm2 in	0.002	5.638	3.856	-13.5	31.52	0.806		**gaff in**	-0.01	3.562	2.039	-8.90	57.79	1.158	
lm2 6	0.639	5.667	4.625	-6.41	28.14	0.967		**gaff 6**	0.191	7.099	6.449	63.97	465.5	3.663	
lm2 24	-1.04	6.939	5.104	-24.7	40.55	1.068		**gaff 24**	-0.01	6.835	4.308	-8.97	189.3	2.447	
lm2 96	-1.04	7.666	5.624	-29.4	45.83	1.176	14	**gaff 96**	0.622	5.927	4.002	5.364	157.7	2.274	4

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

percent and MASE is dimensionless. Here, all values are dimensionless as the input data is a time series of CPU load percentages.

The result can be seen in Table I. For lm, two result sets are included. The first is from a triple exponential smoothing model, but as there was a spike at the end of the fitting data, the function predicted an upward trend while the data was in fact stationary. Simple exponential smoothing was then tried, which gave lower error measures and fewer points outside confidence intervals.

A similar problem existed with real. The spikes predicted by the seasonal model missed the actual traffic spikes most of the time. It seems that the series is not seasonal after all, but rather cyclic. The cause for the spikes is random arrivals of requests, as per queuing theory. Cyclicity is discussed in Hyndman [40]. The important outcome is that exponential smoothing models cannot capture it, while autoregressive models can.

The second model for real in the table is double exponential smoothing, which, interestingly, shows higher error measures, but lower number of missed observations. The cause is that the confidence intervals are computed based on the variance of in-sample errors. Therefore, the closer the error magnitude is between in-sample and out-of sample measurement, the more accurate the model is in the "misses" measure.

Automatic model fitting also failed for gaff. The transition from the nightly spike to daily traffic caused the predicted values to be below zero. A manual adjustment of Alpha parameter was necessary. Computed α=0.22, set α=0.69. The problem probably is that the algorithm optimizes in-sample squared error (MSE) and thus it preferred a slower reaction, which mostly missed the spike. The computed trend from this mean was therefore strongly negative. A quicker reaction to the change in mean improved the model, but even then, series with abrupt changes in mean are not good for the Holt-Winters model.

From Table I., we can see that with the Holt-Winters method, some series are predicted well even for the 3 day interval (bender, lm method 2), for some, the forecast is reasonably accurate for the first 6 hour interval and then deteriorates (oe, lm method 1, wn), for others it is inaccurate (real, gaff).

In addition, when the error measures for in-sample data are worse than for out-of-sample, it is a sign of overtraining - the validation data set was closer to "average" than the training data. This is because we were training on a long period including Christmas and verifying on a normal week. Perhaps shortening the training window would be appropriate.

C. Box-Jenkins / ARIMA models

The tutorial [38] suggests using autocorrelation plot on the residuals of the Holt-Winters model. A significant autocorrelation of the residuals means that they have a structure to them and do not follow the character of white noise. All the models showed significant autocorrelation of residuals at both low lags and lags near the period. The Ljung-Box test is a more rigorous proof of randomness of a time series as its null hypothesis is that a group of autocorrelations up to a certain lag is non-significant. It can thus ignore a random spike in the ACF. All the models failed the test in the first few lags.

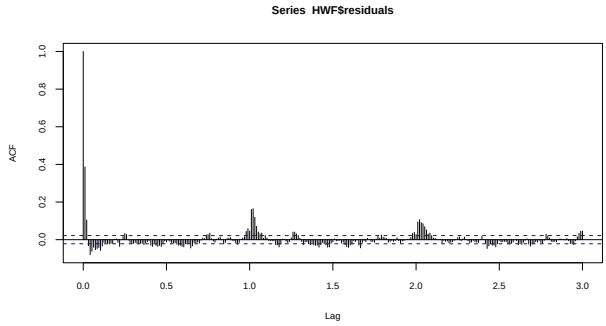


Figure 9. Autocorrelogram of residuals of the H-W model on bender

Having seen autocorrelation plots such as in Figure 9, it was decided to move to better models. ARIMA (Autoregressive, Integrated, Moving average) models are intrinsically based on autocorrelation. They seem to be the state of the art in time series modeling and are a standard in economic prediction (e.g., [39] is a textbook for business schools and MBA).

Neural network methods were also studied, but, as Crone's presentation, which is also a good source on time series decomposition and ARIMA [41], suggests, their forecasting power is equal to ARIMA, only the fitting method is different. It may be more powerful in that it is non-linear and adaptive, but has many degrees of freedom in settings and the result is not interpretable.

As per the NIST Engineering Statistics Handbook [42], chapter 6.4.4.4, which is a good practical source on all methods discussed here, the autoregressive and moving average models were known before, but Box and Jenkins have combined them together and created a methodology for their use.

There are three major steps in the methodology: model selection based on mainly on examination of autocorrelograms (ACF) and partial autocorrelograms (PACF), then model estimation, which uses non-linear least square fitting and/or maximum likelihood and is best left to statistical software, and lastly model validation, which uses ACF and PACF of residuals and the Ljung-Box test.

An autoregressive (AR) model computes the next data point as a linear combination of previous ones, where the number of lagged values considered is determined by the order of the model. The parameters are the mean and the coefficients of each lag. They can be computed by linear least squares fitting. A model of order greater than one with some coefficients negative can exhibit cyclic behavior.

A moving average (MA) model works with errors. The next data point is a linear combination of differences of past lags from the moving average, where the number of lags considered is the order of the model. Again, each term has a parameter that needs to be estimated. The estimation is more difficult as the errors cannot be known before the model

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

exists, which calls for an iterative non-linear fitting procedure.

The I in ARIMA stands for integrated, which represents the inverse operation to differencing. As the AR and MA models assume that the time series is stationary, meaning that it has stable location and variance, the difference operator can often be used to transform a series to stationary. The model is fitted to the transformed series and an inverse transform is used on the resulting forecast.

Other useful transformations are logarithms and power transforms, which may help if the variance depends on the level. They are both covered by the Box-Cox transform (see [39], chapter 2/4).

D. Experiments

1) Model selection

a) Differencing order

The prerequisite for ARIMA is that the time series is stationary. Manually, stationarity can be detected from the time plot. A stationary time series has constant level and variance, and may not exhibit trend or seasonality. The two last effects should be removed for identification of model order, but are covered by ARIMA models with non-zero differencing order and SARIMA (Seasonal ARIMA), respectively. For series with non-linear trend or multiplicative seasonality, the Box-Cox transform should be used, but that was not the case with the series studied here. Additionally, a non-stationary series will have ACF or PACF plots that do not decay to zero.

The statistical approach to identification of differencing order is through unit root tests (see Nielsen [43]). The root referred to here is the root of the polynomial function of the autoregressive model. If it is near one, any shocks to the function will permanently change the level and thus the resulting series will not be stationary. The standard test for this is Augmented Dickey-Fuller (ADF), which has the null hypothesis of unit root. A reversed test is Kwiatkowski-Phillips-Schmidt-Shin (KPSS), where the null hypothesis is stationarity. There is also a class of seasonal unit root tests that can help specify the differencing order for SARIMA, these are Canova-Hansen (CH) and Osborn-Chui-Smith-Birchenhall (OCSB).

In R, there exist functions ndiffs() and nsdiffs(), which automatically search for the differencing and seasonal differencing order, respectively, by repeatedly using these tests and applying differences until the tests pass (for KPSS and CH), or stop failing (for ADF and OCSB). The default confidence level is 5%. The recommended amount of differencing of the experimental time series obtained from the tests is in Table II on the next page. Columns lm4 and real4 will be explained later.

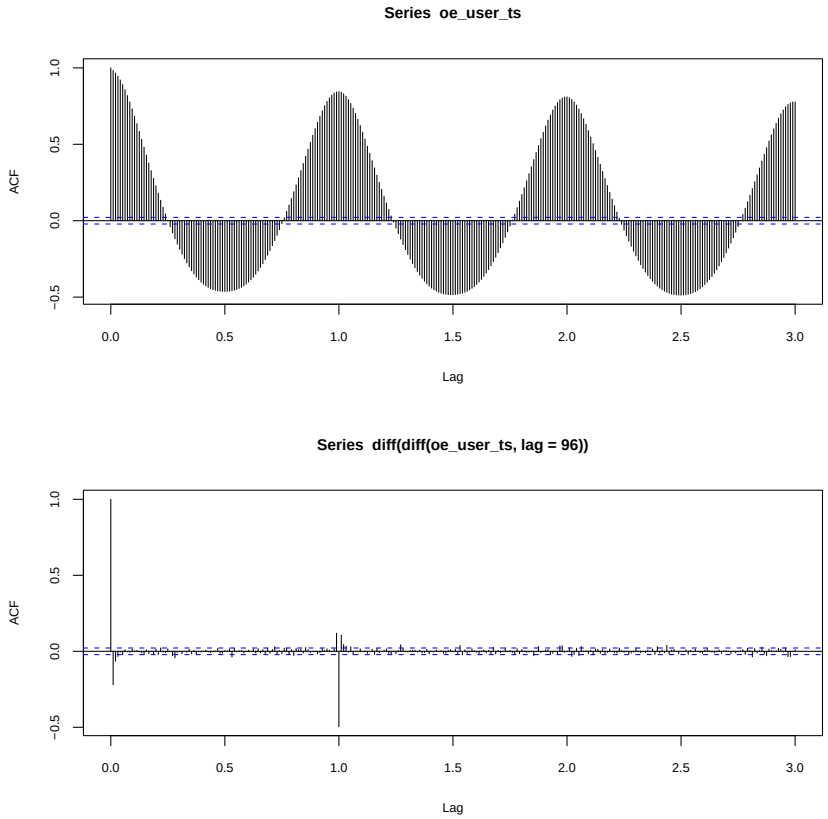


Figure 10. ACF of oe without and with differencing

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE II. ORDER OF DIFFERENCING BASED ON UNIT ROOT TESTS

	oe	bender	lm	lm4	real	real4	wn	gaff
ADF	0	0	0	0	0	0	0	0
KPSS	0	1	1	1	1	1	1	0
OCSB	0	0	0	0	0	0	0	0
CH	0	0	0	1	0	1	0	0

It is evident that the ADF and KPSS tests did not agree with each other with the exception of oe and gaff. According to [43], ADF should be considered primary and KPSS confirmatory. The same is said by Stigler in discussion [44], adding that unit root tests have lower sensitivity than KPSS. In the same discussion, Frain says KPSS may be more relevant as a test concretely for stationarity (there may be non-stationary series without a unit root), if we do not assume a unit root based on underlying theory of the time series. It was also used by Hyndman in the auto.arima() function for iterative model identification.

According to manual heuristic approaches, such as presented by Nau [45], an order of seasonal differencing should always be used if there is a visible seasonal pattern. It also suggests applying a first difference if the ACF does not decay to zero. An example of the impact of first and seasonal differencing on stationarity and thus legibility of an ACF plot is in Figure 10.

The ACF and PACF functions on the test data were looked at with and without differencing with the result that differencing rapidly increases the decay of the ACF function on all series except real.

Moreover, from the ACF of lm and real, it seems there is a strong periodicity of 4 hours. These two series will be also tested with models of this seasonal frequency and will be denoted as lm4 and real4, as in Table II.

For the purpose of order identification, seasonal and then first differences have been taken. It was decided to test if the models fitted with this order of differencing, following the heuristic approach, are better or worse than those with differencing order identified by statistical tests.

b) Order identification

Identification of model order was done using heuristic techniques from [39], [42], [45], and [46]. After seasonal and first differencing is applied in the necessary amount to make the time series look stationary to the naked eye, so that its autocorrelograms converge to zero, the ACF and PACF functions are looked at. The number of the last lag from the beginning where PACF is significant specifies the maximum reasonable order of the AR term, similarly the last significant lag on ACF specifies the MA order. The order of the seasonal autoregressive and moving average terms is obtained likewise, but looking at lags that are multiplies of the seasonal period.

The observed last significant lags and resulting maximum model orders are summed in Table III. Model parameters are denoted as ARIMA(p, d, q)(P, D, Q), where p is the order of the AR term, d is the amount of differencing and q is the order of the MA term. The second parenthesis specifies the seasonal model orders.

TABLE III. LAST SIGNIFICANT LAGS AND MODEL ORDERS

	PACF	ACF	seas. PACF	seas. ACF	estimated maximal model parameters
oe	5	3	11	1	ARIMA(5,1,3)(11,1,1)
bender	17	4	9	1	ARIMA(17,1,4)(9,1,1)
lm	15	16	8	1	ARIMA(15,1,16)(8,1,1)
lm4	9	2	11	∞	ARIMA(9,1,2)(11,1,0)
real	1	2	11	1	ARIMA(1,0,2)(11,1,1)
real4	13	2	11	1	ARIMA(13,1,2)(11,1,1)
wn	39	3	10	1	ARIMA(39,1,3)(10,1,1)
gaff	18	2	6	1	ARIMA(18,1,2)(6,1,1)

Looking at the two variants of lm, the expectation is that the first will perform better, as the non-seasonal part covers the second period of 4 hours. This is not true for real vs. real4.

2) Model estimation

When trying to fit models with high seasonal order, a limitation of the ARIMA implementation in R was found. The maximal supported lag is 350, which with a period of 96 (24 hours * 4 observation per hour) means that the seasonal lag is limited to 3.

Furthermore, the memory requirements of seasonal ARIMA seem to be exponential with the number of data points. A machine with 1 GB of RAM could not handle the 2.8 months of data with lag 288. This constraint is not documented. The experiment had to move to a machine with 32 GB RAM, where computing a model with seasonal order 3 took 7.6 GB RAM, more on subsequent runs as R is a garbage collected language.

For the course of this experiment, the order of the seasonal components will be limited to three, as it should be sufficient when forecasting for a horizon of about a day. The alternatives, which will be examined in further experiments, are to reduce the resolution to 1 hour, which will enable lags up to 12 days.

A model of this sort was fitted on oe, and it did not lead to a better expression of the weekly curve (at least not by visual inspection). With this resolution, it will be however possible to use a seasonal period of one week, which should be able to capture the day-to-day fluctuations. Similarly, we would reduce resolution is we were trying to capture monthly or yearly seasonality.

Another approach, suggested by Hyndman [47], is to model the seasonality using a Fourier series and to use non-seasonal ARIMA on the residuals of that model. This should enable fitting on arbitrarily long seasonal data. This may lead to overfitting, though, as the character of the time series is subject to change over longer time periods.

For the actual parameter estimation, the Arima() function with the model order as parameter can be used. There is however a way to automate a part of the identification-estimation-validation cycle and that is the auto.arima() function. This function repeatedly fits models with different parameters and then returns the one that has minimal Akaike Information Criterion (AIC). This criterion prefers models with lower likelihood function and contains a penalization for the number of degrees of freedom of the model; therefore, it should select the model that best fits the data,

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

but not variations of the same model with superfluous parameters.

The auto.arima() function has two modes depending on the "stepwise" parameter (see help(auto.arima) in R). With this set to TRUE, it does a greedy local search, which selects the best model from previous step and examines its neighborhood in the state space given by adding or subtracting one to each parameter. It continues, until no model in the neighborhood has lower AIC.

The second mode searches from ARIMA(0,0,0)(0,0,0) upwards and based on the description, it should search until the ceiling set for each parameter. The actual behavior however seems to be that is stops when the last iteration examined did not bring any gain. Both search modes are thus prone to getting stuck in a local minimum.

To better specify the models, the auto.arima() function was used on each time series with three sets of parameters. In the first run, it was started from zero with stepwise=FALSE and with ceilings set to the parameters estimated in Table III. In the second run, stepwise was set to TRUE and the ceilings were left at the pre-estimated parameters plus one to account for differencing; the starting values were set to be the same as the ceilings, as, theoretically, the parameters in Table III should be the maximal meaningful numbers, but a model with lower orders might be better. This was tested in the third run, where the starting values remained and the ceilings were effectively removed.

The same procedure was then repeated with the differencing orders computed by OCSB and KPSS. As it is difficult to identify model parameters by naked eye without differencing, the same initial parameters have been used. Please note that the AIC values of models with unequal differencing order are not comparable, while goodness-of-fit test results and prediction errors are.

3) Model validation

As already discussed in Subsection IV/C "Box-Jenkins models", the validation entails manual examination of the autocorrelation plot of residuals and use of the Ljung-Box goodness-of-fit (GOF) test. Table IV contains the models that resulted from the three runs of auto.arima() as described, along with their AIC values, the lag of the first significant autocorrelation and the lag after which the Ljung-Box test failed. Left side is for models with differencing order set to one, right side has differencing set by unit root tests.

The outcome from Table IV is, that is cannot be conclusively said whether it is better to always use seasonal differencing or not. Of the six time series, three have the best fitting model in the left half of the table and three in the right half. However, it seems that in the cases where the non-differenced models were better, the gain in the goodness-of-fit functions was lower than the other way round. It is also interesting that in two of the three cases (oe and gaff), the difference is not only in seasonal, but also in first differencing. It may be a good idea to follow the recommendation of the KPSS test, but always use seasonal differencing, but there is not enough data to say it with certainty.

A more solid fact is that all the best models come from the third row of the table. Of the three tried here, the best algorithm for model selection is to use auto.arima() in greedy mode, starting with parameters identified from ACF and PACF, and leave it room to adjust the parameters upwards.

E. Comparison of the two model families

The last part of the experiment entailed computing forecasts based on the fitted ARIMA models and comparing them with out-of-sample data. The same validation algorithm

TABLE IV. PARAMETERS OF THE ESTIMATED ARIMA MODELS AND THEIR VALIDATION MEASURES

	model	AIC	sig. ACF	fail. GOF		model	AIC	sig. ACF	fail. GOF
oe	ARIMA(0,1,2)(1,1,2)	23016.97	12	14	**oe**	ARIMA(4,0,3)(2,0,2)	23231.26	22	23
	ARIMA(6,1,1)(1,1,2)	23109.12	12	15		ARIMA(5,0,4)(3,0,2)	23259.8	21	27
	ARIMA(5,1,3)(2,1,3)	23066.43	16	20		*ARIMA(5,0,3)(3,0,3)*	*23220.86*	*21*	*27*
bend	ARIMA(1,1,1)(2,1,1)	28082.19	4	6	**bend**	ARIMA(1,1,1)(3,0,2)	27989.07	22	6
	ARIMA(17,1,5)(3,1,2)	27580.45	52	129		ARIMA(17,1,4)(3,0,2)	27812.25	26	60
	ARIMA(17,1,3)(3,1,3)	*27504.15*	*58*	*172*		ARIMA(14,1,1)(3,0,3)	27801.06	17	58
lm	ARIMA(1,1,1)(2,1,1)	47569.93	5	5	**lm**	ARIMA(1,1,3)(1,0,2)	48517.21	5	4
	ARIMA(16,1,17)(2,1,2)	47210.57	94	144		ARIMA(15,1,17)(3,0,1)	48155.89	43	144
	ARIMA(17,1,17)(2,1,3)	*47195.88*	*98*	*500+*		ARIMA(15,1,18)(3,0,1)	48152.6	70	144
lm4	ARIMA(2,1,1)(1,1,1)	49151.84	5	5	**lm4**	ARIMA(1,1,3)(0,0,2)	50789.93	4	4
	ARIMA(10,1,3)(12,1,1)	48398.45	11	14		ARIMA(10,1,2)(12,0,2)	48570.04	10	28
	ARIMA(11,1,2)(16,1,5)	48138.57	21	30		ARIMA(12,1,2)(15,0,4)	48342.89	21	30
real	ARIMA(1,1,1)(2,1,1)	47872.32	4	3	**real**	ARIMA(0,1,2)(3,0,0)	48873.62	4	4
	ARIMA(2,1,3)(2,1,2)	47800.56	1	1		ARIMA(2,1,3)(3,0,2)	48732.74	1	1
	ARIMA(6,1,8)(3,1,3)	46972.94	6	6		ARIMA(10,1,12)(3,0,3)	47344.03	8	9
rea4	ARIMA(2,1,1)(1,1,1)	47574.67	3	3	**rea4**	ARIMA(1,1,1)(3,0,0)	48897.42	3	3
	ARIMA(1,1,3)(1,1,2)	47612.58	2	2		*ARIMA(12,1,2)(11,0,1)*	*47438.97*	*21*	*24*
	ARIMA(4,1,5)(1,1,7)	47373.03	8	7		*ARIMA(12,1,2)(11,0,1)*	*47438.97*	*21*	*24*
wn	ARIMA(4,1,2)(2,1,2)	35599.79	9	14	**wn**	ARIMA(2,1,3) with drift	36214.64	10	9
	ARIMA(40,1,2)(2,1,2)	35608.54	55	500+		ARIMA(39,1,4)(1,0,2)	36177.56	59	95
	ARIMA(39,1,1)(2,1,3)	*35596.67*	*55*	*500+*		ARIMA(38,1,5)(1,0,3)	36146.1	64	191
gaff	ARIMA(2,1,3)(0,1,2)	21501.92	5	5	**gaff**	ARIMA(3,0,0)(1,0,1)	21847.8	5	5
	ARIMA(19,1,3)(0,1,2)	21387.26	42	52		*ARIMA(18,0,3)(1,0,2)*	*21717.96*	*43*	*88*
	ARIMA(17,1,4)(0,1,3)	21118.64	42	88		*ARIMA(18,0,3)(1,0,2)*	*21717.96*	*43*	*88*

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE V. EVALUATION OF THE ARIMA MODELS ON OUT-OF-SAMPLE DATA

	MAPE in	MAPE 6	MAPE 24	MAPE 96	miss		MAPE in	MAPE 6	MAPE 24	MAPE 96	miss
oe	13.43	*7.12*	75.27	94.99	8	**oe**	13.40	8.13	37.52	53.16	22
	13.39	7.13	77.74	98.23	*7*		13.22	8.71	33.96	48.88	22
					failed		*13.16*	8.85	*31.41*	*46.35*	24
bend	19.32	*14.56*	22.18	22.21	25	**bend**	*18.28*	15.34	45.21	41.32	*13*
	18.51	15.51	*20.58*	*21.3*	42		18.79	21.42	36.53	34.38	87
					failed						failed
lm	24.93	19.14	*19.70*	*22.34*	17	**lm**	25.49	17.23	19.80	23.22	19
	23.94	14.79	20.10	25.15	20		23.98	15.74	21.01	26.91	21
					failed		23.97	15.98	21.11	27.02	21
lm4	25.50	13.22	23.36	28.93	8	**lm4**	28.77	26.94	44.30	51.17	*7*
	24.73	*11.66*	22.19	30.21	24		24.61	12.93	21.81	29.02	21
	24.16	15.57	20.29	23.45	19		24.47	12.51	19.88	25.73	21
real	36.26	85.66	73.58	80.20	85	**real**	38.18	72.49	62.91	64.49	59
	37.13	88.91	76.86	83.95	94		38.18	87.33	74.95	81.07	81
					failed		37.56	69.18	52.74	58.08	*39*
rea4	37.67	58.08	48.30	54.23	59	**rea4**	40.83	53.41	47.86	70.73	43
	37.99	53.44	43.22	*45.75*	40		*36.10*	*50.40*	*41.59*	46.06	49
	37.03	55.33	43.54	46.34	61		*36.10*	*50.40*	*41.59*	46.06	49
wn					failed	**wn**	42.03	*24.30*	78.80	82.33	102
	37.68	36.26	*51.12*	*50.25*	*59*		38.92	24.36	64.94	71.68	79
					failed		38.96	26.33	64.16	70.09	78
gaff	*37.62*	*160.67*	128.23	112.77	61	**gaff**	37.65	170.08	136.91	124.57	*59*
	38.19	165.29	*125.89*	*109.42*	60		38.26	165.48	133.85	119.44	*59*
	38.56	187.57	129.88	110.55	61		38.26	165.48	133.85	119.44	*59*

was used as in the case of Holt-Winters models, to facilitate model comparison. The result is in Table V. To conserve space, only MAPE (Mean Average Percentage Error) is shown. The four columns are for in-sample error and forecast errors in horizons 6, 24, and 96 hours. The ordering of models is the same as in Table IV.

Fitting of the forecasts was something of a disappointment, as all of the models with seasonal differencing (the left half of Table IV) that were selected as best using the GOF measures have failed to produce forecasts. The cause was likely the seasonal MA part of the model that was one or two orders higher that the originally identified ceiling. That resulted in an overspecified model where the MA polynomial was not invertible. Invertibility is a prerequisite for the computation of variances of the parameters [48], which in turn are needed to compute confidence intervals for a prediction. Hence, these models were fitted and had a likelihood function and in-sample errors, but could not be used for forecasts with confidence bounds.

When fitting ARIMA models in R, one needs to carefully observe the output for warnings such as:

```
In sqrt(z[[2]] * object$sigma2) : NaNs produced
```

for least-squares fitting, or for maximum likelihood:

```
Error in optim(init[mask], armafn, method =
optim.method, hessian = TRUE, :
non-finite finite-difference value [1]
In log(s2) : NaNs produced
```

because then the prediction will produce wrong results or fail:

```
In predict.Arima(object, n.ahead = h) :
MA part of model is not invertible
```

Therefore, if using auto.arima() beyond the ceiling identified from ACF and PACF, there is a high risk of the model failing and thus it may not be a good idea for automatic forecasts. If that happens, lowering the order or the seasonal MA or MA part should help.

As to the selection of the best model for forecasts, the selection based on out-of sample forecast errors (mainly looking at the 24 and 96-hour horizons) corresponds to the one based on goodness-of-fit criteria. In the case where the model fails to produce forecasts, the next-best one based on GOF can be selected. The second row (ceilings from ACF and PACF adjusted downward by auto.arima()) produced the best result, except on oe and lm, where, however, the difference is seems to be small.

As whether to always use seasonal differencing, the experiment is inconclusive. In the case of oe, there was a significant gain in accuracy by not using it, in the case of wn and bender, the opposite is true.

Looking at the "misses" criterion, one could say that Holt-Winters is better. However, that outcome might be skewed. The criterion counts the number of data points that missed the 80% confidence bounds in the 3-day forecast. That time period contains a total of 288 points, 20% of that is 57.6, and that is the count of data points that are by definition allowed to miss the bounds.

Therefore, the result of this comparison is that the confidence bounds on ARIMA are more accurate, or at least tighter than on Holt-Winters. If this method is to be used as proposed by this article, the confidence level used has to be adjusted upwards to 95 or 99%, depending on the overload sensitivity of the computer infrastructure.

Comparing the two model families using the MAPE error measure, the outcome is that ARIMA did produce better forecasts than Holt-Winters, except for the 6-hour forecasts on oe and bender, and also that simple exponential smoothing outperformed both seasonal methods on 24 and 96-hour forecasts on lm.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

V. FUTURE WORK

Future work planned on the Cloud Gunther can be split into two categories. First and more important is the consideration of interactive load also present on the cluster, which will require a rewrite of the queue engine to utilize the output of the predictor. Second is integration of better queuing disciplines to bring it up to par with existing cluster management tools. Two ideas for that are presented in Subsections A and B. Section C discusses the problem of resource sharing on modern computers.

A. Out-of-order scheduling

Using load predictions to maximize load of course assumes a scheduler that will be capable of using this information. Our vision is a queue discipline that internally constructs a workflow out of disparate tasks. The tasks, each with an associated estimate of duration, will be reordered so that the utilization of the cloud is maximized.

For example, when there is a job currently running on 20 out of 40 slots and should finish in 2 hours, and there is a 40 slot job in the queue, it should try to run several smaller 2 hour jobs to fill the free space, but not longer, since that would delay the large job.

These requirements almost exactly match the definition of the Multiprocessor scheduling problem (see [49]). Since this is a NP-hard class problem, solving it for the whole queue would be costly. The most feasible solution seems to come from the world of out-of-order microprocessor architectures, which re-order instructions to fully utilize all execution units, but only do so with the first several instructions of the program. The batch job scheduler will be likewise able to calculate the exact solution with the first several jobs in the queue, which will otherwise remain Priority FCFS.

B. Dynamic priorities

The estimation of job duration is a problem all for itself. At first, the estimate could be done by the user. Later, a system of dynamic priorities could be built on top of that.

The priorities would act at the level of users, penalizing them for wrong estimates, or better, suspending allocation of resources to users whose tasks have been running for longer time than the scheduler thought.

Inspiration for this idea is taken from the description of the Multilevel Feedback Queue scheduler used historically in Linux [50]. However, the scheduler will set priorities for users, not processes, and allocate VMs to tasks, not jiffies to threads. It also will not have to be real-time and preemptive, making the design simpler.

The scheduler's estimate of process run time could be based on the user estimates, but also on the previous run time of processes from the same task or generally those submitted by the same user for the same environment. That would lead to another machine learning problem.

C. Resource sharing

When we actually have both kinds of traffic competing for resources of the cloud, resource-sharing problems may affect the performance of the system and raise the observed requirements of the interactive traffic.

The effects of different kinds of algorithms on their surroundings will have to be benchmarked and evaluated. We may have to include disk and network bandwidth requirements in the model of the batch job and decide, which jobs may and which may not be run in a shared infrastructure, or ensure their separation through bandwidth limiting.

Second, even if we set up the private cloud so that CPU cores and operating memory are not shared, we still have to count with the problem of shared cache memory. This has been researched by several groups. Gusev and Ristov [51] benchmarked this by running multiple instances of a linear equation solver in a virtualized environment, and Babka, et.al., [52] measured the problem when concurrently running several benchmark kernels from SPEC2000.

VI. CONCLUSION

The cloud presents a platform that can join two worlds that were previously separate – web servers and HPC grids. The public cloud, which offers the illusion of infinite supply of computing resources, will accommodate all the average user's needs, however, new resource allocation problems arise in the resource-constrained space of private clouds.

We have experience using private cloud computing clusters both for running web services and batch scientific computations. The challenge now is to join these two into a unified platform.

The ScaleGuru autoscaling system offers an opportunity to get hand-on experience with automatic scaling, as most other systems are either very simple or are being developed in the commercial sector without public access to source codes.

The Cloud Gunther, although not ready for commercial deployment, already has some state of the art features, like the automatic management of cloud computing instances and a REST-compliant web interface. It also differs from other similar tools by its orientation towards private cloud computing clusters.

In the future, it could become a unique system for managing batch computations in a cloud environment primarily used for web serving, thus allowing to exploit the dynamic nature of private cloud infrastructure and to raise its overall utilization.

This article also presented two methods of time series forecasting, used otherwise mainly in economic forecasts, and which could be applied to server load data. These methods were tested on six time series of CPU load, some of which are web servers with a well defined daily curve (oe, bender, wn), and some have a load of more unpredictable nature (lm, real, gaff).

As it is expected that the cloud will contain mostly load-balanced web servers as the variable component, we think that these methods are viable for further research in the optimization of cloud computing.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

ACKNOWLEDGMENTS

Credit for the implementation of Cloud Gunther, mainly the user friendly and cleanly written web application goes to Josef Šín. The ScaleGuru application and all its modules were written by Karol Danko.

We thank the company Centrum for providing hardware for our experiments and insights on private clouds from the business perspective.

This work was supported by the Grant Agency of the Czech Technical University in Prague, grant no. SGS13/141/OHK3/2T/13, Application of artificial intelligence methods to cloud computing problems.

REFERENCES

[1] T. Vondra and J. Šedivý, "Maximizing Utilization in Private IaaS Clouds with Heterogenous Load," in CLOUD COMPUTING 2012: The Third International Conference on Cloud Computing, GRIDs, and Virtualization, IARIA, 22 July 2012, pp. 169-173.

[2] D. M. Smith, "Hype Cycle for Cloud Computing," Gartner, 27 July 2011, G00214915.

[3] T. Vondra and J. Šedivý, "Od hostingu ke cloudu," Research Report GL 229/11, CTU, Faculty of Electrical Engineering, Gerstner Laboratory, Prague, 2011, ISSN 1213-3000.

[4] R. Grossman and Y. Gu, "Data mining using high performance data clouds: experimental studies using sector and sphere," in Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08). ACM, New York, NY, USA, 2008, pp. 920-927, doi: 10.1145/1401890.1402000.

[5] M. J. Litzkow, M. Livny, and M. W. Mutka, "Condor-a hunter of idle workstations," in 8th International Conference on Distributed Computing Systems, 1988, pp. 104-111.

[6] W. Gentzsch, "Sun Grid Engine: towards creating a compute power grid," in Proceedings of the first IEEE/ACM International Symposium on Cluster Computing and the Grid, 2001, pp. 35-36.

[7] "Amazon Elastic Compute Cloud (EC2) Documentation," Amazon, <http://aws.amazon.com/documentation/ec2/> 27 May 2012.

[8] T. P. Morgan, "Univa skyhooks grids to clouds: Cloud control freak meets Grid Engine," The Register, 3rd June 2011, <http://www.theregister.co.uk/2011/06/03/univa_grid_engine_cloud/ > 19 March 2012.

[9] "Installing Eucalyptus 2.0," Eucalyptus, <http://open.eucalyptus.com/wiki/EucalyptusInstallation_v2.0> 19 March 2012.

[10] "StarCluster," Massachusetts Institute of Technology, < http://web.mit.edu/star/cluster/index.html> 11 May 2012.

[11] H. Eriksson, et al., "A Cloud-Based Simulation Architecture for Pandemic Influenza Simulation," in AMIA Annu Symp Proc., 2011, pp. 364–373.

[12] D. de Oliveira, E. Ogasawara, K. Ocaña, F. Baião, and M. Mattoso, "An adaptive parallel execution strategy for cloud-based scientific workflows," Concurrency Computat.: Pract. Exper. (2011), doi: 10.1002/cpe.1880.

[13] "Cloud Scheduler," University of Victoria, <http://cloudscheduler.org/> 11 May 2012.

[14] D. Warneke and O. Kao, "Nephele: efficient parallel data processing in the cloud," in MTAGS '09: Proceedings of the 2nd Workshop on Many-Task Computing on Grids and Supercomputers, November 2009, doi: 10.1145/1646468.1646476.

[15] R. N. Calheiros, C. Vecchiola, D. Karunamoorthya, and R. Buyya, "The Aneka platform and QoS-driven resource provisioning for elastic applications on hybrid Clouds," Future Generation Computer Systems 28 (2012), pp. 861-870, doi: 10.1016/j.future.2011.07.005.

[16] L.Yang, I. Foster, and J.M. Schopf, „Homeostatic and Tendency-based CPU Load Predictions," in Proceedings of IPDPS 2003, April 2002, p.9.

[17] M. Iverson, F. Özgüner, and L.C. Potter, "Statistical prediction of task execution times through analytic benchmarking for scheduling in a heterogeneous environment," in Proceedings of Eighth Heterogeneous Computing Workshop, HCW'99, IEEE, 1999, pp. 99-111.

[18] H. Li, „Performance evaluation in grid computing: A modeling and prediction perspective," in CCGRID, Seventh IEEE International Symposium on Cluster Computing and the Grid, 2007, IEEE, pp. 869-874.

[19] J. Šedivý, "3C: Cloud Computing Center," CTU, Faculty of Electrical Engineering, dept. of Cybernetics, Prague, <https://sites.google.com/a/3c.felk.cvut.cz/cloud-computing-center-preview/> 19 March 2012.

[20] T. Vondra, P. Michalička, and J. Šedivý, „UpCF: Automatic deployment of PHP applications to Cloud Foundry PaaS," 2012, unpublished.

[21] K. Danko, "Automatic Scaling in Private IaaS," Master's Thesis, CTU, Faculty of Electrical Engineering, Supervisor T. Vondra, Prague, 3 January 2013.

[22] "Amazon Auto Scaling Documentation," Amazon, <http://aws.amazon.com/documentation/autoscaling/> 12 March 2013

[23] ObjectWeb Consortium, "RUBiS: Rice University Bidding System," 2003, <http://rubis.ow2.org/> 12 March 2013.

[24] J. Šín, "Production Control Optimization in SaaS," Master's Thesis, CTU, Faculty of Electrical Engineering and University in Stavanger, Department of Electrical and Computer Engineering, Supervisors J. Šedivý and C. Rong, Prague, 20 December 2011.

[25] K. Ramachandran, H. Lutfiyya, and M. Perry, "Decentralized approach to resource availability prediction using group availability in a P2P desktop grid," Future Generation Computer Systems 28 (2012), pp. 854–860, doi: 10.1109/CCGRID.2010.54.

[26] E. Keogh, "A Decade of Progress in Indexing and Mining Large Time Series Databases," in Proceedings of the 32nd international conference on Very large data bases. VLDB Endowment, 14 September 2006.

[27] M. Babka, "Photovoltaic power plant output prediciton," Bachelor's Thesis, CTU, Faculty of Electrical Engineering, Supervisor P. Kordík, Prague, 1 May 2011.

[28] J. Brutlag, "Aberrant behavior detection in time series for network monitoring," in Proceedings of the 14th USENIX conference on System administration, 2000, pp. 139-146.

[29] P.S. Kalekar, "Time series forecasting using Holt-Winters exponential smoothing," Kanwal Rekhi School of Information Technology, 6 December 2004.

[30] R.J. Hyndman, "Hyndsight - Forecast estimation, evaluation and transformation," 10 November 2010 <http://robjhyndman.com/hyndsight/forecastmse/> 12 March 2013.

[31] R Development Core Team "R: A language and environment for statistical computing," R Foundation for Statistical Computing,Vienna, Austria, ISBN 3-900051-07-0 (2010), <http://www.R-project.org> 12 March 2013.

[32] R.J. Hyndman, "forecast: Forecasting functions for time series," R package version 4.0, 2011, <http://CRAN.R-project.org/package=forecast> 12 March 2013.

[33] M. Lundholm, "Introduction to R's time series facilities," ver. 1.3, 22 September 2011, <http://people.su.se/~lundh/reproduce/introduction_ts.pdf> 12 March 2013.

[34] R.J. Hyndman, „CRAN Task View: Time Series Analysis," 10 March 2013, <http://cran.r-project.org/web/views/TimeSeries.html> 12 March 2013.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[35] A.I. McLeod, H. Yu, and E. Mahdi, „Time Series Analysis in R," Handbook of Statistics, Volume 30, Elsevier, 27 July 2011, <http://www.stats.uwo.ca/faculty/aim/tsar/tsar.pdf> 12 March 2013.

[36] R Development Core Team "R Data Import/Export." R Foundation for Statistical Computing,Vienna, Austria, ver. 2.15.3, 1 March 2013, ISBN 3-900051-10-0, <http://www.R-project.org> 12 March 2013.

[37] G. Grothendieck, "Time series in half hourly intervals- how do i do it?" R-SIG-Finance news group, 27 Septemper 2010, < https://stat.ethz.ch/pipermail/r-sig-finance/2010q3/006729.html> 12 March 2013.

[38] A. Coghlan, "Little Book of R for Time Series," 2010, <http://a-little-book-of-r-for-time-series.readthedocs.org/en/latest/> 12 March 2013.

[39] R.J. Hyndman and G. Athanasopoulos, "Forecasting: principles and practice, chapter 8/9 Seasonal ARIMA models," online textbook, March 2012, <http://otexts.com/fpp/8/9/> 12 March 2013.

[40] R.J. Hyndman, "Cyclic and seasonal time series," in Hyndsight, 14 December 2011, <http://robjhyndman.com/hyndsight/cyclicts/> 12 March 2013.

[41] S.F. Crone, "Forecasting with Artificial Neural Networks," Tutorial at the 2005 IEEE Summer School in Computational Intelligence EVIC'05, Santiago, Chile, 15 December 2005, <http://www.neural-forecasting.com/tutorials.htm> 12 March 2013.

[42] NIST/SEMATECH, „Chapter 6.4. Introduction to Time Series Analysis," in e-Handbook of Statistical Methods, created 1 June 2003, updated 1 April 2012, <http://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4.htm> 12 March 2013.

[43] H.B.Nielsen, „Non-Stationary Time Series and Unit Root Tests," Lecture for Econometrics II, Department of Economics, University of Copenhagen, 2005, <http://www.econ.ku.dk/metrics/Econometrics2_05_II/Slides/08_unitroottests_2pp.pdf> 12 March 2013.

[44] K. Bhattach, M. Stigler, and J.C. Frain, "Which one is better?" Discussion in RMetrics, 1 Janueary 2010, <http://r.789695.n4.nabble.com/Which-one-is-better-td991742.html> 12 March 2013.

[45] R.F. Nau, „Seasonal ARIMA models," Course notes for Decision 411 Forecasting, Fuqua School of Business, Duke University, 16 May 2005, <http://people.duke.edu/~rnau/seasarim.htm> 12 March 2013.

[46] „Chapter 4: Seasonal Models," in STAT 510 - Applied Time Series Analysis, online course at Department of Statistics, Eberly College of Science, Pennsylvania State University, 2013, < https://onlinecourses.science.psu.edu/stat510/?q=book/export/html/50 > 12 March 2013.

[47] R.J. Hyndman, "Forecasting with long seasonal periods," in Hyndsight, 29 September 2010, <http://robjhyndman.com/hyndsight/longseasonality/> 12 March 2013.

[48] R Development Core Team "arima0: ARIMA Modelling of Time Series – Preliminary Version" in R-Documentation, R Foundation for Statistical Computing,Vienna, Austria, 2010, ISBN 3-900051-07-0, <http://stat.ethz.ch/R-manual/R-patched/library/stats/html/arima0.html> 12 March 2013.

[49] "Multiprocessor scheduling," in Wikipedia: the free encyclopedia, San Francisco (CA): Wikimedia Foundation, 12 March 2012, <http://en.wikipedia.org/wiki/Multiprocessor_scheduling> 19 March 2012.

[50] T. Groves, J. Knockel, and E. Schulte, "BFS vs. CFS - Scheduler Comparison," 11 December 2011 < http://slimjim.cs.unm.edu/~eschulte/data/bfs-v-cfs_groves-knockel-schulte.pdf > 11 May 2012.

[51] M. Gusev and S. Ristov, "The Optimal Resource Allocation Among Virtual Machines in Cloud Computing," in CLOUD COMPUTING 2012: The Third International Conference on Cloud Computing, GRIDs, and Virtualization, IARIA, 22 July 2012, pp. 36-42.

[52] V. Babka, P. Libič, T. Martinec, and P.Tůma, "On The Accuracy of Cache Sharing Models," in Proceedings of ICPE 2012, Boston, USA, ACM, April 2012, pp. 21-32, ISBN 978-1-4503-1202-8.

APPENDIX A – EXPERIMENTAL TIME SERIES AND THEIR FORECASTS FROM HOLT-WINTERS AND BOX-JENKINS

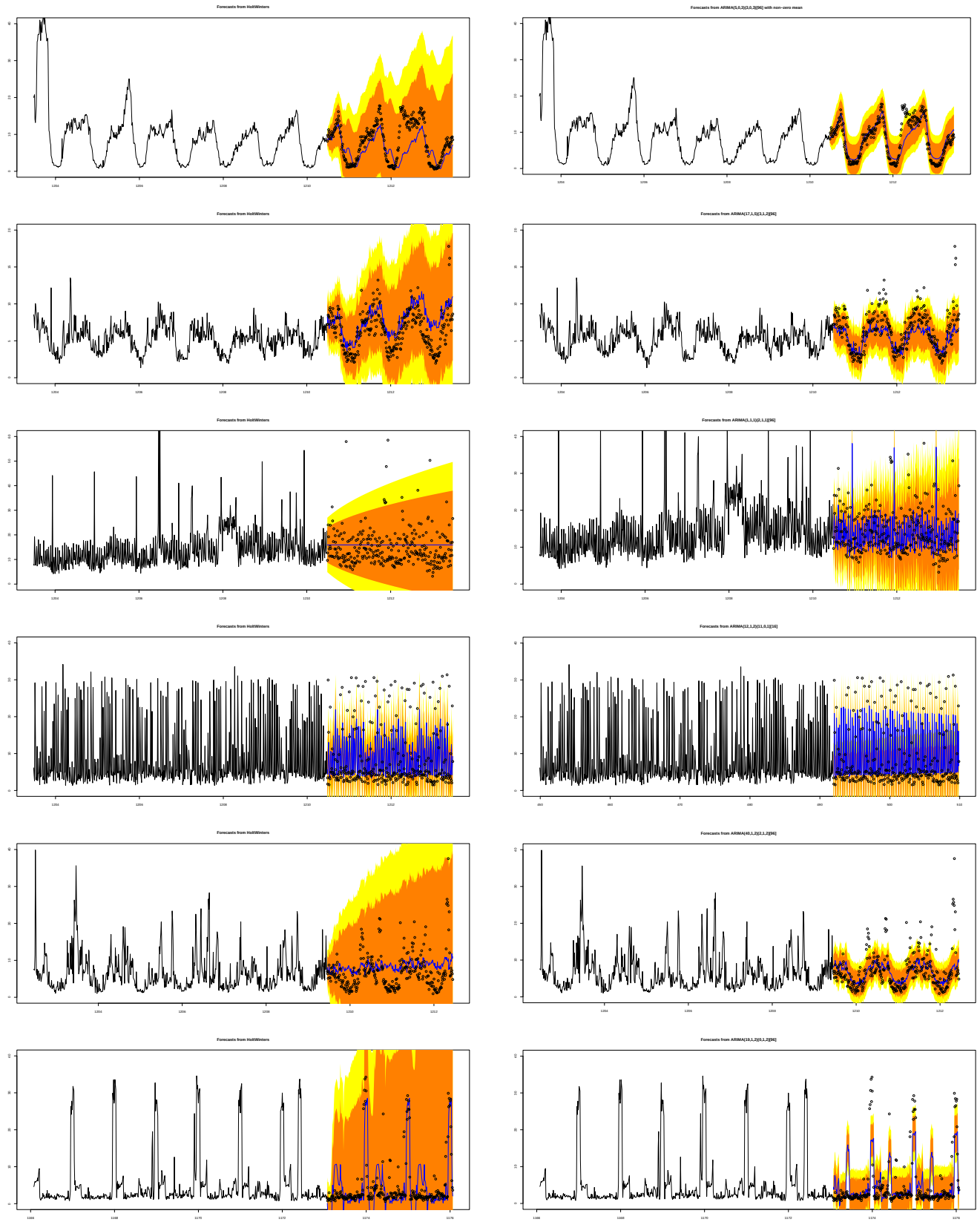
The next page contains the forecasts of each examined time series from the best model of exponential smoothing and ARIMA methods.

The exponential smoothing on in the left half of the page, ARIMA on the right. The series are, from top to bottom: oe, bender, lm, real, wn, gaff.

The graphs contain the last week of the time series to present their character. The blue line then represents the point forecasts; the orange area is the 80% confidence band and the yellow area the 95% confidence band. Overlaid as "o" symbols are the actual data points, which were recorded during the forecast horizon.

It is not important to read the axes of the graphs, the scale is 2 days per tick on the x axis and in percent of an unspecified CPU on y. The character of the time plot and the response of the forecasting algorithms is important.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

RobustMAS: Measuring Robustness in Hybrid Central/Self-Organising Multi-Agent Systems

Yaser Chaaban and Christian Müller-Schloer
Institute of Systems Engineering
Leibniz University of Hanover
Hanover, Germany
e-mails: {chaaban, cms}@sra.uni-hannover.de

Jörg Hähner
Institute of Organic Computing
University of Augsburg
Augsburg, Germany
e-mail: joerg.haehner@informatik.uni-augsburg.de

***Abstract*—It is noteworthy that the definition of system robustness varies according to the context in which the system is used. Therefore, manifold meanings of system robustness were introduced in literature. Additionally, various formal measures and metrics were presented to achieve the system robustness. In previous papers, we proposed a new concept to keep a multi-agent system at a desired performance level when deviations from planned (desired) behaviour occur in the system (robustness). This concept introduces a robust hybrid central/self-organising multi-agent system. The scenario used in this work is a traffic intersection without traffic lights. In this paper, we analyse two previous quantitative approaches presented, among others, in the literature towards a generalised robustness metric. Furthermore, we extend our prototype implementation with the aim of making it capable of handling disturbances (accidents) occur in the system environment (intersection) aiming to completely realise our vision. Simultaneously, we develop an appropriate metric for the quantitative determination of the robustness. The experimental results demonstrated a high degree of the robustness of the developed concept against disturbances.**

Keywords-Robustness; Organic Computing; Hybrid Coordination; Multi-Agent Systems; Performance measurement systems

I. INTRODUCTION

This article is an extension of a previously published paper [1]. Organic Computing (OC) has the objective to use principles that are detected in natural systems. In this case, nature can be considered as a model for technical systems aiming to cope with the increasing complexity [2][3]. Consequently, OC tries to develop systems that are adaptive, flexible and robust at the same time utilising advantage of the organic properties of OC. In this regard, the robustness of OC systems is a key property, because the environments of such systems are dynamic.

Organic systems or autonomic systems [4][5] try to realise quality in several aspects of system engineering including: functional correctness, safety, security, robustness/reliability, credibility, and usability [6][7].

In organic systems, the design of the system architecture plays a main role in achieving a robust system so that its performance has to remain acceptable in the face of deviations or disturbances occurred in the system (intern) or in the environment (extern). That means, the development of robust systems needs to take into account that degradation of the system's performance in the presence of such disturbances should be limited in order to maintain a satisfying performance. Therefore, a robust system has the capability to act satisfactorily even when conditions change from those taken into account in the system design phase. Nevertheless, this capability has to be retained, because of the increasing complexity of novel systems where the environments change dynamically. As a result, fragile systems may fail unexpectedly even due to slightest disturbances. Thus, a robust system will continue working in spite of the presence of disturbances by counteracting them with corrective interventions.

Considering the system design paradigm, it should be decided whether the system architecture will be centralised or decentralised. A centralised approach is the paradigm where the system is based on a centralised architecture (there is a central controller and the components of the system are not fully autonomous). On the other hand, a decentralised approach means that the system has a distributed (there is no central controller and all components of the system are autonomous) or a hierarchical architecture (the components of the system are semi-autonomous in which they are locally centralised) [8]. Based on this, distribution possibilities of system architecture have important implications for system robustness.

Although the decentralised approach would have some advantages over the centralised one, especially scalability, the hybrid approach containing both centralised and decentralised elements at the same time is applicable and even may be much better than the use of each one separately. The hybrid approach should be robust enough against disturbances, because robustness is an indispensable property of novel systems. Additionally, it represents the interaction between decentralised mechanisms and centralised interventions. In other words, the hybrid approach exhibits the central/self-organising traits simultaneously. This means that a conflict between a central controller (e.g., a coordination algorithm) and the autonomy of the system components must be solved in order to achieve the robustness of the system.

For this purpose, OC uses an observer/controller (O/C) architecture as an example in system design. Using the O/C design pattern proposed in [9], the behaviour of OC systems can be observed and controlled. A generic O/C architecture

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

was presented in [10] to establish the controlled self-organisation in technical systems. This architecture is able to be applied to various application scenarios.

During the last years, the progress in communication and information technologies was significant. Consequently, a lot of investigations were done aiming to improve transport systems so that the "Intelligent Transportation Systems (ITS)" was developed. ITS have several applications in traffic and automotive engineering. According to ITS, numerous notions were distinguished such as, among others, intelligent vehicles, intelligent intersections, and autonomous vehicles. In this context, a traffic intersection without traffic lights was chosen as a main testbed to apply the hybrid approach, where autonomous agents are autonomous vehicles, and the controller of the intersection is the central unit. However, the basic idea of a hybrid approach is applicable for other systems as well.

This paper is organised as follows. Section II describes our original system introduced in [11][12]. Section III presents a survey of related work concerning robust agent-based approaches used for fully autonomous vehicles within an intersection without traffic lights, in addition to various methods for measuring robustness. Section IV is the main part of this paper. Firstly, it describes the interdisciplinary methodology, "Robust Multi-Agent System" (RobustMAS), developed in this paper. After that, it presents the measurement of robustness and gain according to the RobustMAS concept. Section V introduces the evaluation of the system performance by means of experimental results. Section VI draws the conclusion of this work. Finally, the future work is explicated in Section VII.

II. The Original System

This paper is an extended version of our conference paper [1] presented at Cognitive2012. With respect to [1], this paper presents an expanded discussion of related work, allowing us to analyse two previous quantitative approaches towards a generalised robustness metric. Furthermore, the robustness measurement will be considered in two ways in this paper, while there was only one way in [1]. Finally, this paper shows detailed version of results using cumulative throughput values in upper figures and throughput values per time unit in lower figures.

In previous papers, we introduced a system for coordinating vehicles at a traffic intersection using an O/C architecture [11][12]. The traffic intersection is regulated by a controller, instead of having physical traffic lights. Figure 1 shows a screenshot from our project. In this regard, we proposed a new multi-agent approach which deals with the problem occurring in the system wherever multiple agents (vehicles) move in a common environment (traffic intersection without traffic lights). We presented the desired system architecture together with the technique that is to be used to cope with this problem. This architecture was an O/C architecture adapted to the scenario of traffic intersection.

In both earlier papers, we implemented the generic O/C architecture adapted to our traffic scenario and accomplished our experiments assuming that no deviations from plan occur in the system. The evaluation of the concept was carried out based on the basic metrics: throughput, waiting time and response times [11] [12].

Moreover, specifying the desired behaviour of agents in a shared environment was considered in [13]. So, we presented a convenient method to achieve such desired behaviour. For this purpose, A*-algorithm for path planning of agents (vehicles) was proposed [13].

Additionally, handling of deviations from planned (desired) behaviour was studied in [14]. To address this issue, we extended our prototype implementation with the aim of making it capable of handling deviations from planned behaviour. In this way, the hybrid central/self-organising concept tolerates that some agents behave autonomously. Here, the autonomy of the agents is recognised as a deviation from the plan of the central algorithm, if the agents are not respecting this plan [14].

Furthermore, we provided an overview of a several robustness approaches in multi-agent systems (MAS) in [15]. The survey is concerned with MAS in a variety of research fields.

In this paper, we continue with the implementation of the case when disturbances (accidents) arise in the system (intersection) to completely realise our vision. Consequently, the system performance remains effective and will not deteriorate significantly or at least the system will not fail completely.

Additionally, an appropriate metric for the quantitative determination of the robustness will be developed and presented in this paper.

III. State of the Art

Keeping a system at a desired performance level in presence of disturbances or deviations from plan has been investigated by researchers for years. Consequently, many approaches or architectures were introduced towards building robust systems.

In the literature, there are enormous works concerning safety properties of usual traffic intersections that concerns only human-operated vehicles. Additionally, there are some works in connection with safety measures of autonomous vehicles within an intersection. In this paper, we focus the discussion of related work on robust agent-based approaches used for fully autonomous vehicles within an intersection without traffic lights. Furthermore, we consider various methods for measuring robustness.

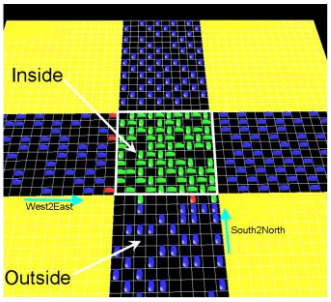


Figure 1. The traffic intersection without traffic lights

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

In this regard, according to our knowledge, there are no projects that focus on the robustness of autonomous vehicles within an intersection without traffic lights, where disturbances occur.

A study of the impact of a multi-agent intersection control protocol for fully autonomous vehicles on driver safety is presented in [16]. In this study, the simulations deal only with collisions in intersections of autonomous vehicles aiming to minimise the losses and to mitigate catastrophic events. However, it can be noted that the study has not considered the robustness of the intersection system.

A. Measures for robustness

In order to have the ability to design robust multi-agent systems, robustness metrics are required. These metrics play the role to mitigate the expected degradation of the system performance when any disturbances occur. Many research projects deal with system robustness. Their objective is to measure robustness and to find an appropriate metric for it. These projects are in various kinds of science.

There is a clear lack of study of these metrics in designing robust multi-agent systems. This paper raises the question how the robustness can be guaranteed and measured in technical systems.

In literature, there are diverse potential measures of system robustness proposed. Every robustness measure is based and designed according to the definition of the robustness concept in a specific context. The most common robustness measure uses the robustness definition related to the definition of a performance measure. Some robustness measures estimate the system performance using the average performance and its standard deviation, the signal-to-noise ratio, or the worst-case performance. Other robustness measures take into account the probability of failure of a system as well as the maximum deviation from a benchmark where the system has still the ability to deal with failures [17].

B. Generalised robustness metric

Viable quantitative approaches in order to measure robustness are required. Some approaches were introduced, among others, in [18][19][20]. Among those, both the FePIA (Features Perturbation Impact Analysis) procedure in [18] and the statistical approach in [19] are general approaches and consequently can be adapted to specific purposes (arbitrary environment). In both approaches, diverse general metrics were used to quantify robustness. This metrics estimate specific system features in the case of disturbances (perturbations) in components or in the environment of the system. Additionally, these metrics were mathematically described. Both approaches in [18] and in [19] are applicable in embedded systems design [20] where embedded systems are designed as Systems on Chip (SoC).

In the following, the FePIA procedure and the statistical approach will be explained.

1) FePIA procedure

The FePIA procedure is presented in [18] in order to derive a robustness metric so that it can be used for an arbitrary system. The authors there discussed the robustness of resource allocations in parallel and distributed computing systems. Consequently, a derived metric from the FePIA procedure was designed for a certain allocation of independent applications in a heterogeneous distributed system demonstrating the utility of the robustness metric. Here, the goal was to maximise the robustness of the produced resource allocations. Moreover, the authors have defined the robustness (indeed, a resource allocation is to be robust) as a restricted degradation of the system performance against uncertainties (perturbations) in specified system parameters.

FePIA stands for Features Perturbation Impact Analysis. The FePIA procedure defines a schema that presents a robustness-radius for the system based on a tolerance region. This procedure identifies four general steps [18][20]:

1. The important system performance features f_i that may cause degradation of the system performance. They are combined into a feature vector Φ: $\Phi = \{\varphi_1, ..., \varphi_n\}$.
2. The perturbation parameters: $\pi = \{\pi_1, ..., \pi_m\}$.
3. The impact of perturbation parameters on system performance features. This is modelled with individual functions $f_{ij} : \pi_i \rightarrow \varphi_j$, selecting a tolerance region (β_j^{min} , β_j^{max}) for each φ_j (see Figure 2).
4. The analysis (it analyses the values of π_i) to determine the degree of robustness.

The main point here is to produce a mathematical relationship between the system performance features and the perturbation parameters (in the sense of the impact). After that, a variation in the perturbation parameters, which lead to a performance degradation exceeding the allowable performance limits (tolerance region), can be detected. This variation represents the robustness radius (optimisation problem) [19].

So, r (φ_j , π_i) represents the robustness-radius of the system according to the system performance feature φ_j and the perturbation parameter π_i. Accordingly, in order to calculate the robustness of the whole system in the case of a certain perturbation parameter, the minimum across all features of system performance has to be found. Figure 2 illustrates the FePIA procedure.

Here, a tolerance region is defined by a lower boundary (β^{min}) and an upper boundary (β^{max}), which can be expressed as in the next formulas:

$$\beta^{min} = \min \left\{ f\left(\pi^{orig} - r\right), f\left(\pi^{orig} + r\right) \right\} \quad (1)$$

$$\beta^{max} = \max \left\{ f\left(\pi^{orig} - r\right), f\left(\pi^{orig} + r\right) \right\} \quad (2)$$

A robustness definition for analog and mixed signal systems was derived in [20] using the FePIA procedure. The author has evaluated the proposed robustness formula applying affine arithmetic (modelling the deviations by affine

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

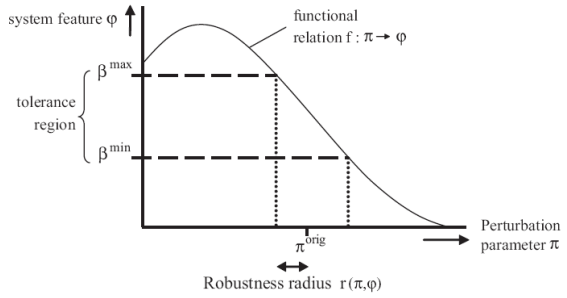


Figure 2. The general FePIA procedure [20]

expressions as in [21]) with a semi-symbolic simulation. The symbolic representation used in semi-symbolic simulations makes designers aware of the contribution of uncertainty to the deviation at the output of the simulated system. Also, the outcomes of the simulation are affine expressions, which semi-symbolically represent possible deviations [21].

As a result, a robustness definition for analog and mixed signal systems was derived that is based on the estimation of precision versus the robustness radius using the FePIA procedure as described in the next formula:

$$robustness(\varphi,\pi) := \frac{r(\varphi,\pi)}{rad(\pi)} \quad (3)$$

where $rad(\pi)$ characterises the confidence interval of deviations from π [20].

According to this formula, which can be used in the design phase, three cases can be considered.

- First, the robustness is less than 1 and hence the system is not robust and it may fail.
- Second, the robustness is equal to 1 and therefore the system is robust to some extent and it fulfils the minimum requirements.
- Third, the robustness is greater than 1 and hence the system is robust against additional deviations [20].

The drawback of the FePIA procedure is that the tolerance regions (the limits of the performance features) are arbitrarily selected. Thus, the FePIA procedure is applicable for systems where the system performance and the tolerable deviations can be well-defined [20].

2) Statistical approach

The statistical approach has been introduced by England et al. in [19] to obtain a type of robustness metric, which can be used for an arbitrary system. The authors there present a methodology aiming to characterise and measure the robustness of a system (using a quantitative metric) in the face of a specific disturbance (perturbation).

The authors define robustness as follows: "Robustness is the persistence of certain specified system features despite the presence of perturbations in the system's environment." [19].

Similar to the FePIA procedure, system performance features in the statistical approach will be taken into consideration versus the perturbation size (disturbance size). Therefore, the intention of the authors was to measure the amount of degradation of the system performance relative to the perturbation size [20][19]. For this purpose, the cumulative distribution function (CDF) of a system performance feature is used. CDF is the proportion of observations less than or equal to a specified value (x) when a set of performance observations (X) is given [19]. The robustness can be determined according to the difference between functions F and F*. The function F is the CDF of a performance feature in the case of normal operating conditions; whereas the function F* is the CDF of a performance feature in the case of perturbations.

The maximum distance between F and F* represents the amount of performance degradation. This distance (δ) was computed by means of the Kolmogorov-Smirnov (K-S) statistic (sup is the supremum):

$$\delta = \sup_{-\infty<x<\infty} \left(F(x) - F^*(x)\right) \quad (4)$$

Moreover, the distance (δ) has to be weighted with a weighting function (to compensate for the underestimation of δ) producing the adjusted K-S statistic (δ_w):

$$\delta_w = \sup_{-\infty<x<\infty} \left(F(x) - F^*(x)\right) \Psi(x) \quad (5)$$

The advantage of this method is that it considers the complete distribution of system performance (performance observations); whereas other methods consider only average measurements. In this context, it can be inferred that the system is robust against the applied perturbation when the distance between F and F* (the amount of performance degradation) is very small. Therefore, the smaller the distance is, the more robust the system becomes. Figure 3 illustrates the statistical approach (the adjusted K-S statistic) [19].

In Figure 3, the robustness of a system is characterised by the measurement of δ_w as a function of the applied perturbation size (in other words, by the gradient of δ_w relative to the amount of perturbation experienced [20]). This means that this system can withstand different levels of perturbation. Here, three cases can be recognised.

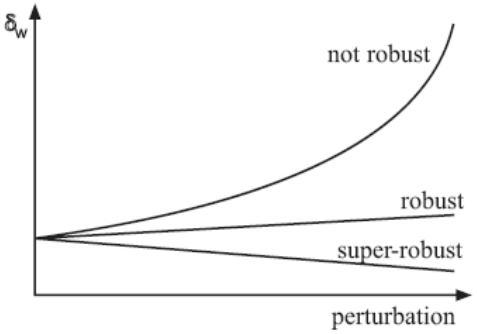


Figure 3. Characterising the robustness of a system according to the statistical approach [20]

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

First, the robust system, wherein δ_w exhibits a slight increase with increasing the perturbation size. Second, non-robust system, wherein δ_w shows a great (probably non-linear) increase with increasing the perturbation size. Third, the super-robust system, wherein δ_w exhibits a slight decrease with increasing the perturbation size. The perturbation in the last case is a profitable perturbation (see [19] for an example).

According to [20], the proposed robustness metric based on the statistical approach is appropriate to use in the design process, where it acts as absolute robustness indicator for profiling targets. In this case, specifications must be executable, so that simulations can be carried out to supply an adequate amount of statistical data.

Comparing with the FePIA procedure, this methodology is generally applicable to various classes of computing systems. Also, it is easier to determine the robustness. That means, the statistical approach has avoided the drawback of the FePIA procedure, so that a tolerance region needs not to be formed. Additionally, they employed their methodology in three applications of job scheduling: backfilling jobs on supercomputers (parallel machines), overload control in a streaming video server, and routing requests in a distributed network service. The third application shows the role of robustness to obtain improvements in system design. Additionally, as mentioned above, this robustness metric would have the advantage of the consideration of the complete distribution of system performance.

C. Summary: Measures for robustness

Several research projects propose diverse measures of system robustness. These projects measure robustness according to their definition of the robustness in different application areas. In this context, some quantitative approaches were used, such as the FePIA procedure in [18] and the statistical approach in [19]. However, there is a clear lack of study of the robustness metrics in designing robust multi-agent systems in technical systems. Therefore, there still is the question how the robustness can be guaranteed and measured in technical systems. As a result, both approaches discussed above do not comply with the RobustMAS concept introduced in this paper to characterise robustness.

This non-compliance can be traced back to the fact that RobustMAS focuses on the robustness of hybrid central/self-organising multi-agent systems. For this purpose, RobustMAS proposes the concept of relative robustness for measuring the ability to maintain a specific minimum level of system performance (a desired performance level) in the presence of deviations from desired behaviour (e.g., unplanned autonomous behaviour) and disturbances in the system environment. Based on this, according to the RobustMAS concept, robustness is the ability of the system, with minimal central planning intervention, to return after disturbances (internal and external changes) to the normal state.

To the best of our knowledge, this paper represents the first study towards measuring the robustness of hybrid central/self-organising multi-agent systems in intersections without traffic lights using the organic computing (OC) concept.

IV. The Approach

The Organic Computing initiative aims to build robust, flexible and adaptive technical systems. Future systems shall behave appropriately according to situational needs. But this is not guaranteed in novel systems, which are complex and act in dynamically changing environments.

The focus of this paper is to investigate and measure the robustness of coordination mechanisms for multi-agent systems in the context of Organic Computing. As an application scenario, a traffic intersection without traffic lights is used. Vehicles are modelled as agents.

A. Robust Multi-Agent System (RobustMAS)

An interdisciplinary methodology called “Robust Multi-Agent System” (RobustMAS), has been developed and evaluated regarding different evaluation scenarios and system performance metrics.

The new developed methodology (RobustMAS) has the goal of keeping a multi-agent system running at a desired performance level when disturbances (accidents, unplanned autonomous behaviour) occur (for details see Definition 4: *Disturbance strength*). The result is an interaction between decentralised mechanisms (autonomous vehicles) and centralised interventions. This represents a robust hybrid central/self-organising multi-agent system, in which the conflict between a central planning and coordination algorithm on one hand and the autonomy of the agents on the other has to be solved.

The hybrid coordination takes place in three steps:

1. A course of action with no disturbance: central planning of the trajectories without deviation of the vehicles.
2. Observation of actual trajectories by an Observer component, identifying deviations from plan.
3. Replanning and corrective intervention.

In the scenario of this paper, an intersection without traffic lights, the participants are modelled as autonomous (semi-autonomous) agents (Driver Agents) with limited local capabilities. The vehicles are trying as quickly as possible to cross the intersection without traffic lights.

An intersection manager is responsible for coordinating tasks. It performs first a path planning to determine collision-free trajectories for the vehicles (central). This path planning is given to vehicles as a recommendation. In addition, an observation of compliance with these trajectories is done, since the vehicles are autonomous (decentralised) and thus deviations from the plan in principle are possible. Of particular interest is the ability of the system, with minimal central planning intervention, to return after disturbances to the normal state.

For the path planning, common path search algorithms are investigated in our earlier paper [11]. Particularly interesting here is the A*- algorithm. The path planning is considered as a resource allocation problem (Resource Allocation Conflict), where several agents move in a shared environment and have to avoid collisions. The

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

implementation was carried out under consideration of virtual obstacles. Virtual obstacles model blocked surfaces, restricted areas (prohibited allocations of resources), which may arise as a result of reservations, accidents or other obstructions. In addition, virtual obstacles can be used for traffic control.

In [13], we focused on planning of the desired behaviour of agents in a shared environment. Based on this, an adapted A*-algorithm for path planning of agents has been applied. The adaptation was necessary for the requirements of the used traffic scenario, because a vehicle can only take a "rational" path, whereas an agent (e.g., robot) can take any calculated path. Consequently, the designed algorithm calculates collision-free trajectories (central planning) for all agents (vehicles) in a shared environment (the centre of the intersection) enabling them to avoid collisions. The experimental results demonstrated a high performance of our adapted A*- algorithm.

Different types of deviations of the vehicles from the plan have been investigated in our previous paper [11]. The controller is informed by the observer about the detected deviations from the plan, so that it can intervene in time. The controller selects the best corrective action that corresponds to the current situation so that the target performance of the system is maintained.

In this paper, we introduce an appropriate metric for the quantitative determination of the system robustness. The robustness measurement will be made when disturbances (accidents) occur in the system (intersection).

B. *Measurement of robustness and gain according to the RobustMAS concept*

Since RobustMAS aims to keep a multi-agent system at a desired performance level even though disturbances and deviations occur in the system, a new appropriate method to measure the robustness of a multi-agent system is required. The equivalent goal of RobustMAS by the application scenario, a traffic intersection without traffic lights, is to keep the traffic intersection at a desired performance level even though deviations from the planned trajectories and accidents occur in the intersection. Therefore, a new concept will be introduced in order to define the robustness of multi-agent systems. Additionally, the gain of RobustMAS will be defined and used to show the benefit of the hybrid central/self-organising concept.

According to the RobustMAS concept, the robustness of a multi-agent system can be defined as follows:

Definition 1: Robustness.

"A (multi-agent) system is considered robust against disturbances if its performance degradation is kept at a minimum".

Consequently, the RobustMAS concept assumes that a robust system keeps its performance acceptable after occurrence of disturbances and deviations from the plan.

Definition 2: Relative robustness.

"The relative robustness of a (multi-agent) system in the presence of a disturbance is the ratio of the performance degradation due to the disturbance divided by the undisturbed performance".

In order to measure the robustness of RobustMAS in the traffic intersection system, the throughput metric is used for determining the reduction of the performance (system throughput) of RobustMAS after disturbances (accidents) and deviations from the planned trajectories. That is because throughput is one of the most commonly used performance metrics. Therefore, the comparison of the throughput values is required in the three cases:

(1) Without disturbance.
(2) With disturbance with intervention.
(3) With disturbance without intervention.

Based on this, the robustness measurement of RobustMAS will be considered in two ways:

- Using cumulative system performance, i.e., cumulative throughput (# Agents), where the system is considered only until the time when the disturbance ends.
- Using system performance, i.e., throughput per time unit (# Agents/sec), where the system is considered until the time when the system returns after disturbances to its normal state like before.

For this explanation of the robustness measurement, the words agent and vehicle can be used interchangeably.

1) Using cumulative system performance (cumulative throughput)

Figure 4 illustrates this comparison where t_1 is the time at which the disturbance (accident) occurs. The disturbance is assumed to remain active until the time t_2. This figure shows the cumulative performance (throughput) values of the system before and after the disturbance comparing the three mentioned cases.

The black curve is the performance (throughput) of the system if no disturbance occurs. The green curve is the performance of the system when a disturbance at time t_1 occurs and the central planning intervenes on time. The system is considered until time t_2 when the disturbance ends. The red curve is the performance of the system when a disturbance at time t_1 occurs and the central planning does not intervene. Here, two areas can be distinguished: $Area_1$ and $Area_2$ in order to measure the robustness of RobustMAS as depicted in Figure 5.

This figure shows the idea of how the robustness of the system as well as the gain of the system can be determined according to the RobustMAS concept.

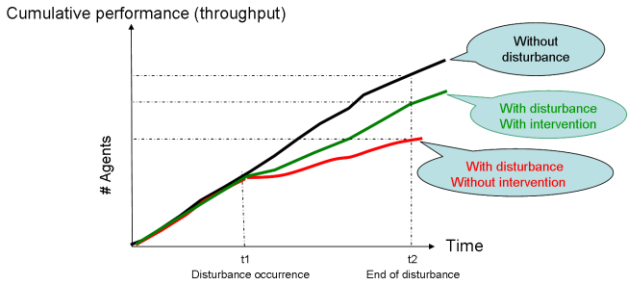


Figure 4. Comparison of cumulative system performance (throughput) for three situations

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

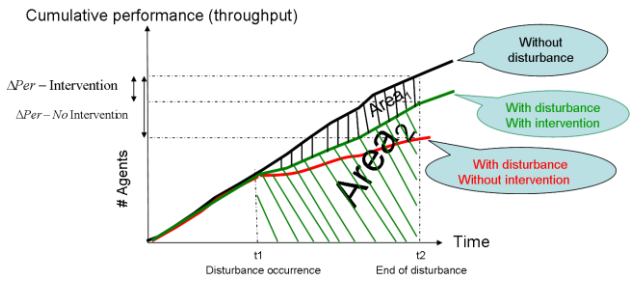


Figure 5. Measuring robustness and gain using cumulative system performance

The relative robustness (R) of a system (S) is determined as follows:

$$R = \frac{\text{Area}_2}{\text{Area}_1+\text{Area}_2} = \frac{\int_{t1}^{t2} Per(t)_{(withIntervention)} d(t)}{\int_{t1}^{t2} Per(t)_{(NoDisturbance)} d(t)} \quad (6)$$

This means that the robustness is $Area_2$ divided by the sum of the two areas 1 and 2. $Area_2$ is the integral of the green curve (disturbance with intervention) between t_1 and t_2. The sum of $Area_1$ and $Area_2$ is the integral of the black curve (no disturbance) between t_1 and t_2.

Additionally, the gain of the system can be used as a secondary measure. In this context, the gain of a system can be defined according to the RobustMAS concept as follows:

Definition 3: Gain.

"The gain of a system is the benefit of the system through central planning (compared to decentral planning). Accordingly, the gain of a system represents the difference between the system performance (throughput) in the two cases, with and without intervention of the central planning algorithm".

This issue is expressed by the following equation:

$$Gain = \Delta Per_{(NoIntervention)} - \Delta Per_{(Intervention)} \quad (7)$$

As depicted in Figure 5, the gain of the system can be calculated using the values of the system performance (throughput values) at the time t_2. Here, $\Delta Per(Intervention)$ represents the difference between the system performance in the two cases, without disturbance and disturbance with intervention of the central planning algorithm; whereas $\Delta Per(NoIntervention)$ represents the difference between the system performance in the two cases, disturbance with and without intervention of the central planning algorithm.

2) Using system performance (throughput per time unit)

In this case, the system performance, i.e., throughput per time unit (# Agents/sec) is used. Additionally, the system is considered longer than in the case of the cumulative performance (cumulative throughput) values. Therefore, compared to that case that defines time t_1, the occurrence time of disturbance, and time t_2, the end time of disturbance,

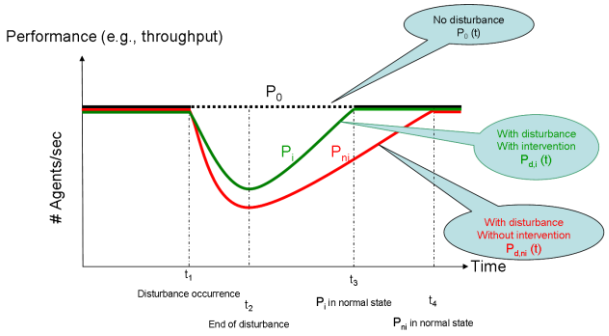


Figure 6. Comparison of system performance (throughput per time unit) for three situations

the times t_3 and t_4 will also be defined. Here, t_3 is the time at which the system returns to its normal state with minimal central planning intervention, while t_4 is the time at which the system returns to its normal state without central planning intervention. In this regard, the normal state represents the system performance level at its best when no disturbances occur (under normal operating conditions).

Here, we use the following functions:

- P_0 (t): represents the system performance when no disturbances occur (normal state).
- $P_{d, ni}$ (t): represents the system performance with a disturbance with no intervention by the central planning.
- $P_{d, i}$ (t): represents the system performance with a disturbance with an intervention of the central planning.

Figure 6 shows the performance (throughput per time unit) values of the system before and after the disturbance until the time when the system returns to its normal state like before comparing the three mentioned cases.

In accordance with the definition 2 mentioned above, the relative robustness (R) of a system (S) is determined as follows:

$$R = \frac{\int_{t1}^{t4} P_{d,i}(t) d(t)}{\int_{t1}^{t4} P_0(t) d(t)} \; ; \quad 0 \le R \le 1 \quad (8)$$

Here, the lower and upper boundaries can be set as follows:

- R = 0 represents the lower boundary case of the relative robustness, where the system is considered as non-robust against disturbances (very poor performance). It appears when $P_{d, i}$ (t) << P_0 (t), i.e., the performance degradation is very strong due to the disturbance in spite of the intervention, compared to the performance when no disturbance occurs. Thus, the system behaviour is not acceptable in the face of disturbances.
- R = 1 represents the upper boundary case of the relative robustness, where the system is considered as strongly robust against disturbances (an optimal performance, an ideal behaviour). It occurs, when

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$P_{d,i}$ (t) = P_0 (t), i.e., there is no performance degradation due to the intervention despite the presence of disturbances.

Furthermore, the system could be also weakly robust if its performance level is acceptable but not optimal in the presence of disturbances. Therefore, the system behaviour is acceptable but not ideal.

Similar to the definition 3 mentioned above, the gain of a system is determined as the difference between the performance in both cases, disturbances with and without intervention:

$$Gain(i \rightarrow ni) = \#Agents(i) - \#Agents(ni) = \int_{t1}^{t4} [P_{d,i}(t) - P_{d,ni}(t)]\, d(t) \quad (9)$$

Consequently, the loss of a system is determined as the difference between the performance in both cases, no disturbance and disturbances with intervention:

$$Loss = \int_{t1}^{t4} [P_0(t) - P_{d,i}(t)]\, d(t) \quad (10)$$

The discussion of the robustness measurement using the system throughput metric will be based on the parameter *disturbance strength*. In this regard, the disturbance strength can be defined according to the RobustMAS concept as follows:

Definition 4: Disturbance strength.

"A disturbance strength is a positive constant defining the strength (size) of the disturbance".

This parameter represents the size of the accident in the used traffic system. Accordingly, the robustness measurement was repeated in the cases that the disturbance strength is 1, 2, and 4. That means, the accident occupies an area of size 1, 2 and 4 cells in the traffic intersection as depicted in Figure 7.

Obviously the disturbance strength influences the system performance, which in turn leads to different degrees of system robustness. When the disturbance strength is increased, then the system performance will be reduced. This means that the increase of the disturbance strength is inversely proportional to the degree of the system robustness.

However, the definition of system robustness can be extended to include the strength of disturbances experienced (amount of disturbances applied). Accordingly, the robustness (Rob) of a given system depending on the disturbance strength ($Dist_{str}$) can be determined in formula (11).

This means that Rob = R * $Dist_{str}$, where R is the relative robustness defined above. In this case, the integral will be between the time t_1 at which the disturbance begins, and time t_2, at which the disturbance ends. This formula implies that a system shows varying degrees of robustness (Rob) while the disturbance strength is varied.

$$Rob = \underbrace{\frac{\int_{Start\ dist.}^{End\ dist.} P_{d,i}(t)\, d(t)}{\int_{Start\ dist.}^{End\ dist.} P_0(t)\, d(t)}}_{R} * Dist_{str} \quad (11)$$

According to the used application scenario, the size of the accident influences the intersection throughput (the number of vehicles that have left the intersection area), which in turn leads to different degrees of the robustness of the intersection. When the size of the accident increases, then the intersection performance will decrease. This can be justified simply on the ground that accidents will cause obstacles for the vehicles in the intersection. These obstacles will impede the movement of vehicles which are behind the accident location. Additionally, the central plan algorithm considers the accidents as virtual obstacles (restricted areas) and therefore it limits the planned trajectories of potential traffic. The autonomous vehicles which do not obey their planned trajectories have to avoid the accident location by performing a lane change (to the right or to the left of the accident location) if it is possible as depicted in Figure 8. Certainly, autonomous vehicles have to check the possibility to avoid the accident by pulling into another lane before they take this evasive action. So, the vehicle behind the accident location tries to overtake the accident location on the right if the intended position is not occupied by another vehicle. Otherwise, if the intended position is occupied by another vehicle, then the vehicle tries to overtake the accident location on the left if the intended position is not occupied by another vehicle. If all potential intended positions are occupied, then the vehicle stops (does not change its position) and repeats this behaviour (the evasive action) again in the next simulation step.

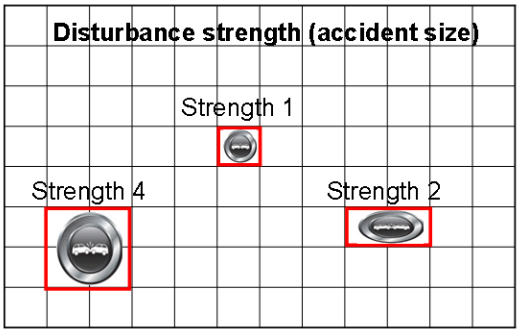


Figure 7. The disturbance strength (the accident size) in three cases: 1, 2, and 4 cells in the traffic intersection

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

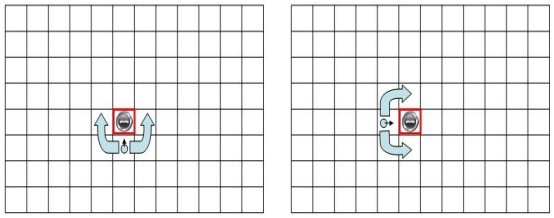


Figure 8. The evasive action of autonomous vehicles that check the possibility (right or left) to avoid the accident by pulling into another lane

V. PERFORMANCE EVALUATION

In this section, we present a complete empirical evaluation of our system using the model of a traffic intersection, which was designed and described in our earlier paper [11]. This evaluation includes experiments for measuring the robustness of the system, in which deviations from plan occur and disturbances (accidents) appear in the intersection system. That means, it deals with deviations from planned (desired) behaviour of agents (vehicles), in addition to disturbances (accidents).

A. Test situation

In this test situation, the vehicles do not obey their planned trajectories (the central plan) and thus deviations from the plan will occur as well as accidents in the intersection.

In this regard, an observation of actual trajectories by the observer will be made in order to detect any deviations from plan and to detect potential accidents in the intersection allowing the controller to make replanning for all affected trajectories using the path planning algorithm. This will be carried out via the deviation detector component and the accident detector component in the observer [11][12].

The test situation serves to measure the robustness of the traffic intersection system and to assess the degree of the robustness of RobustMAS during disturbances (e.g., accidents) and deviations (e.g., unplanned autonomous behaviour).

B. Measuring robustness and gain

As mentioned above, the throughput metric is used to determine the reduction of the performance (system throughput) of RobustMAS after disturbances (accidents) and consequently to measure the robustness of RobustMAS in the intersection system. Additionally, how the discussion of the robustness measurement is carried out depends on the disturbance strength $Dist_{str}$ (the size of the accident) involved in the experiments. As illustrated in Figure 7, $Dist_{str}$ is varied (1, 2 or 4). The results were obtained in an interval between 0 und 3000 ticks, where the maximum number of vehicles (V_{max}) is 40 vehicles in both directions and the traffic level (TL) is 5 vehicles/tick in each direction.

It can be concluded that the increase in the size of the accident is inversely proportional to the degree of the intersection robustness.

RobustMAS tries to guarantee a relatively acceptable reduction of the intersection robustness when the size of the accident increases. RobustMAS ensures at least that increasing of size of the accident will not lead to failure of the intersection.

Because the location of the accident within the intersection plays a major role in the performance of the intersection system, the simulation was repeated 10 times. Each time of repetition, an accident will be generated in a random position of the intersection by choosing a random (x, y) coordinate pair within the intersection. This (x, y) coordinate pair represents the central cell of the accident. The other cells which represent the whole accident location will be chosen also randomly depending on the value of the simulation parameter "size of accident", so that the chosen cells will surround the central cell (x, y) of the accident. So, it can be ensured that accidents will be generated in different parts of the intersection achieving more realistic study. The average values of the system throughput will be calculated from several repetitions of the simulation (random accident locations), so that a picture of how an accident would affect the system performance is created.

The simulation parameter "Disturbance occurrence time" (Accident occurrence time) represents the time (the time step in the simulation) at which the accident will be generated. The time is measured in ticks. In the simulation, the "Accident tick" was adjusted to the value of the tick "1000", i.e., an accident should be generated at tick "1000". That means, the simulation has no accident in the interval [0-1000]; whereas it has an accident in the remaining simulation interval [1000-3000] as depicted in Figure 9. Here, the system performance is the intersection throughput. The throughput is measured by the number of vehicles that left the intersection area (cumulative throughput values in the upper figure or throughput values per time unit in the lower figure).

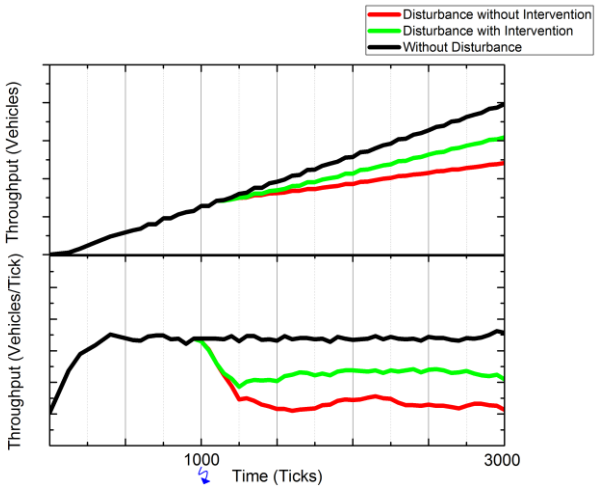


Figure 9. The "Disturbance occurrence time" adjusted to the tick 1000 and the simulation length is 3000 ticks (upper figure is cumulative throughput; lower figure is throughput per time unit)

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

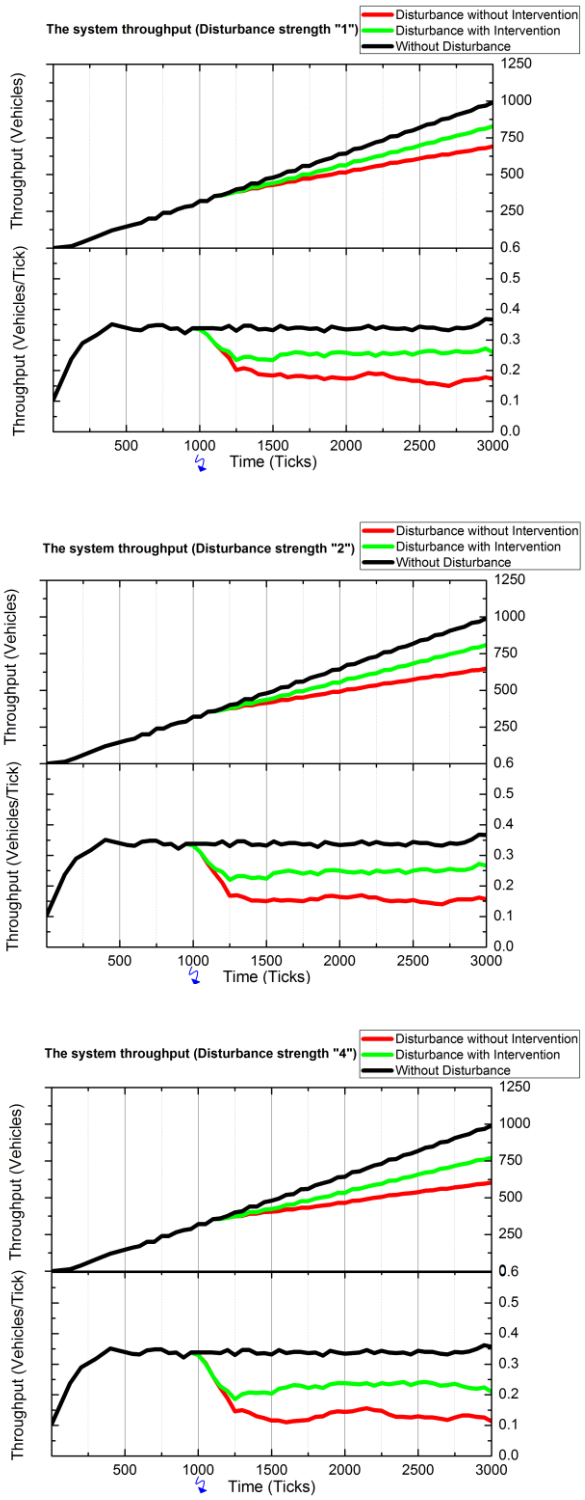


Figure 10. The system throughput per time unit (lower figure) and the cumulative system throughput (upper figure) using different values of the disturbance strength (size of the accident)

TABLE I. THE ROBUSTNESS AND THE GAIN OF THE SYSTEM FOR VARIOUS VALUES OF DISTURBANCE STRENGTH

Disturbance strength (Accident size)	Robustness (R) (%)	Gain (Vehicles)
1	87	137
2	86	161
4	83	169

The upper figure of Figure 10 shows the cumulative system performance values (throughput) of the intersection system in an interval between 0 und 3000 ticks comparing the three mentioned cases (without disturbance, disturbance without intervention and disturbance with intervention) using various values of the disturbance strength (size of the accident). Furthermore, the lower figure of Figure 10 shows the same as the upper figure using the throughput per time unit (# Vehicles/tick).

The robustness and the gain of the traffic intersection system can be determined using the two formulas of the relative robustness (R) and the gain of the system described above.

In order to see the effect of the disturbance strength (size of the accident), Table I compares the obtained results of the robustness and the gain of the system for various values of disturbance strength after 3000 ticks.

It can be concluded that when the disturbance strength increases, the robustness of the system decreases, but very slightly showing a high degree of robustness. This emphasises that a degradation of the system throughput was established when an accident has occurred in the intersection and the vehicles made deviations violating their planned trajectories. Therefore, in case of disturbances (accidents), the intervention of the central plan (a central planning algorithm) led to better system performance than the decentralised solution in which agents (vehicles) have to plan locally their trajectory.

On the other hand, when the disturbance strength increases, the gain of the system increases. This confirms the conclusion that the intervention of the central plan was better demonstrating an improvement of the system throughput.

Therefore, it is inferred that a global problem (e.g., an accident in the intersection) should be solved at global level, because there is a central unit (the O/C architecture) that has the global view of the system. This central unit can plan better than a decentral unit. A central unit needs only longer time than a decentral unit. This issue can be solved simply by providing central units that have sufficient resources, e.g., CPU capacity (real-time requirements), memory capacity, etc, as well as the management of these resources.

VI. CONCLUSION

In this paper, we extended the implementation of the generic O/C architecture adapted to our traffic scenario and accomplished our experiments assuming that accidents (disturbances), in addition to deviations from plan, occur in the system environment (intersection).

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Additionally, we introduced an interdisciplinary methodology called "Robust Multi-Agent System" (RobustMAS). We developed and evaluated RobustMAS aiming to keep a multi-agent system at a desired performance level when disturbances (accidents, unplanned autonomous behaviour) occur. RobustMAS represents a robust hybrid central/self-organising multi-agent system, in which the conflict between centralised interventions (central planning) and the autonomy of the agents (decentralised mechanisms, autonomous vehicles) was solved.

In this regard, we measured the system performance and compared the two cases, the system performance with disturbances on one side and the system performance without disturbances from the other side. This comparison showed that the system performance remains effective (robust) despite disturbances and deviations occurred in the system. Furthermore, we discussed two quantitative approaches introduced in the literature to quantify robustness. Afterwards, we presented an appropriate metric for the quantitative determination of the robustness of such hybrid multi-agent systems. Subsequently, we measured the robustness and gain of a multi-agent system using the RobustMAS concept. The experiments showed a high degree of the robustness of RobustMAS.

VII. Future Work

One aspect that may be of interest for future work is the fairness between the system's agents (vehicles). In order to achieve this fairness, there are different approaches that deal with this issue. The other aspect that will be an important issue in future is the coordination and cooperation of multiple intersections without traffic lights. Finally, since the RobustMAS concept is applicable for other systems, this paper leaves space for the applicability of the RobustMAS concept for shared spaces. The current traffic scenario used in this work has similarities to shared spaces in the working environments and conditions, where vehicles move autonomously in a shared environment.

References

[1] Y. Chaaban, J. Hähner, and C. Müller-Schloer. "Measuring Robustness in Hybrid Central/Self-Organising Multi-Agent Systems". In Cognitive12: proceedings of the Fourth International Conference on Advanced Cognitive Technologies & Applications, July 2012, pp. 133-138, Nice, France.

[2] CAS-wiki: Organic Computing. http://wiki.cas-group.net/index.php?title=Organic_Computing, [retrieved: June, 2013].

[3] C. Müller-Schloer, H. Schmeck, and T. Ungerer. "Organic Computing — A Paradigm Shift for Complex Systems". Birkhäuser, Verlag 2011.

[4] J.O. Kephart and D.M. Chess. "The vision of autonomic computing". IEEE Comput. 1, 2003, pp. 41-50.

[5] R. Sterritt. "Autonomic Computing". Innov. Syst. Softw. Eng. 1(1), 2005, pp. 79-88.

[6] M. Zeller. Fraunhofer Institute for Communication Systems ESK, Germany. IARIA Work Group Meeting: Autonomic and Autonomous, 1. PANEL ICAS, Topic: Robustness and Trust in Autonomic Systems. Panel @ ICAS 2010. The Sixth International Conference on Autonomic and Autonomous Systems ICAS 2010, March 7-13, 2010 - Cancun, Mexico. http://www.iaria.org/conferences2010/filesICAS10/ICAS_2010_Panel.pdf

[7] J.P. Steghöfer, R. Kiefhaber, K. Leichtenstern, Y. Bernard, L. Klejnowski, W. Reif, T. Ungerer, E. André, J. Hähner, and C. Müller-Schloer, "Trustworthy Organic Computing Systems: Challenges and Perspectives", Proceedings of the 7th International Conference on Autonomic and Trusted Computing (ATC 2010), Springer.

[8] Y. Uny Cao, Alex S. Fukunaga, and Andrew B. Kahng. "Cooperative Mobile Robotics: Antecedents and Directions". Autonomous Robots, 1997, pp. 4:226-234.

[9] C. Müller-Schloer. "Organic computing: on the feasibility of controlled emergence". In CODES+ISSS '04: Proceedings of the 2nd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis, 2004, pp. 2-5. ACM.

[10] U. Richter, M. Mnif, J. Branke, C. Müller-Schloer, and H. Schmeck. "Towards a generic observer/controller architecture for organic computing". In Christian Hochberger and Rüdiger Liskowsky, editors, INFORMATIK 2006 - Informatik für Menschen!, volume P-93 of GI-Edition - Lecture Notes in Informatics (LNI), 2006, pp. 112-119. Bonner Köllen Verlag.

[11] Y. Chaaban, J. Hähner, and C. Müller-Schloer. "Towards fault-tolerant robust self-organizing multi-agent systems in intersections without traffic lights". In Cognitive09: proceedings of The First International Conference on Advanced Cognitive Technologies and Applications, November 2009, pp. 467-475, Greece. IEEE.

[12] Y. Chaaban, J. Hähner, and C. Müller-Schloer. "Towards Robust Hybrid Central/Self-organizing Multi-agent Systems". In ICAART2010: proceedings of the Second International Conference on Agents and Artificial Intelligence, Volume 2, January 2010, pp. 341-346, Valencia, Spain.

[13] Y. Chaaban and C. Müller-Schloer. "Specifying Desired Behaviour in Hybrid Central/Self-Organising Multi-Agent Systems". In Cognitive13: proceedings of the Fifth International Conference on Advanced Cognitive Technologies and Applications, May 27 - June 1, 2013, pp. 1-6, Valencia, Spain.

[14] Y. Chaaban, J. Hähner, and C. Müller-Schloer. "Handling of Deviations from Desired Behaviour in Hybrid Central/Self-Organising Multi-Agent Systems". In Cognitive12: proceedings of the Fourth International Conference on Advanced Cognitive Technologies and Applications, July 2012, pp. 122-128, Nice, France.

[15] Y. Chaaban and C. Müller-Schloer. "A Survey of Robustness in Multi-Agent Systems". In Cognitive13: proceedings of the Fifth International Conference on Advanced Cognitive Technologies and Applications, May 27 - June 1, 2013, pp. 7-13, Valencia, Spain.

[16] K. Dresner and P. Stone. "Mitigating catastrophic failure at intersections of autonomous vehicles". In AAMAS '08: Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems, pp. 1393-1396, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.

[17] H. Schmeck, C. Müller-Schloer, E. Cakar, M. Mnif, and U. Richter. "Adaptivity and Self-organisation in Organic Computing Systems". ACM Transactions on Autonomous and Adaptive Systems, 2009, pp. 10:1-10:32.

[18] V. Shestak, H. J. Siegel, A. A. Maciejewski, and S. Ali. "The robustness of resource allocations in parallel and distributed computing systems". In Proceedings of the International Conference on Architecture of Computing Systems (ARCS 2006), pp. 17-30.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[19] D. England, J. Weissman, and J. Sadagopan. “A new metric for robustness with application to job scheduling”. In IEEE International Symposium on High Performance Distributed Computing 2005 (HPDC-14), Research Triangle Park, NC, July 2005, pp. 24-27.

[20] K. Waldschmidt and M. Damm. “Robustness in SOC Design”, Digital System Design: Architectures, Methods and Tools, 2006. DSD 2006. 9th EUROMICRO Conference on Digital System Design, pp. 27-36, Volume: Issue: , 0-0.

[21] W. Heupke, C. Grimm, and K. Waldschmidt. “Modeling uncertainty in nonlinear analog systems with affine arithmetic”. Advances in Specification and Design Languages for SOCs Selected Contributions from FDL ’05, 2005.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Optimization and Evaluation of Bandwidth-Efficient Visualization for Mobile Devices

Andreas Helfrich-Schkarbanenko, Roman Reiner, Sebastian Ritterbusch, and Vincent Heuveline
Engineering Mathematics and Computing Lab (EMCL)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
{*andreas.helfrich-schkarbanenko, roman.reiner, sebastian.ritterbusch, vincent.heuveline*}*@kit.edu*

***Abstract*—The visual analysis of large numerical simulations on mobile devices needs remote parallelizable visualization methods for low-bandwidth and high-latency networks. Based on a mathematical model for multi-layered planar impostor representation of arbitrary complex and unbounded scenes, we adapt an algorithm for optimal viewport placement from the theory of optimal experimental design. The results are evaluated in a realistic setting, proving the practical relevance of the theoretical findings, leading to a bandwidth-efficient remote visualization concept for high performance computing simulation results.**

Keywords-Remote Visualization, Mobile Visualization, Optimal Experimental Design, Bandwidth Efficiency.

I. Introduction

Remote visualization is vital wherever local storage, data transfer rates or graphical capabilities are limited. Even though the capabilities of modern smartphones are increasing rapidly, without efficient visualization methods as introduced in [1] many desirable applications are impeded by limitations of the current hardware [2].

Image-based rendering techniques [3] are widely used to reduce the geometric complexity of virtual environments by replacing parts of a scene with a textured representation approximating the original geometry. Since these so-called *impostors* have a significantly simplified geometry, parallax errors [4] occur when rendering the approximation. An impostor is generated for an initial *viewport* (that is, a position and viewing direction) and is said to be *valid* as long as the visual difference to the (hypothetically rendered) original geometry is below a certain threshold.

In our application, these impostors are rendered remotely on render servers and streamed to a mobile device where they are used to approximate the scene. One substantial advantage of the impostor approach [5] is that the render time on the device only depends on the number of impostors and the resolution of the textures, not on the amount of data they display. As long as servers can generate and transfer the impostor textures sufficiently fast, every scene can be displayed remotely, regardless of its actual complexity. In this setting, network bandwidth is the bottleneck and a careful analysis of bandwidth consumption becomes mandatory.

We develop a mathematical model that allows us to quantify the display error and propose an approximation method that proves to be optimal with respect to the derived error metric. We can show that our method significantly reduces the total amount of image data that needs to be transferred. The key aspects of our method are illustrated in Figure 1: In this simplified two-dimensional case, a traditional remote visualization using one layer would need at least 32 images to provide the same visual accuracy as one layer set of 5 images. This effect is amplified by each additional degree of freedom of the viewer. Based on the error metric that was already presented in [1], this paper extends the method described in [6] with respect to optimally chosen viewport sets locations for fixed numbers of layers, and evaluates the realistic performance of the concepts.

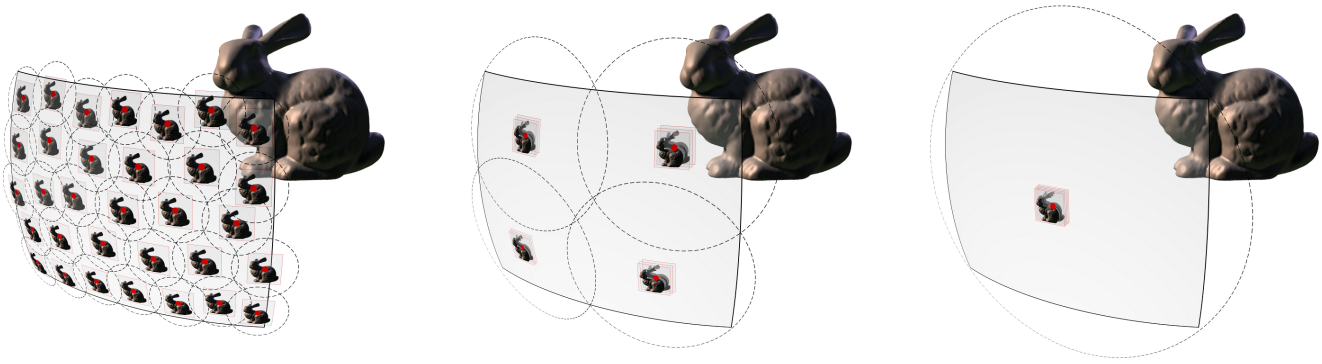
In the following Section II, we discuss related work. Then we introduce the underlying mathematical model in Section III, on which we derive the fundamental error metrics. In Section IV, this leads us to the optimal impostor placement and directly corresponding bounds for the visualization error of one impostor set. The practical outcome of the findings, using as many impostor sets as needed, is proven and evaluated theoretically in Section V. The general placement of viewports for impostor sets is solved by adaption of an algorithm from optimal experimental design to the visualization problem in Section VI. The proposed method is evaluated in Section VII in a realistic setting, which leads us to the conclusions in Section VIII.

II. Related work

A variety of image-based rendering techniques are reviewed in [5] and [3]. The first paper focuses mainly on techniques using planar impostors but also mentions more exotic approaches like depth images (planar impostors with per-pixel depth information) and light fields. These and other techniques, such as view morphing and view dependent textures, are examined in more detail in the second paper.

In the majority of cases, planar impostors stacked with increasing distance to the observer are used (see [4], [7],

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



(a) 32 impostor sets with one layer each (b) Four impostor sets with three layers each (c) One impostor set with five layers

Figure 1. An impostor representation is only valid inside a small region around the initial viewport for which it was originally created. For observer viewports within this validity region (indicated by the dotted line) the display error does not exceed a given maximum value. To faithfully approximate the scene for all observer viewports inside the shaded area, several impostor sets have to be transmitted.
The validity regions can be enlarged (while keeping the maximum error unaltered) by increasing the number of layers per impostor set. As the number of required impostor sets decreases faster than the number of layers per set increases, this significantly reduces the total number of layers needed to approximate the scene to a given accuracy .

[8]), usually to approximate distant parts of the scene or single objects. In contrast, our approach uses impostors to represent the full scene.

For large objects, different parts of continuous surfaces can end up on different impostors which makes them tear apart when viewed from a shallow angle. Avoiding this particular problem was one focus of the method developed in [4]. Another interesting use of planar impostors is [9], which treats the rendering of volume data on mobile phones.

Several approaches using geometrically more complex impostors can be found in [8], [10] and [11]. In [5], so-called *billboard clouds* are used to approximate the shape of an object using several intersecting planar impostors. While the impostor creation process for this approach is quite costly, the result allows examination from different viewing directions.

A very current example is Street Slide [12]. Street Slide sticks photos of front facades of urban environments to "panorama strips" that can be browsed by sliding sideways.

The need for accurate analysis of bandwidth and accuracy estimates is discussed in [5], [7], without further specifying how to choose which viewports to load. A more in-depth analysis on the subject of pre-fetching is given in [13] and [14]. The former defines a so-called benefit integral, indicating which parts of the scene – quality-wise – contribute most to the final image, the latter deals with rendering an indoor scene remotely. The task of remote rendering on mobile devices is addressed in [15] and [16], which mostly focuses on the technical aspects of the server-client communication.

Usually, depending on the complexity of the approximation, an impostor is either easy to generate but only valid inside a small region and thus needs to be updated very often, or it is valid inside a large domain but complex and difficult to generate and display [3]. Since the former strains bandwidth and the latter strains render speed, any image-based rendering approach is usually a trade-off between these limiting factors.

III. Visualization model and error metrics

To begin with, a mathematical model describing viewports and projections thereon needs to be established, with which the rendering and approximation processes can be described. This yields an error function describing the maximum parallax error of a scene as a function of the observer movement, called *domain error*.

Finally, modeling the observer movement as a probability distribution, we can describe the expected value of this error. This *interaction error* will be the cost function that we intend to minimize.

A. Perspective projection

Using homogeneous coordinates and projective transformations [17], we can express perspective projection as a 4×4 matrix multiplication on the projective space $\mathbb{P}^3$:

Definition 1. The perspective projection onto the plane $x_3 = d$ towards the origin is a function

$$\pi_d : \begin{cases} \mathbb{P}^3 \backslash \{(0,0,0,1)^\top\} & \longrightarrow & \mathbb{P}^3 \\ x & \longmapsto & P_d x \end{cases}$$

with the parameter $d > 0$ defining the proximity of the projection plane.

From the intercept theorems, one can easily see that the perspective projection of a point $v = (v_1, v_2, v_3)^\top \in \mathbb{R}^3$, $v_3 \neq 0$ onto the plane $x_3 = d$ is given by

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$(\frac{d}{v_3}v_1, \frac{d}{v_3}v_2, d)^\top$ which, using homogeneous coordinates, equals $(v_1, v_2, v_3, \frac{v_3}{d})^\top$. This yields the projection matrix

$$P_d := \left(\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 1/d & 0 \end{array}\right).$$

B. Viewports

Any viewport can be described by five values $c_1, c_2, c_3 \in \mathbb{R}$, $\vartheta \in [-\pi/2, \pi/2]$, $\varphi \in [-\pi, \pi)$, defining an affine transformation χ, which is the combination of a translation by the vector $(c_1, c_2, c_3)^\top$ followed by a rotation around the x_1-axis with the angle ϑ and a rotation around the x_2-axis with the angle φ (cf. Figure 2). Actually, there is a sixth value which represents a rotation around the viewing direction. Such a rotation, however, does not change the image besides rotating it. We assume the rotation to be lossless, which is why we do not need it for our purposes.

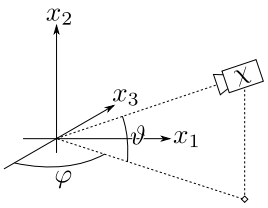


Figure 2. The angles φ and ϑ of a viewport χ

We condense all five values into a single vector $c := (c_1, c_2, c_3, \vartheta, \varphi)^\top$. When describing viewports, we will use this vector c and the associated transformation χ_c interchangeably. In particular, we will identify sets of viewports with subsets of $\mathbb{R}^5$:

Definition 2. The set

$$X := \mathbb{R}^3 \times [-\pi/2, \pi/2] \times [-\pi, \pi) \subset \mathbb{R}^5$$

will be called the *viewport set*. For all practical purposes, however, we want to restrict to viewports inside a given set of *feasible viewports* $\Lambda \subset X$.

Projective matrix representations of χ_c and its inverse are

$$Q_c = \left(\begin{array}{c|c} B_{\vartheta,\varphi} & B_{\vartheta,\varphi}c \\ \hline 0 & 1 \end{array}\right) \quad \text{and} \quad Q_c^{-1} = \left(\begin{array}{c|c} B_{\vartheta,\varphi}^\top & -c \\ \hline 0 & 1 \end{array}\right)$$

where

$$B_{\vartheta,\varphi} := \begin{pmatrix} \cos\varphi & -\sin\varphi\sin\vartheta & -\sin\varphi\cos\vartheta \\ 0 & \cos\vartheta & -\sin\vartheta \\ \sin\varphi & \cos\varphi\sin\vartheta & \cos\varphi\cos\vartheta \end{pmatrix}.$$

We can now calculate a matrix representation of a projection onto an arbitrary viewport, by combining the matrices above with the matrix representations of the default projection π_d.

Definition 3. Let χ be a viewport with an associated matrix representation Q and let π_χ denote a projection onto the viewport χ. Then, a matrix representation of π_χ is given by $P_{\chi,d} = QP_dQ^{-1}$, where P_d is the perspective projection matrix defined in Definition 1.

C. Rendering process

Let renderable objects be located in a domain Ω. We aim to simplify the scene by dividing Ω into m disjoint parts Ω_i called *cells*, replacing each with a planar representation of their contained objects. These so-called *impostors* will be created for the same initial viewport(s), that is, for a certain viewport we will create an *impostor set* with one impostor per cell, all for that particular viewport. This will be done for n initial viewports resulting in n impostor sets with m impostors each.

As long as the current viewport matches the initial viewport for which the impostors have been created, the impostor representation coincides with the image of the actual scene. Changing the viewport, however, will introduce parallax errors, since depth information is lost in the impostor creation process.

To determine this error, we will first regard a single cell Ω_i and a single vertex $v \in \Omega_i$. For a fixed initial viewport χ_1 we calculate the impostor representation $\overline{v}$ of the actual point v. Then we consider a variable viewport χ and calculate the screen coordinates v' of v and $\overline{v}'$ of $\overline{v}$ as functions of the viewports χ and χ_1 (cf. Figure 3).

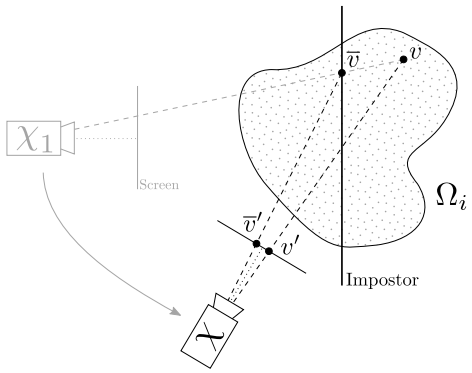


Figure 3. Rendering process for changed viewport

D. The domain error

If we reiterate the procedure above, we obtain two images for each point in Ω: one image of itself (v', depending on χ) and one of its impostor representation ($\overline{v}'$, depending on both χ and χ_1). The screen distance of these two, measured in (sub-)pixels is called the *screen space error*. As we are not interested in the error of a single point, but rather in error functions expressing the error of the entire scene, for example the mean error or the maximum error, we aggregate the screen space error over all points in Ω. As the distribution

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

of vertices inside Ω is supposed to be unknown, we assume a uniform distribution and integrate the screen space error over the entire domain Ω. We will be using the maximum error which replaces the integral with a supremum.

Definition 4. Denote the number of cells with m. For an initial viewport χ_1 we define the *domain error*

$$\begin{aligned} \mathrm{D}(\chi, \chi_1) &:= \sup_{v \in \Omega} \left\| v'(\chi) - \bar{v}'(\chi, \chi_1) \right\|_2 \\ &= \max_{0 \le i \le m} \Big\{ \sup_{v \in \Omega_i} \left\| v'(\chi) - \bar{v}'(\chi, \chi_1) \right\|_2 \Big\}. \end{aligned}$$

This domain error depends on a variable observer viewport χ and the fixed viewport χ_1, for which the displayed impostor set was initially created. The dependence on χ implies that we cannot evaluate our impostor approximation without knowledge of the observer movement. Clearly, we want to optimize our setup a priori, and hence we need to find a way to evaluate it without knowledge of χ.

E. The interaction error

Assume that we have n impostor sets at hand for viewports $\chi_1, \ldots, \chi_n \in \Lambda \subset X$. As before, we denote the observer's viewport with $\chi \in \Lambda$. Since we can choose from several impostor sets, we display that set whose initial viewport χ_k satisfies

$$\mathrm{D}(\chi, \chi_k) = \min_{1 \le j \le n} \mathrm{D}(\chi, \chi_j).$$

For $1 \le k \le n$ let Ξ_k denote that subset of Λ, on which $\mathrm{D}(\chi, \chi_k)$ is the smallest of all domain errors:

$$\Xi_k := \big\{ \chi \in \Lambda \,\big|\, \mathrm{D}(\chi, \chi_k) = \min_{1 \le j \le n} \mathrm{D}(\chi, \chi_j) \big\}. \quad (1)$$

Next, we define a probability distribution P with an associated probability density function μ on Λ, for instance, a uniform distribution over Λ or a normal distribution around the current viewport χ. These distributions represent the probability for the respective viewport to occur, thus modeling the expected observer movement. We can then calculate the expected value of the error by integrating the domain error D over Λ with respect to the probability distribution P.

Definition 5. Let $n \ge 1$. We define the *interaction error* $\mathrm{I} : \Lambda^n \to \mathbb{R}$, where

$$\begin{aligned} \mathrm{I}(\chi_1, \ldots, \chi_n) &:= \int_\Lambda \min_{1 \le j \le n} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi) \quad (2) \\ &= \sum_{j=1}^{n} \int_{\Xi_j} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi). \end{aligned}$$

The following Lemma shows that the interaction error will decrease as we add more viewports.

Lemma 1. Let $\chi_1, \ldots, \chi_n \in \Lambda$. Then

$$\mathrm{I}(\chi_1) \ge \mathrm{I}(\chi_1, \chi_2) \ge \cdots \ge \mathrm{I}(\chi_1, \ldots, \chi_n).$$

Proof: For $1 \le k \le n$, it is

$$\begin{aligned} \mathrm{I}(\chi_1, \ldots, \chi_k) &= \int_\Lambda \min_{1 \le j \le k} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi) \\ &\le \int_\Lambda \min_{1 \le j \le k-1} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi) \\ &= \mathrm{I}(\chi_1, \ldots, \chi_{k-1}). \end{aligned}$$

■

IV. Impostor placement and error bounds

The efficiency of the proposed method is based on an optimal choice of initial viewports for the impostor sets, as well as an optimized cell partition for each set.

Theorem 2. Given renderable objects located in

$$\Omega := \big\{ (x_1, x_2, x_3, 1)^\top \in \mathbb{P}^3 \,\big|\, 0 < a_0 < x_3 < a_{m+1} \le \infty \big\},$$

the optimal cell boundaries for viewport translations are given by $a_i = (1/a_0 - i\delta)^{-1}$, $i = 1, \ldots, m$ for a suitable $\delta(m) > 0$, and the optimal impostor placement with respect to the error metric is

$$d_i = \frac{2 a_i a_{i+1}}{a_i + a_{i+1}}.$$

Note that m is finite even for domains with infinite depth, that is, when $a_{m+1} = \infty$ for which $d_m = 2a_m$.

Proof: For viewport translations the minimum of the domain error D with respect to the projection plane distance $d \in [a, b]$ can be found analytically. For details see [18, Theorem 3.2]. ■

With this impostor placement, we have the following asymptotic behavior of the error with respect to viewport translations:

Theorem 3. For a fixed maximal screen space error $\varepsilon > 0$, the radius r of maximal permissible viewport change is proportional to the number of impostors per set m.

Proof: This property emerges during the proof of Theorem 2. For details see [18, Remark 3.5]. ■

This Theorem shows that increasing the number of impostors per set will strongly decrease the interaction error, but the number of displayable impostors is bounded by the graphical capabilities of mobile devices. Due to such limitations, several impostors sets have to be transmitted.

Denote the number of impostor sets with n. Under certain assumptions we can show that the inspection error can be bounded by

$$C_1 n^{-1/5} \le \mathrm{I}(\chi_1, \ldots, \chi_n) \le C_2 n^{-1/5},$$

for constants $C_{1/2} = C_{1/2}(\Lambda, m)$. Proving these bounds will be the endeavor of the next section.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

V. Model evaluation

Proposition 1. Using the $\mathbb{R}^5$-parametrization of the viewport space, we can regard the domain error $\mathrm{D}(\chi, \chi_k)$ as a continuous function $f : \mathbb{R}^5 \times \mathbb{R}^5 \to \mathbb{R}$ which, for moderate viewport changes, behaves almost linear.

More precisely, we can find positive constants $a_1, \dots a_5$ and $\bar{a}_1, \dots, \bar{a}_5$ such that

$$\|A_1(x-y)\| \le f(x,y) \le \|A_2(x-y)\| \tag{3}$$

where $A_1 := \mathrm{diag}(a_1, \dots, a_5)$ and $A_2 := \mathrm{diag}(\bar{a}_1, \dots, \bar{a}_5)$.

Proposition 2. The matrices A_1 and A_2 depend on the number of cells m. For viewport translations they are proportional to m^{-1} as a direct consequence of Theorem 3.

Before proceeding, we need the following Lemmata.

Remark 1. In the following $A = B+C$ means that the set A is the direct sum of the sets B and C, that is, $A = B \cup C$ and $B \cap C = \emptyset$. In particular, $\mathrm{vol}\,(A+B) = \mathrm{vol}\,(A) + \mathrm{vol}\,(B)$.

Similarly, $A = B - C$ means that $B = A + C$, that is, $C \subset B$ and $\mathrm{vol}\,(B-C) = \mathrm{vol}\,(B) - \mathrm{vol}\,(C)$.

Lemma 4. Let G be a bounded, measurable, d-dimensional subset of $\mathbb{R}^d$ and let B be a d-dimensional ball (with respect to a norm $\|\cdot\|$) of equal volume (cf. Figure 4a). Then

$$\int_G \|x\| \,\mathrm{d}x \;\ge\; \int_B \|x\| \,\mathrm{d}x.$$

Proof: Denote the radius of B with R. Due to $G = G \cap B + G \backslash B$ and $B = G \cap B + B \backslash G$, we can express G as $G = (B - B\backslash G) + G\backslash B$. As the volumes of G and B are equal, this also implies $\mathrm{vol}\,(G\backslash B) = \mathrm{vol}\,(B\backslash G)$.

Moreover, the distance from the origin to all points in $G\backslash B$ is larger than R while for all points in $B\backslash G$ it is smaller. Hence,

$$\int_{G\backslash B} \|x\| \,\mathrm{d}x \ge \int_{G\backslash B} R \,\mathrm{d}x = R\,\mathrm{vol}\,(G\backslash B)$$

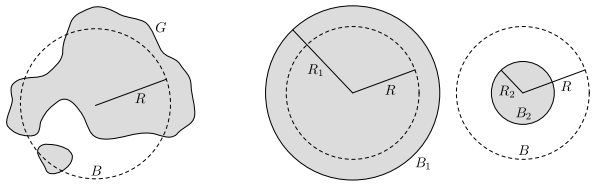
and, conversely,

$$\int_{B\backslash G} \|x\| \,\mathrm{d}x \le \int_{B\backslash G} R \,\mathrm{d}x = R\,\mathrm{vol}\,(B\backslash G)\,.$$

This implies

$$\begin{aligned}\int_G \|x\| \,\mathrm{d}x &= \int_B \|x\| \,\mathrm{d}x - \int_{B\backslash G} \|x\| \,\mathrm{d}x + \int_{G\backslash B} \|x\| \,\mathrm{d}x \\ &\ge \int_B \|x\| \,\mathrm{d}x - R\underbrace{\big(\mathrm{vol}\,(B\backslash G) - \mathrm{vol}\,(G\backslash B)\big)}_{=0}.\end{aligned}$$

■



(a) Lemma 4. (b) Lemma 5.

Figure 4. Accompanying illustrations for the lemmata.

Lemma 5. Let B and $B_1, \dots, B_n$ be d-dimensional balls (with respect to a norm $\|\cdot\|$), such that the volume of B is the arithmetic mean of the volumes of $B_1, \dots, B_n$. Then

$$\sum_{k=1}^{n} \int_{B_k} \|x\| \,\mathrm{d}x \;\ge\; n \int_B \|x\| \,\mathrm{d}x.$$

Proof: We first regard the case $n = 2$. Without loss of generality, let $R_1 \ge R \ge R_2$.

We define $G := (B_1 - B) + B_2$. Then, $\mathrm{vol}\,(G) = \mathrm{vol}\,(B_1) - \mathrm{vol}\,(B) + \mathrm{vol}\,(B_2) = \mathrm{vol}\,(B)$ and Lemma 4 yields

$$\begin{aligned}\int_B \|x\| \,\mathrm{d}x &\le \int_G \|x\| \,\mathrm{d}x \\ &= \int_{B_1} \|x\| \,\mathrm{d}x - \int_B \|x\| \,\mathrm{d}x + \int_{B_2} \|x\| \,\mathrm{d}x.\end{aligned}$$

From this, the general case follows by induction. ■

Lemma 6. Let B be a 5-dimensional ball with radius R. Then

$$\int_B \|x\|_2 \,\mathrm{d}x \;=\; \frac{4}{9}\pi^2 R^6.$$

Proof: Straightforward calculation using 5-dimensional polar coordinates. ■

With these Lemmata, we can prove the following estimation of the inspection error:

Theorem 7. Let Λ be bounded and assume a uniform distribution of observer viewports. Then, the interaction error can be bounded from below by

$$\mathrm{I}(\chi_1, \dots, \chi_n) \;\ge\; C_1 n^{-1/5},$$

with the constant

$$C_1 := \frac{5}{6}\left(\frac{15}{8\pi^2}\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{1/5},$$

where $A_1 := \mathrm{diag}(a_1, \dots, a_5)$ with constants $a_i > 0$ as in Proposition 1.

Proof: Let us first recall (1) and (2). Assuming a uniform distribution $\mu(\chi) = \mathrm{vol}\,(\Lambda)^{-1}$ we can rewrite (2) as

$$\mathrm{I}(\chi_1, \dots, \chi_n) = \mathrm{vol}\,(\Lambda)^{-1} \sum_{k=1}^{n} \int_{\Xi_k} \mathrm{D}(\chi, \chi_k) \,\mathrm{d}\chi. \tag{4}$$

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

On the right-hand side, we have to evaluate n integrals of the form $\int_G f(x,y)\,dx$. Using (3) we define a transformation of coordinates $\Phi(x) := A_1(x-y)$ (which is the same for all n integrals) and obtain

$$\int_G f(x,y)\,dx \;\geq\; \int_G \|\Phi(x)\|\,\mathrm{d}x \;=\; \frac{1}{\det(A_1)}\int_{\Phi(G)} \|x\|\,\mathrm{d}x.$$

Applying this to (4) yields

$$\mathrm{I}(\chi_1,\ldots,\chi_n) \geq (\det(A_1)\mathrm{vol}\,(\Lambda))^{-1}\sum_{k=1}^{n}\int_{\Phi_k(\Xi_k)}\|x\|\,\mathrm{d}x. \tag{5}$$

Using Lemmata 4 and 5 (with $d=5$), we obtain

$$\sum_{k=1}^{n}\int_{\Phi_k(\Xi_k)}\|x\|\,\mathrm{d}x \geq \sum_{k=1}^{n}\int_{B_k}\|x\|\,\mathrm{d}x \geq n\int_B\|x\|\,\mathrm{d}x,$$

where

$$\begin{aligned}\mathrm{vol}\,(B) &= \frac{1}{n}\sum_{k=1}^{n}\mathrm{vol}\,(B_k) = \frac{1}{n}\sum_{k=1}^{n}\mathrm{vol}\,(\Phi_k(\Xi_k))\\ &= \frac{1}{n}\det(A_1)\mathrm{vol}\,(\Lambda)\,.\end{aligned} \tag{6}$$

With this, the estimation (5) yields

$$\mathrm{I}(\chi_1,\ldots,\chi_n) \geq (\det(A_1)\mathrm{vol}\,(\Lambda))^{-1}\,n\int_B\|x\|\,\mathrm{d}x \tag{7}$$

Now, we choose to use the Euclidean norm $\|\cdot\| = \|\cdot\|_2$ for which a 5-dimensional ball with radius R has the volume $\mathrm{vol}\,(B) = \frac{8}{15}\pi^2R^5$. Then, (6) implies

$$R = \left(\frac{15}{8n\pi^2}\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{1/5}.$$

Hence, using Lemma 6,

$$\int_B\|x\|\,\mathrm{d}x = \frac{5}{6n}\det(A_1)\mathrm{vol}\,(\Lambda)\left(\frac{15}{8n\pi^2}\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{1/5}.$$

Inserting this into (7) we finally obtain

$$\mathrm{I}(\chi_1,\ldots,\chi_n) \;\geq\; \frac{5}{6}\left(\frac{15}{8n\pi^2}\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{1/5}.$$

■

This theorem shows, that the efficiency of any choice of impostor sets cannot be better than the given estimate. The following theorem constructively proves, that a choice of impostor sets with the desired asymptotic dependence exists, that is, that this estimate is actually achievable.

Theorem 8. Let Λ be bounded with a uniform distribution and let $\tilde{\Lambda}\supset\Lambda$ be an enclosing cuboid. Then, there is a set of viewports $\chi_1,\ldots\chi_n$ for which the interaction error satisfies

$$\mathrm{I}(\chi_1,\ldots,\chi_n) \;\leq\; C_2 n^{-1/5},$$

with the constant

$$C_2 := \frac{\pi^2}{36}\frac{(\max\{\bar{a}_1,\ldots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)},$$

where $A_2 := \mathrm{diag}(\bar{a}_1,\ldots,\bar{a}_5)$ with constants $\bar{a}_i > 0$ as in Proposition 1.

Proof: To begin with, we will prove the assertion for those n which are the fifth power of a whole number, that is, for $n^{1/5}\in\mathbb{N}$. The general case will be derived from this case later.

First, a bounded set Λ can be embedded into a cuboid $\tilde{\Lambda}$. For an n chosen as above, there is a regular decomposition of $\tilde{\Lambda}$ into five-dimensional cuboids Ξ_k with initial viewports χ_k at their respective centers.

Using the estimation $f(x,y)\leq\|A_2(x-y)\| = \|\Psi(x)\|$ with the same arguments as in the proof of Theorem 7, we obtain

$$\begin{aligned}\mathrm{I}(\chi_1,\ldots,\chi_n) &\leq \mathrm{vol}\,(\Lambda)^{-1}\sum_{k=1}^{n}\int_{\Xi_k}\mathrm{D}(\chi,\chi_k)\,\mathrm{d}\chi\\ &\leq (\det(A_2)\mathrm{vol}\,(\Lambda))^{-1}\sum_{k=1}^{n}\int_{\Psi_k(\Xi_k)}\|x\|\,\mathrm{d}x\\ &\leq (\det(A_2)\mathrm{vol}\,(\Lambda))^{-1}\,n\int_B\|x\|\,\mathrm{d}x,\end{aligned} \tag{8}$$

where we used that all cuboids $\Psi_k(\Xi_k)$ are identical and can be embedded into a ball B in the last step. For this the radius needs to be at least

$$R = \frac{1}{2}\mathrm{diam}(\Psi_k(\Xi_k)) \geq \max\{\bar{a}_1,\ldots,\bar{a}_5\}\frac{\mathrm{diam}(\tilde{\Lambda})}{2n^{1/5}}.$$

With this and Lemma 6 we finally obtain from (8)

$$\mathrm{I}(\chi_1,\ldots,\chi_n) \;\leq\; \frac{\pi^2}{72}\frac{(\max\{\bar{a}_1,\ldots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)}n^{-1/5}.$$

Now, for the general case, we divide $\tilde{\Lambda}$ into $\tilde{n} := \lfloor n^{1/5}\rfloor^5 \leq n$ cubes. This is possible because $\tilde{n}$ is the fifth power of a whole number ($\tilde{n}^{1/5}\in\mathbb{N}$). Moreover,

$$\frac{\tilde{n}^{-1/5}}{n^{-1/5}} = \frac{n^{1/5}}{\lfloor n^{1/5}\rfloor} \leq \frac{\lfloor n^{1/5}\rfloor+1}{\lfloor n^{1/5}\rfloor} = 1+\frac{1}{\lfloor n^{1/5}\rfloor} \leq 2,$$

that is, $\tilde{n}^{-1/5}\leq 2n^{-1/5}$. Hence, by this and Lemma (1)

$$\begin{aligned}\mathrm{I}(\chi_1,\ldots,\chi_n) &\leq I(\chi_1,\ldots,\chi_{\tilde{n}})\\ &\leq \frac{\pi^2}{72}\frac{(\max\{\bar{a}_1,\ldots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)}\tilde{n}^{-1/5}\\ &\leq \frac{\pi^2}{36}\frac{(\max\{\bar{a}_1,\ldots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)}n^{-1/5}.\end{aligned}$$

■

Remark 2. As stated earlier, the matrices A_1, A_2 depend on the number of cells m. With the assumptions in Proposition 2, it follows that $\mathrm{I} = \mathcal{O}(m^{-1}n^{-1/5})$.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

VI. Positioning Optimization

In general, the space decomposition into cuboids as utilized in Theorem 8 is far from being optimal. For a given number n of viewports, the optimal placement of viewports is a high dimensional optimization problem, similar to optimal experimental design (OED) problems.

OED provides techniques, that help to optimize the process of computing unknown parameters in experiments from measurements. The goal is to design the data collection process in such a way that the sensitivity of the measurements with respect to changes in the parameters is maximal, that is, the covariance of the measurement errors is to be minimized.

For the most part, this section follows the description in [19], mainly because of its conciseness. For more profound descriptions refer to (in order of extent) [20], [21], [22].

We assume a compact experimental region $\Omega \subset \mathbb{R}^d$ and denote the unknown parameters with $\theta = (\theta_1, \ldots, \theta_p)^\top \in \Theta \subseteq \mathbb{R}^p$. Let $y_\theta(x)$ denote the outcome of the experiment at the location $x \in \Omega$. At fixed locations $\xi = (\xi_1, \ldots, \xi_n)^\top$, $\xi_k \in \Omega$ we take measurements

$$z(\xi_k) = y_\theta(\xi_k) + \varepsilon_k \quad \text{for} \quad k = 1, 2, \ldots, n, \tag{9}$$

which are prone to measurement errors modeled as independently and identically distributed random variables ε_k with mean zero and variance σ^2.

To begin with, we assume a linearized model $y_\theta(\xi_k) = f(\xi_k)^\top \theta$ with a response function $f = (f_1, \ldots, f_p)^\top$, $f_k \in C(\Omega)$, which allows us to rewrite (9) in matrix notation as $Z = X\theta + \varepsilon$. where $Z = (z(\xi_1), \ldots, z(\xi_n))^\top \in \mathbb{R}^n$,

$$X = \begin{pmatrix} f_1(\xi_1) & \cdots & f_p(\xi_1) \\ \vdots & \ddots & \vdots \\ f_1(\xi_n) & \cdots & f_p(\xi_n) \end{pmatrix} \in \mathbb{R}^{n \times p},$$

and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^\top \in \mathbb{R}^n$. If $X^\top X$ is regular, we can compute the least squares estimate (e.g., [20, Th. 1.2.1]) $\hat{\theta} = (X^\top X)^{-1} X^\top Z$. Then, at a point $x \in \Omega$, the predicted response is

$$\hat{z}(x) := f(x)^\top \hat{\theta} \tag{10}$$

with covariance

$$\operatorname{cov}(\hat{z}(x)) = \sigma^2 f(x)^\top (X^\top X)^{-1} f(x) = f(x)^\top M^{-1} f(x), \tag{11}$$

where

$$M = M(\xi) := \frac{1}{\sigma^2} X^\top X. \tag{12}$$

The design problem is to find an *optimal design* ξ such that (10) is optimal in describing the actual experiment, that is, that $\operatorname{cov}(\hat{z})$ is *minimal*, often implemented for example in optimization of the determinant of M for D-optimality, or of the eigenvalues of M denoted as E-optimality.

Any design ξ can be regarded as a measure on Ω [19, p. 16]: Suppose we are taking n measurements at the locations $\xi_1, \ldots, \xi_n$. Then we can interpret ξ as a probability measure on Ω if we define

$$\xi(x) := \frac{1}{n} \sum_{k=1}^{n} \delta_{\xi_k}(x).$$

For such a design ξ, we can define M by

$$m_{i,j}(\xi) := \frac{1}{\sigma^2} \int_\Omega f_i(x) f_j(x) \, \mathrm{d}\xi$$

for $i, j = 1, \ldots, p$, $M(\xi) := (m_{i,j}(\xi))$. Similarly, the variance function (11) can be generalized to

$$d(x, \xi) := f(x)^\top M(\xi)^{-1} f(x).$$

An algorithm due to Wynn, Mitchell and Miller for a fixed number n-point design optimization for D-optimality reads [19, p. 20]:

1) Begin with an arbitrary n-point design $\xi^{(0)}(n)$.
2) Find ξ_{n+1} such that
$$d\big(\xi_{n+1}, \xi^{(j)}(n)\big) = \max_{x \in \Omega} \; d\big(x, \xi^{(j)}(n+1)\big)$$
and add ξ_{n+1} to the n-point design.
3) Find ξ_k such that
$$d\big(\xi_k, \xi^{(j)}(n+1)\big) = \min_{1 \le i \le n+1} d\big(\xi_i, \xi^{(j)}(n+1)\big)$$
and remove ξ_k from the $(n+1)$-point design.
4) Repeat steps 2 and 3 until the exchange does not result in an increase of $\det\big[M\big(\xi^{(j)}(n)\big)\big]$.

This algorithm optimizes an n-point design by repeating the two following steps:

- Add the point of minimum covariance to the n-point design.
- Remove the point of maximum covariance from the $(n+1)$-point design.

Note, that the covariances (and thereby the points added and removed in each step) depend on the current design.

This algorithm is adapted and extended to the viewport positioning optimization problem. Our goal is to find a viewport set $\{\chi_1, \ldots, \chi_n\} \subset \Lambda$ for a given $n \in \mathbb{N}$, which minimizes the inspection error $\mathrm{I}(\chi_1, \ldots, \chi_n)$. Theorem 1 states, that adding any viewport will cause a decrease of the inspection error, while removing any viewport with will increase it. Emulating the algorithm above, we start with an initial set of n viewports and hope to find an optimal viewport set by repeating the two following steps:

- Add that viewport for which the decrease of the inspection error is maximal.
- Remove that viewport for which the increase of the inspection error is minimal.

An implementation is given in

Algorithm 1

1. Begin with an arbitrary set of n viewports $\chi_1, \ldots, \chi_n$

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

2. **repeat**
3. $I_0 \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_n)$
4. $I_{\min} \leftarrow I_0$
5. **for** $\chi \in \Lambda$
6. **if** $\mathrm{I}(\chi_1, \ldots, \chi_n, \chi) < I_{\min}$
7. $\chi_{n+1} \leftarrow \chi$
8. $I_{\min} \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_n, \chi)$
9. Add χ_{n+1} to the set of viewports
10. $I_{\min} \leftarrow I_0$
11. **for** $i \leftarrow 1$ **to** $n+1$
12. **if** $\mathrm{I}(\chi_1, \ldots, \chi_{i-1}, \chi_{i+1}, \ldots, \chi_{n+1}) < I_{\min}$
13. $k \leftarrow i$
14. $I_{\min} \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_{k-1}, \chi_{k+1}, \ldots, \chi_{n+1})$
15. Remove χ_k from the set of viewports
16. $(\chi_1, \ldots, \chi_n) \leftarrow (\chi_1, \ldots, \chi_{k-1}, \chi_{k+1}, \ldots, \chi_{n+1})$
17. **until** $I_0 - I_{\min} < \kappa$

Lemma 9. Algorithm 1 is monotonically decreasing in I.

Proof: At the j-th iteration of the algorithm denote the set of starting viewports with X_j and the viewports added and removed in the intermediate steps with χ_{n+1} and χ_k respectively. Let further $X_{j+\frac{1}{2}} := X_j \cup \{\chi_{n+1}\}$ and $X_{j+1} := X_{j+\frac{1}{2}} \setminus \{\chi_k\}$. Then, $\mathrm{I}(X_{j+1}) = \mathrm{I}(X_{j+\frac{1}{2}} \setminus \{\chi_k\}) = \min_i [\mathrm{I}(X_{j+\frac{1}{2}} \setminus \{\chi_i\})] \leq \mathrm{I}(X_{j+\frac{1}{2}} \setminus \{\chi_{n+1}\}) = \mathrm{I}((X_j \cup \{\chi_{n+1}\}) \setminus \{\chi_{n+1}\}) = \mathrm{I}(X_j)$. ∎

We implement the algorithm for our well-understood reduced problem of parallel viewport translations. This has several advantages. Firstly, it is rather simple to implement; secondly, since the domain error does not need to be calculated by integration, it has moderate calculation times; and lastly, since Λ is only two-dimensional, the results can be easily displayed.

We observe that, rather than converging to a design with n points, the algorithm, after a while, cyclically generates subsets of cardinality n of one design with $n+1$ points (cf. Figure 5).

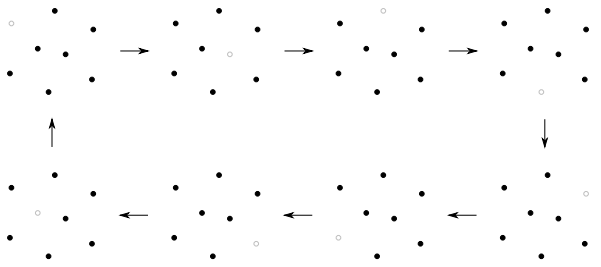


Figure 5. Cyclically generated subsets for $n = 7$

Further analysis shows, that at this point the first step of the algorithm always reproduces the same $(n+1)$-point design, from which in the second step one point is removed resulting in a subsets with n elements. That the points are removed cyclically is due to the fact, that in our implementation of the algorithm, from several points of equal weakness the first one is removed, while new elements are always added at the end. Since, in a way, the algorithm converges to an $(n+1)$-point design, we can eradicate that problem by simply switching the two steps of the algorithm. Then the algorithm converges to a design with n-points, and the subsets of cardinality $n-1$ are those generated between the steps. Once the optimal n-point design has been reached, the algorithm cyclically picks a point from the set, removes it from the design in the first step and immediately adds it again in the second step. Hence, the algorithm converges once n iterations in a row do not result in a different design.

With the two steps interchanged the algorithm does indeed converge to an n-point design as desired. However, especially for large n, it requires quite a lot of steps to turn the arbitrary initial design into a "reasonable" design which is then optimized further. Therefore, rather than starting with a random design, we hope to improve the algorithm by generating an initial design as follows.

Emulating the OED optimization algorithm by Federov, described in [19], we start with an empty design and successively add points, which are in some sense optimal, until we get to an n-point design. We do this by simply running the second step of the algorithm n times. This way, every point added to the initial design maximizes the decrease of the inspection error, resulting in an initial design which is rather good already. Note, that even though the point added in the k-th step in this manner is the optimal choice, the resulting k-point design is usually not optimal. For example, for $n = 3$ and a normal distribution, the first point will end up in the center and the second and third point will end up opposite of each other forming a line with the first point, while the optimal design would be three points forming an equilateral triangle around the center.

An updated version of our algorithm including the generation of an initial design is given below.

Algorithm 2

1. **for** $j \leftarrow 1$ **to** n
2. $I_{\min} \leftarrow \infty$
3. **for** $\chi \in \Lambda$
4. **if** $\mathrm{I}(\chi_1, \ldots, \chi_{j-1}, \chi) < I_{\min}$
5. $\chi_j \leftarrow \chi$
6. $I_{\min} \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_{j-1}, \chi)$
7. Add χ_j to the set of viewports
8. **repeat**
9. $I_0 \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_n)$
10. $I_{\min} \leftarrow I_0$
11. **for** $i \leftarrow 1$ **to** n
12. **if** $\mathrm{I}(\chi_1, \ldots, \chi_{i-1}, \chi_{i+1}, \ldots, \chi_n) < I_{\min}$
13. $k \leftarrow i$
14. $I_{\min} \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_{i-1}, \chi_{i+1}, \ldots, \chi_n)$

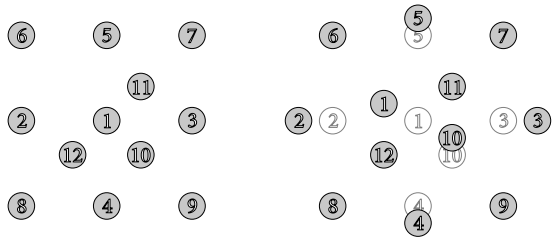
2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

15. Remove χ_k from the set of viewports
16. $(\chi_1, \ldots, \chi_{n-1}) \leftarrow (\chi_1, \ldots, \chi_{k-1}, \chi_{k+1}, \ldots, \chi_n)$
17. $I_{\min} \leftarrow I_0$
18. **for** $\chi \in \Lambda$
19. **if** $\mathrm{I}(\chi_1, \ldots, \chi_{n-1}, \chi) < I_{\min}$
20. $\chi_n \leftarrow \chi$
21. $I_{\min} \leftarrow \mathrm{I}(\chi_1, \ldots, \chi_{n-1}, \chi)$
22. Add χ_n to the set of viewports
23. **if** $I_0 = I_{\min}$
24. $m \leftarrow m + 1$
25. **else**
26. $m \leftarrow 0$
27. **until** $m = n$

Lemma 10. After step 7, Algorithm 2 is monotonically decreasing in I.

Proof: At the j-th iteration of the algorithm denote the set of starting viewports with X_j and the viewports added and removed in the intermediate steps with χ_{n+1} and χ_k respectively. Let further $X_{j+\frac{1}{2}} := X_j \setminus \{\chi_k\}$ and $X_{j+1} := X_{j+\frac{1}{2}} \cup \{\chi_{n+1}\}$. Then, $\mathrm{I}(X_{j+1}) = \mathrm{I}(X_{j+\frac{1}{2}} \cup \{\chi_{n+1}\}) = \min_{\chi \in \Lambda} \left[\mathrm{I}(X_{j+\frac{1}{2}} \cup \{\chi\}) \right] \leq \mathrm{I}(X_{j+\frac{1}{2}} \cup \{\chi_k\}) = \mathrm{I}((X_j \setminus \{\chi_k\}) \cup \{\chi_k\}) = \mathrm{I}(X_j)$. ■

The following Figure shows the resulting patterns of the initial and the optimal design. The numbering of the points reflects their order of appearance in phase 1, i.e., after step 7.



(a) Initial design after step 7. (b) Optimal design after step 27.

Figure 6. Testing Algorithm 2 for $n = 12$ and a normal distribution.

As long as the observer is not moving, pre-fetching the viewports as proposed by the algorithm results in an optimal set of data sets. However, in general, the optimal viewport distribution for any two different observer locations do not share any viewports. Hence, if we always pre-fetch those data sets which are optimal for the current observer location, we need to update *all* data sets whenever the observer is moving, rendering this approach useless. Instead, since the algorithm works by optimizing a given viewport distribution, it can be used adaptively. That is, rather than pre-fetching those data sets for the optimal distribution, we can update the current state, where all updates are in order of their importance. This way, even though the intermediate steps might not be optimal, every update is taking account of the data sets already available on the device. An optimal distribution is only attained, if the observer is standing still long enough for all data sets to be updated to their optimal viewport.

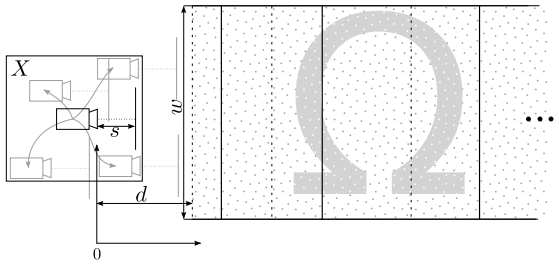
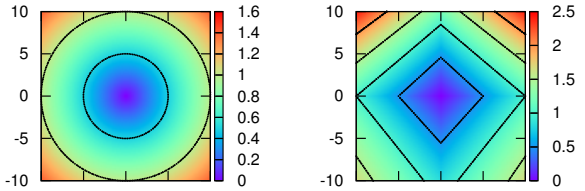


Figure 7. Test setup for evaluation



(a) Error in x_1, x_2 for $x_3 = 0$. (b) Error in x_1, x_3 for $x_2 = 0$.

Figure 8. Screen space error distribution in pixels over view position change in pixels for 5-layer viewport centered in X.

VII. Numerical Tests

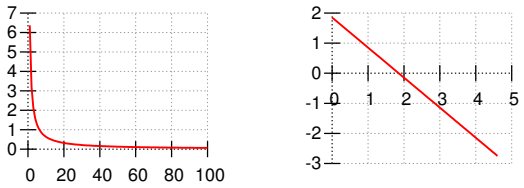
Since the positioning optimization is not trivial, we present the actual performance of the method in a test setup. In Figure 7, we see a half-infinite cylinder $\Omega \in \mathbb{R}^3$ with diameter w representing the visualization volume. The unmoved camera is placed in distance d from Ω and the visualization screen is fixed at distance s from the camera. When the camera is moving away from the center of viewport set X, the screen is moving with the camera, but Ω remains fixed.

In the following evaluation, the parameters $w = 100$, $d = s = 100$, and $X = [-10, 10]^3$ with uniform distribution were used. For illustration, the camera is not rotated, and thus we have a three dimensional viewport space and expect to achieve $\mathrm{I} = \mathcal{O}(m^{-1} n^{-1/3})$ in this setup.

First of all, Figure 8 presents the error distribution for one viewport with 5 layers in two cut-planes through X for camera motion $x_3 = 0$ and $x_2 = 0$. Moving the camera in constant distance yields a symmetric error distribution, as it can be seen in Figure 8 (a). Moving towards Ω results in higher errors than moving away in Figure 8 (b), due to the displayed size increase of nearer objects.

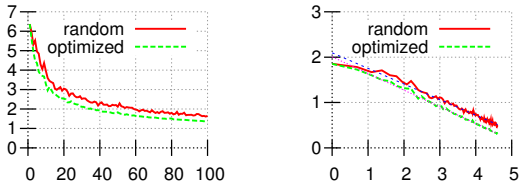
Figures 9 and 10 each evaluate the average interaction error in pixels for an increasing number of viewports and layers. As expected, the evaluation of the error appears to be proportional to the inverse of the number of layers $\mathrm{I} = \mathcal{O}(m^{-1})$, as presented in Theorem 3. The increase of

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



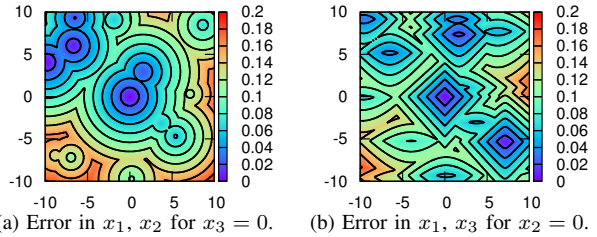
(a) Interaction error over m layers. (b) Log-Log scale.

Figure 9. Interaction error for $n = 1$ viewport.



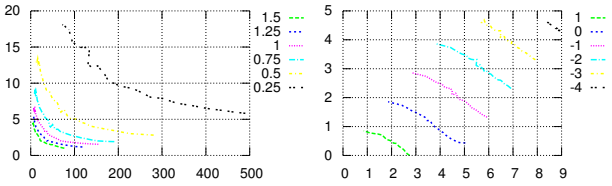
(a) Interaction error over n viewports. (b) Log-Log scale.

Figure 10. Interaction error for $m = 1$ layer.



(a) Error in x_1, x_2 for $x_3 = 0$. (b) Error in x_1, x_3 for $x_2 = 0$.

Figure 11. Screen space error distribution in pixels over view position changes in pixels for 27 random viewports with 27 layers.



(a) Bandwidth over layers. (b) Log-Log-Log contour lines.

Figure 12. Error contour lines for number of total images over layers.

viewports, on the other hand, involves choosing the locations of each reference viewport.

The theoretic results detail the asymptotic behavior, but they are quite rough for viewport numbers other than fifth power of whole numbers. Thus, the performance could deviate from our expectations for specific cases in real scenarious. But this is not the case, as Figure 10 illustrates the validity of the asymptotics for realistic cases.

Due to the computational costs for viewport position optimization, the results were compared to random distribution of viewports within X. To avoid unnecessarily skewed results for random placing, the first viewport was set to the center of X. In general, the optimized placement improves the performance, but both methods expose nearly the same order of convergence of $\mathrm{I} \approx \mathcal{O}(n^{-1/3})$. The optimized curve also shows some irregularities due to geometric effects for certain viewport numbers, for example a cubic number of viewports can be place positioned more efficiently in a cube than any other number. The gain of optimization is significant, as for example the interaction error of 2 is reached for $n = 36$ in the optimized version, whereas the random method needs $n = 59$. Also the rate of convergence appears to be slightly worse, but considering the computational costs for optimization, a trade-off can be considered for real-time applications.

Figure 11 illustrates the pixel errors for 27 randomly chosen viewports with 27 layers each in two cut-planes through X for $x_3 = 0$ and $x_2 = 0$, corresponding to Figure 8. The first viewport located at $x_1 = x_2 = x_3 = 0$ is the only viewport placed on the cut-plane, all others were randomly distributed in X. Depending of the position, the scene will be visualized using the layers from the viewport with lowest error contribution, leading to a continuous error function over X.

The actual performance of the method is evaluated in Figure 12, plotting the interaction error over the total bandwidth in dependence of number of layers needed. The total bandwidth is estimated by the number of images nm for n viewports each having m layers that need to be transmitted. There is a problem of sparse data for evaluation, for example 15 total images can result from either 3 viewports with 5 layers, and vice versa. This was overcome by using differing layer numbers during the tests, yielding the fractional layer number needed, for example 15 total images split on 2 viewports with 7.5 layers, in average. Figure 12 (a) denotes the total images on the x-axis, and average layers on the y-axis. The graphs are contour-lines of same interaction error. It is clearly visible, that both the increase of bandwidth and the increase in layers with constant bandwidth reduces the error. This clearly shows, that given a limited bandwidth, the interaction error can be as low as how many layers the output device can handle. Additionally, Figure 12 (b) presents the performance $\mathrm{I} \approx \mathcal{O}(m^{-1}n^{-1/3})$ in the experiment, as it was predicted before.

VIII. Conclusion

In this paper, we developed a mathematical model which allows to measure, analyze and optimize the display error of image-based approximation techniques, presented an algorithm for viewport location optimization, and evaluated the performance of the method under realistic conditions. Both the error asymptotics derived for our method based on parallelized rendering, as well as the experimental results, show a clear advantage over traditional remote visualization

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

concepts like Virtual Network Computing (VNC) which, under ideal conditions, represent the scene by one image $m = 1$ without image warping, leading to doubled interaction errors than presented here.

In contrast to this, $m = 10$ impostors with $n = 1$ viewport cover the same volume of permissible viewports as $m = 1$ impostors for $n = 100000$ optimally chosen viewport sets. The latter is using a bandwidth of $\mathcal{O}(mn)$ that is 10000-fold higher. Comparing this to the bandwidth needed for transmission of impostors compared with their error contribution $\mathcal{O}(m^{-1}n^{-1/5})$, the method offers significant decrease of bandwidth consumption. By avoiding high network latencies, the user experiences low latency rendering.

The proposed method strongly benefits from graphical capabilities of clients, such as mobile devices, and will increase its efficiency for each new generation providing increased graphical performance. Due to the parallelization of server-sided image generation, and the proven efficiency thereof, the method is applicable to large and distributed data sets for visualization on mobile devices and thin clients, also including augmented reality applications [23].

ACKNOWLEDGMENT

The authors appreciate the support of the 'Federal Ministry of Education and Research' and 'Eurostars' within the Project E! 5643 MobileViz. The Eurostars Project is funded by the European Union.

REFERENCES

[1] A. Helfrich-Schkarbanenko, V. Heuveline, R. Reiner, and S. Ritterbusch, "Bandwidth-efficient parallel visualization for mobile devices," in *INFOCOMP 2012, The Second International Conference on Advanced Communications and Computation*, 2012, pp. 106–112.

[2] F. Lamberti and A. Sanna, "A streaming-based solution for remote visualization of 3D graphics on mobile devices," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 2, pp. 247–260, march-april 2007.

[3] H.-Y. Shum and S. B. Kang, "A review of image-based rendering techniques," in *IEEE/SPIE Visual Communications and Image Processing*, 2000, pp. 2–13.

[4] S. Jeschke and M. Wimmer, "An error metric for layered environment map impostors," Institute of Computer Graphics and Algorithms, Vienna University of Technology, Technical Report TR-186-2-02-04, 2002.

[5] S. Jeschke, M. Wimmer, and W. Purgathofer, "Image-based representations for accelerated rendering of complex scenes," *Eurographics 2005 STAR Report*, vol. 1, 2005.

[6] V. Heuveline, M. Baumann, S. Ritterbusch, and R. Reiner, "Method and system for scene visualization," Mar. 1 2013, wO Patent 2,013,026,719.

[7] S. Jeschke, M. Wimmer, and H. Schuman, "Layered environment-map impostors for arbitrary scenes," *Graphics Interface*, pp. 1–8, May 2002.

[8] W.-C. Wang, K.-Y. Li, X. Zheng, and E.-H. Wu, "Layered Textures for Image-Based Rendering," *Journal of Computer Science and Technology*, vol. 19, no. 5, pp. 633–642, September 2004.

[9] M. Moser and D. Weiskopf, "Interactive volume rendering on mobile devices," in *13th Fall Workshop: Vision, Modeling, and Visualization 2008*, O. Deussen, D. Keim, and D. Saupe, Eds. Akademische Verlagsgesellschaft AKA GmbH, 2008, p. 217.

[10] J. Cohen, D. Manocha, and M. Olano, "Simplifying polygonal models using successive mappings," in *VIS '97: Proceedings of the 8th Conference on Visualization '97*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1997, p. 395.

[11] P. Debevec, Y. Yu, and G. Boshokov, "Efficient view-dependent image-based rendering with projective texture-mapping," University of California at Berkeley, Berkeley, CA, USA, Technical Report, 1998.

[12] J. Kopf, B. Chen, R. Szeliski, and M. Cohen, "Street slide: browsing street level imagery," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 1–8, 2010.

[13] J. Shade, D. Lischinski, D. H. Salesin, T. DeRose, and J. Snyder, "Hierarchical image caching for accelerated walkthroughs of complex environments," in *SIGGRAPH '96: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY, USA: ACM, 1996, pp. 75–82.

[14] T. A. Funkhouser, "Database management for interactive display of large architectural models," in *GI '96: Proceedings of the Conference on Graphics Interface '96*. Toronto, Ont., Canada, Canada: Canadian Information Processing Society, 1996, pp. 1–8.

[15] M. Hoffmann and J. Kohlhammer, "A Generic Framework for Using Interactive Visualization on Mobile Devices," *Communications in Computer and Information Sciene*, vol. 53, no. 4, pp. 131–142, 2009.

[16] F. Lamberti, C. Zunino, A. Sanna, A. Fiume, and M. Maniezzo, "An accelerated remote graphics architecture for PDAs," in *Web3D '03: Proceedings of the Eighth International Conference on 3D Web Technology*. New York, NY, USA: ACM, 2003, pp. 55–61.

[17] A. Beutelspacher and U. Rosenbaum, *Projective Geometry: From Foundations to Applications*. Cambridge University Press, February 1998.

[18] R. Reiner, "Numerical Methods for Optimal Impostor Prefetching in Scientific Visualization," Diploma Thesis, Karlsruhe Institute of Technology, 2011.

[19] R. C. St.John and N. R. Draper, "D-optimality for regression designs: a review," *Technometrics*, vol. 17, no. 1, pp. 15–23, 1975.

[20] V. B. Melas, *Functional Approach to Optimal Experimental Design (Lecture Notes in Statistics)*. Springer-Verlag New York, Inc., 2005.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[21] A. F. Emery and A. V. Nenarokomov, "Optimal experiment design," *Measurement Science and Technology*, vol. 9, no. 6, pp. 864–876, 1999.

[22] D. Ucinski, *Optimal measurement methods for distributed parameter system identification*. CRC Press, 2004.

[23] V. Heuveline, S. Ritterbusch, and S. Ronnas, "Augmented reality for urban simulation visualization," in *INFOCOMP 2011, The First International Conference on Advanced Communications and Computation*. IARIA, 2011, pp. 115–119.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

LUT Saving in Embedded FPGAs for Cache Locking in Real-Time Systems

Antonio Martí Campoy, Francisco Rodríguez-Ballester, and Rafael Ors Carot
Departamento de Informática de Sistemas y Computadores
Universitat Politècnica de València
46022, València, Spain
e-mail: {amarti, prodrig, rors}@disca.upv.es

***Abstract*—In recent years, cache locking have appeared as a solution to ease the schedulability analysis of real-time systems using cache memories maintaining, at the same time, similar performance improvements than regular cache memories. New devices for the embedded market couple a processor and a programmable logic device designed to enhance system flexibility and increase the possibilities of customisation in the field. This arrangement may help to improve the use of cache locking in real-time systems. This work proposes the use of this embedded programmable logic device to implement a logic function that provides the cache controller the information it needs in order to determine if a referenced main memory block has to be loaded and locked into the cache; we have called this circuit a Locking State Generator. Experiments show the requirements in terms of number of hardware resources and a way to reduce them and the circuit complexity. This reduction ranges from 50% up to 80% of the number of hardware resources originally needed to build the Locking State Generator circuit.**

***Keywords*—Real-Time Systems; Cache Locking; FPGA; Memory Hierarchy**

I. INTRODUCTION

In a previous work [1], the authors proposed and evaluated the use of an embedded Field-Programmable Gate Array (FPGA) to implement a lockable cache. The FPGA was used to build a logic circuit that signals to the cache controller if a main memory block should be loaded and locked in cache. This paper extends previous work presenting a way to reduce hardware resources when implementing the logic circuit by means of an FPGA.

Cache memories are an important advance in computer architecture, offering a significant performance improvement. However, in the area of real-time systems, the use of cache memories introduces serious problems regarding predictability. The dynamic and adaptive behaviour of a cache memory reduces the average access time to main memory, but presents a non deterministic fetching time [2]. This way, estimating execution time of tasks is complicated. Furthermore, in preemptive multi-tasking systems, estimating the response time of each task in the system becomes a problem with a solution hard to find due to the interference on the cache contents produced among the tasks. Thus, schedulability analysis requires complicated procedures and/or produces overestimated results.

In recent years, cache locking have appeared as a solution to ease the schedulability analysis of real-time systems using cache memories maintaining, at the same time, similar performance improvements of systems populated with regular cache memories. Several works have been presented to apply cache locking in real-time, multi-task, preemptive systems, both for instructions [3][4][5][6] and data [7]. In this work, we focus on instruction caches only, because 75% of accesses to main memory are to fetch instructions [2].

A locked cache is a cache memory without replacement of contents, or with contents replacement in a priori and well known moments. When and how contents are replaced define different uses of the cache locking mechanism.

One of the ways to use cache locking in preemptive real-time systems is called dynamic use [3]. In the dynamic use cache contents change only when a task starts or resumes its execution. From that moment on cache contents remain unchanged until a new task switch happens. The goal is that every task may use the full size of the cache memory for its own instructions.

The other possible use of cache locking is called static use [8][9]. When a cache is locked in this way the cache contents are pre-loaded on system power up and remain constant while the system runs. For example, a simple software solution may be used to issue processor instructions to explicitly load and lock the cache contents. How the cache contents are pre-loaded is irrelevant; what is important is that the cache behaviour is now completely deterministic. The drawback of this approach is that the cache must be shared among the code of all tasks so the performance improvement is diminished.

This paper focuses on the dynamic use of locked cache and is organized as follows. Section II describes previous implementation proposals for dynamic use of cache locking in real-time systems, and the pursued goals of this proposal to improve previous works. Section III presents a detailed implementation of the Locking State Generator (LSG), a logic function that signals the cache controller whether to load a referenced main memory block in cache or not. Section IV presents some analysis about the complexity of the proposal, and then Section V shows results from experiments carried out to analyse resource requirements in the LSG implementation in terms of number of LUTs (Look-Up Tables) needed to build the circuit. Section VI presents a way to reduce the complexity of the LSG by means of reusing LUTs when implementing the mini-terms of the LSG logic function. Finally, this paper ends with the ongoing work and conclusions.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

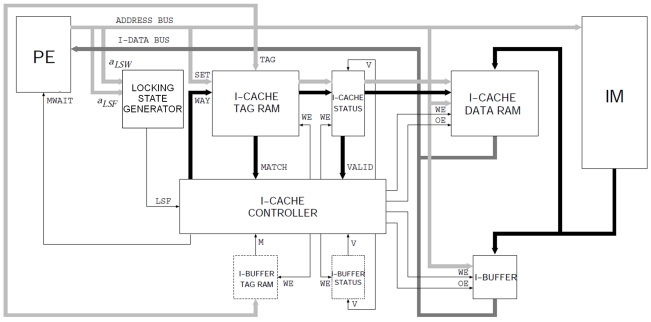


Fig. 1: The LSG architecture.

II. State of the Art

Two ways of implementing dynamic use of cache locking can be found in the bibliography. First of them, [3], uses a software solution, without hardware additions and using processor instructions to explicitly load and lock the cache contents. This way, every time a task switch happens, the operating system scheduler runs a loop to read, load and lock the selected set of main memory blocks into the cache memory for the next task to run. The list of main memory blocks selected to be loaded and locked in cache is stored in main memory.

The main drawback of this approach is the long time needed to execute the loop, which needs several main memory accesses for each block to be loaded and locked.

In order to improve the performance of the dynamic use of cache locking, a Locking State Memory (LSM) is introduced in [4]. This is a hardware solution where the locking of memory blocks in cache is controlled by a one-bit signal coming from a specialized memory added to the system. When a task switch happens, the scheduler simply flushes the cache contents and a new task starts execution, fetching instructions from main memory. But not all referenced blocks during task execution are loaded in cache, only those blocks selected to be loaded and locked are loaded in cache. In order to indicate whether a block has to be loaded or not the LSM stores one bit per main memory block. When the cache controller fetches a block of instructions from main memory, the LSM provides the corresponding bit to the cache controller. The bit is set to one to indicate that the block has to be loaded and locked in cache, and the cache controller stores this block in cache. If the bit is set to zero, indicates that the block was not selected to be loaded and locked in cache, so the cache controller will preclude the store of this block in cache, thus change in cache contents are under the control of the LSM contents and therefore under the control of system designer.

The main advantage of the LSM architecture is the reduction of the time needed to reload the cache contents after a preemption compared against the previous, software solution.

The main drawback of the LSM is its poor scalability. The size of the LSM is directly proportional to main memory and cache-line sizes (one bit per each main memory block, where the main memory block size is equal to the cache line size). This size is irrespective of the size of the tasks, or the number of memory blocks selected to be loaded and locked into the cache. Moreover, the LSM size is not related to the cache size. This way, if the system has a small cache and a very large main memory, a large LSM will be necessary to select only a tiny fraction of main memory blocks.

In this work, a new hardware solution is proposed, where novel devices found in the market are used. These devices couples a standard processor with an FPGA, a programmable logic device designed to enhance system flexibility and increase the possibilities of customisation in the field. A logic function implemented by means of this FPGA substitutes the work previously performed by the LSM. For the solution presented here hardware complexity is proportional to the size of system, both software-size and hardware-size. Not only the circuit required to dynamically lock the cache contents may be reduced but also those parts of the FPGA not used for the control of the locked cache may be used for other purposes. We have called this logic function a Locking State Generator (LSG) and think our proposal simplifies and adds flexibility to the implementation of a real-time system with cache locking.

III. The proposal: Locking State Generator

Recent devices for the embedded market [10][11] couple a processor and an FPGA, a programmable logic device designed to enhance system flexibility and increase the possibilities of customisation in the field. This FPGA is coupled to an embedded processor in a single package (like the Intel's Atom E6x5C series [10]) or even in a single die (like the Xilinx's Zynq-7000 series [11]) and may help to improve the use of cache locking in real-time systems.

Deciding whether a main memory block has to be loaded in cache is the result of a logic function with the memory address bits as its input. This work proposes the substitution of the Locking State Memory from previous works by a logic function implemented by means of this processor-coupled FPGA; we have called this element a Locking State Generator (LSG).

Two are the main advantages of using a logic function instead of the LSM. First, the LSG may adjust its complexity and circuit-related size to both the hardware and software characteristics of the system. While the LSM size depends only on the main memory and cache-line sizes, the number of circuit elements needed to implement the LSG depends on the number of tasks and their sizes, possibly helping to reduce hardware. Second, the LSM needs to add a new memory and data-bus lines to the computer structure. Although LSM bits could be added directly to main memory, voiding the requirement for a separate memory, in a similar way as extra bits are added to ECC DRAM, the LSM still requires modifications to main memory and its interface with the processor. In front of that the LSG uses a hardware that is now included in the processor package/die. Regarding modifications to the cache controller, both LSM and LSG present the same requirements as both require that the cache controller accepts an incoming bit to determine whether a referenced memory block has to be loaded and locked into the cache or not.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Figure 1 shows the proposed architecture, similar to the LSM architecture, with the LSG logic function replacing the work of the LSM memory.

A. Implementing logic functions with an FPGA

An FPGA implements a logic function combining a number of small blocks called logic cells. Each logic cell consists of a Look-Up Table (LUT) to create combinational functions, a carry-chain for arithmetic operations and a flip-flop for storage. The look-up table stores the value of the implemented logic function for each input combination, and a multiplexer inside the LUT is used to provide one of these values; the logic function is implemented simply connecting its inputs as the selection inputs of this multiplexer.

Several LUTs may be combined to create large logic functions, functions with input arity larger than the size of a single LUT. This is a classical way of implementing logic functions, but it is not a good option for the LSG: the total number of bits stored in the set of combined LUTs would be the same as the number of bits stored in the original LSM proposal, just distributing the storage among the LUTs.

1) Implementing mini-terms: In order to reduce the number of logic cells required to implement the LSG, instead of using the LUTs in a conventional way this work proposes to implement the LSG logic function as the sum of its mini-terms (the sum of the input combinations giving a result of 1).

This strategy is not used for regular logic functions because the number of logic cells required for the implementation heavily depends on the logic function itself, and may be even larger than with the classical implementation. However, the arity of the LSG is quite large (the number of inputs is the number of memory address bits) and the number of cases giving a result of one is very small compared with the total number of cases, so the LSG is a perfect candidate for this implementation strategy.

A mini-term is the logic conjunction (AND) of the input variables. As a logic function, this AND may be built using the LUTs of the FPGA. In this case, the look-up table will store a set of zero values and a unique one value. This one value is stored at position j in order to implement mini-term j. Figure 2 shows an example for mini-term 5 for a function of arity 3, with input variables called C, B and A, where A is the lowest significant input.

For the following discussion we will use 6-input LUTs, as this is the size of the LUTs found in [11]. Combining LUTs to create a large mini-term is quite easy; an example of a 32-input mini-term is depicted in Figure 3 using a two-level associative network of LUTs. Each LUT of the first level (on the left side) implements a 1/6 part of the mini-term (as described in the previous section). At the second level (on the right side), a LUT implements the AND function to complete the associative property.

2) Sum of mini-terms: For now, we have used 7 LUTs to implement one mini-term. To implement the LSG function we have to sum all mini-terms that belong to the function; a mini-term k belongs to a given logic function if the output of the function is one for the input case k. In this regard,

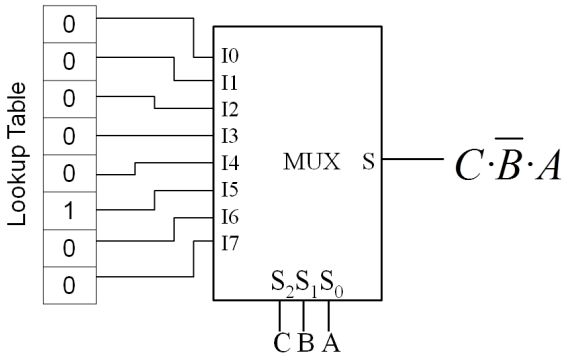


Fig. 2: Implementing mini-term 5 of arity 3 (C, B, A are the function inputs).

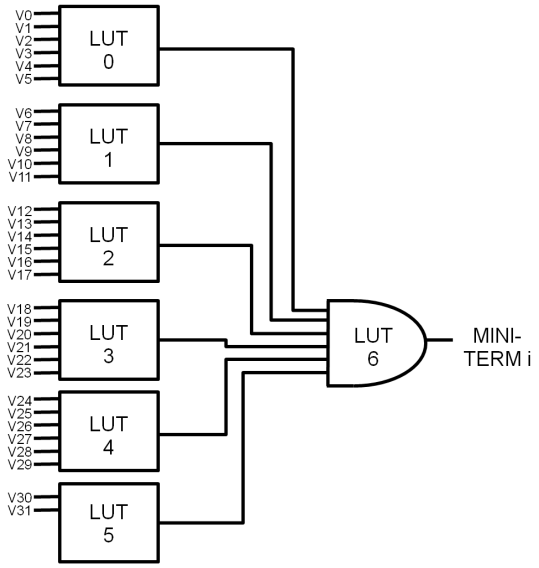


Fig. 3: Implementing a 32-input mini-term using 6-input LUTs.

two questions arise: first, how many mini-terms belong to the function, and second, how to obtain the logic sum of all of them.

The first question is related to the software parameters of the real-time system we are dealing with. If the real-time system comprises only one task, the maximum number of main memory blocks that can be selected to load and lock in cache is the number of cache lines (L). If the real-time system is comprised of N tasks this value is $L \times N$ because, in the dynamic use of cache locking, each task can use the whole cache for its own instructions.

A typical L1 instruction cache size in a modern processor is 32KB; assuming each cache line contains four instructions and that each instructions is 4B in size, we get $L = (32KB/4B)/4$ instructions $= 2K$ lines.

This means that, for every task in the system, the maximum number of main memory blocks that can be selected is around

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

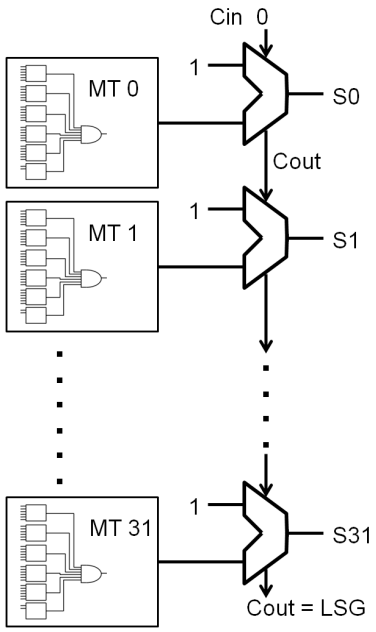


Fig. 4: Implementing the LSG function.

2000. Supposing a real-time system with ten tasks, we get a total maximum of 20 000 selectable main memory blocks. That is, the LSG function will have 20 000 mini-terms. Summing all these mini-terms by means of a network of LUTs to implement the logic OR function with 20 000 inputs would require around 4000 additional LUTs in an associative network of 6 levels.

The solution to reduce the complexity of this part of the LSG is to use the carry chain included in the logic cells for arithmetic operations. Instead of a logic sum of the mini-terms, an arithmetic sum is performed: if a binary number in which each bit position is the result of one of the mini-terms is added with the maximum possible value (a binary sequence consisting of ones), the result will be: i) the maximum possible value and the final carry will be set to zero (if the outputs of all mini-terms are zero for the memory address used as input to the LSG), or ii) the result will be $M-1$ and the final carry will be set to one (being $M > 0$ the number of mini-terms producing a one for the memory address). Strictly speaking, mini-terms are mutually exclusive, so one is the maximum value for M. In the end, the arithmetic output of the sum is of no use, but the final carry indicates if the referenced main memory block has to be loaded and locked in cache. Figure 4 shows a block diagram of this sum applied to an example of 32 mini-terms, each one nominated MTk.

Using the carry chain included into the LUTs which are already used to calculate the LSG function mini-terms produce a very compact design. However, a carry chain adder of 20 000 bits (one bit per mini-term) is impractical, both for performance and routing reasons. In order to maintain a compact design with a fast response time, a combination of LUTs and carry-chains are used, as described below.

First, the 20 000 bits adder is split into chunks of reasonable size; initial experiments carried out indicate this size to be between 40 and 60 bits in the worst case, resulting into a set of 500 to 330 chunks. All these chunk calculations are performed in parallel using the carry chains included into the same logic cells used to calculate the mini-terms, each one providing a carry out. These carries have to be logically or-ed together to obtain the final result. A set of 85 to 55 6-input LUTs working in parallel combine these 330 to 500 carries, whose outputs are arithmetically added with the maximum value using the same strategy again, in this case using a single carry chain. The carry out of this carry chain is the LSG function result.

IV. Evaluation of the LSG

The use of the LSG to lock a cache memory is a flexible mechanism to balance performance and predictability as it may have different modes of operation. For real-time systems, where predictability is of utmost importance, the LSG may work as described here; for those systems with no temporal restrictions, where performance is premium, the LSG may be easily forced to generate a fixed one value, obtaining the same cache behaviour with a locked cache than with a regular cache. It can even be used in those systems mixing real-time and non real-time tasks, as the LSG may select the proper memory blocks for the former in order to make the tasks execution predictable and provide a fixed one for the latter to improve their performance as with a regular cache memory.

Initial experiments show timing is not a problem for the LSG as its response time has to be on par with the relatively slow main memory: the locking information is not needed before the instructions from main memory. Total depth of the LSG function is three LUTs and two carry chains; register elements are included into the LSG design to split across several clock cycles the calculations in order to increase the circuit operating frequency and to accommodate the latency of main memory as the LSG has to provide the locking information no later the instructions from main memory arrive. Specifically, the carry out of all carry chains are registered in order to increase the operating frequency.

Regarding the circuit complexity, the following calculations apply: although the address bus is 32 bits wide, the LSG, like the cache memory, works with memory blocks. Usually a memory block contains four instructions and each instruction is 4B, so main memory blocks addresses are actually 28 bits wide.

Generating a mini-term with a number of inputs between 25 to 30 requires 6 LUTs in a two-level network. Supposing a typical cache memory with 2000 lines, 12 000 LUTs are required. But if the real-time system has ten tasks, the number of LUTs needed for the LSG grows up to 120 000. It is a large number, but more LUTs may be found on some devices currently available [11]. Calculating the logic OR function of all these mini-terms in a classical way adds 4000 more LUTs to the circuit, but the described strategy merging LUTs and carry chains reduce this number to no more than 500 LUTs in the worst case.

The estimated value of 120 000 LUTs required to build the LSG function is an upper bound, and there are some ways this

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE I: Cache sizes used in experiments

	Size (lines)	Size (instructions)	Size (bytes)
1	64	256	1K
2	128	512	2K
3	256	1K	4K
4	512	2K	8K
5	1024	4K	16K
6	2048	8K	32K
7	4096	16K	64K

number may be reduced. A real-time system with five tasks will need just half this value of LUTs. The same is true if the cache size is divided by two. Following sections show some experiments and a easy way to reduce the total number of LUTs.

V. EXPERIMENTS

Previous sections have detailed, in a theoretical way, an upper bound of the number of LUTs required to implement the LSG. Experiments conducted in this section provide more realistic values, and identify both hardware and software characteristics that affect the number of required LUTs in order to implement the LSG for a particular system.

Regarding hardware characteristics the size of cache memory, measured in lines, is the main parameter because this number of lines is the maximum number of blocks a task may select to load and lock in cache. And, in a first approach, every block selected to be locked needs a mini-term in the LSG implementation in order to identify it when it is fetched by the processor.

As described previously, it is not possible to build a mini-term with only one LUT, because the number of inputs of the latter, ranging from 4 up to 7 inputs [12] in today devices is not enough to accommodate the inputs of the former.

Mini-terms are then implemented combining several LUTs. Thus the number of inputs of LUTs is also a main characteristic, because the lower the number of LUT inputs, the higher the number of LUTs needed to build a mini-term. Finally, width of address bus (measured in bits) is also a parameter to be taken into account, because the number of variables in a mini-term is the number of lines in the address bus.

Regarding software parameters, the number of tasks in the system presents the larger impact in the number of needed LUTs. In dynamic use of cache locking, irrespective of the use of software reload, LSM or the here proposed LSG, every task in the system may select as many blocks to load and lock in cache as cache lines are available. So, the number of LUTs needed to build the LSG circuit will be a multiple of the number of system tasks.

Other software parameters like size, periods or structure of tasks do not affect the number of LUTs needed, or their effect is negligible.

In order to evaluate the effect of these characteristics, and to obtain realistic values about the number of required LUTs, experiments described below have been accomplished.

TABLE II: Main characteristics of systems used in experiments

System	Number of tasks	Task average size (blocks)
1	4	849
2	5	158
3	4	429
4	4	641
5	5	424
6	3	855
7	8	205
8	3	1226
9	5	617
10	3	1200
11	3	476
12	3	792

Hardware architecture and software systems are the same, or a subset of those described and used in [3][13].

The hardware architecture is based on the well-known MIPS R2000 architecture, added with a direct-mapping instruction cache memory (i-cache). The data size of this i-cache range from 64 up to 4096 lines. The size of one cache line is the same as one main memory block, and it is 16B (four instructions of four bytes each). Seven cache sizes have been used, as described in Table I. Although MIPS R2000 address bus is 32 bits wide, it has been reduced to 16 bits in the following experiments, giving a maximum size of 64KB of code.

Regarding the number of LUT inputs, four cases have been studied: LUTs with 4, 5, 6, and 7 input variables.

Regarding the software used in experiments, tasks are artificially created to stress the cache locking mechanism. Main parameters of tasks are defined, such as the number of loops and their nesting level, the size of the task, the size of its loops, the number of if-then-else structures and their respective sizes. A simple tool is used to create such tasks. The workload of any task may be a single loop, if-then-else structures, nested loops, streamline code, or any mix of these. The size of a task code may be large (close to the 64KB limit) or short (less than 1KB). 12 different sets of tasks are defined, and with these sets a total of 24 real-time systems have been created modifying the periods of the tasks. Task periods are hand-defined to make the system schedulable, and the task deadlines are set to be equal to the task period. Finally, the priority is assigned using a Rate Monotonic policy (the shorter the period the higher the priority). Table II shows the main characteristics of the systems used for this experimentation.

Using cache locking requires a careful selection of those instructions to be loaded and locked in cache. It is possible to make a random selection of instructions: that would provide predictability to the temporal behaviour of system, but there would be no warranty about system performance. Several algorithms have been proposed to select cache contents [14].

For this work, a genetic algorithm is used. The target of the genetic algorithm is to find the set of main memory blocks that, loaded and locked in cache, provides the lower utilisation for the whole system. In order to achieve this objective, the genetic algorithm gets as inputs the number of tasks, their periods and temporal expressions [15] that are needed to calculate Worst Case Execution Time and Worst Case Response Time.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

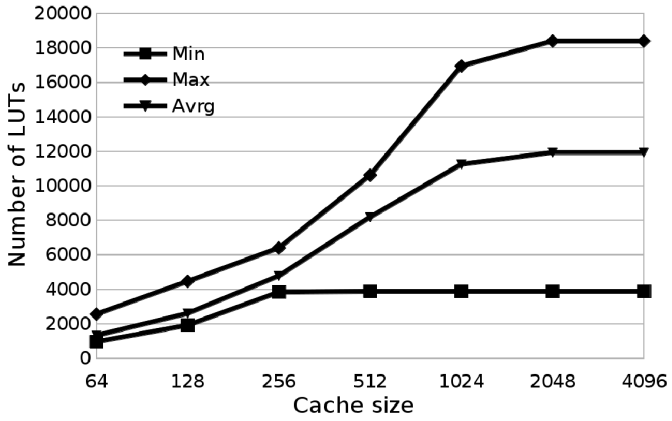


Fig. 5: Number of 4-inputs LUTs required.

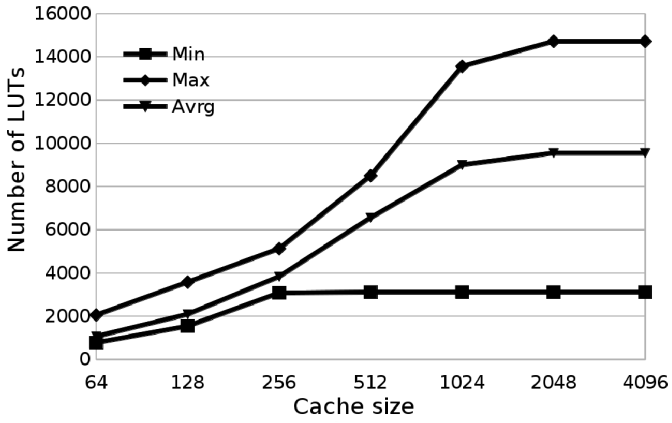


Fig. 6: Number of 5-inputs LUTs required.

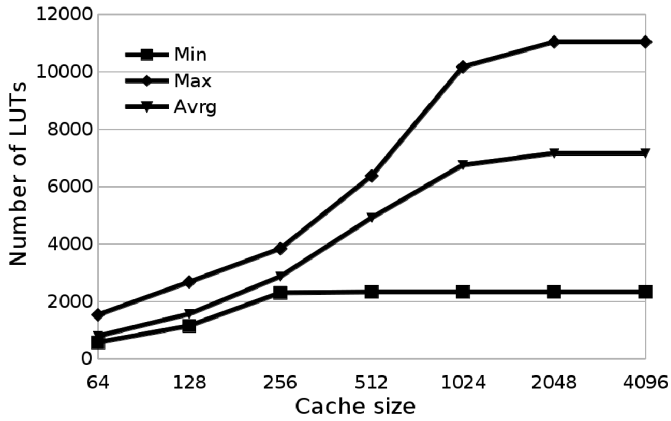


Fig. 7: Number of 6-inputs LUTs required.

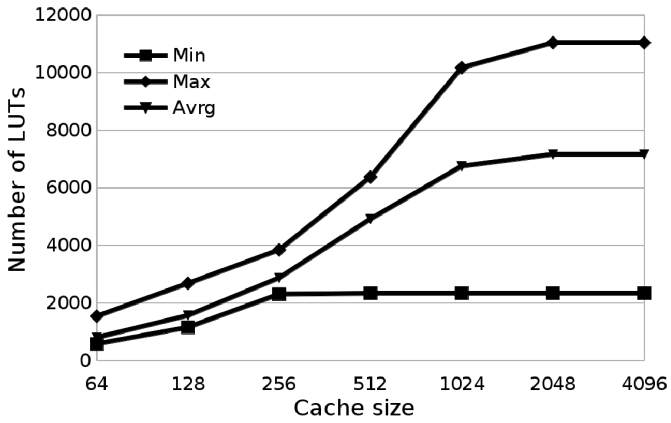


Fig. 8: Number of 7-inputs LUTs required.

Also the cache parameters like line size, total cache size, mapping policy and hit and miss times are inputs to the genetic algorithm.

The solution found by the genetic algorithm, that is, the set of main memory blocks selected for each task in the system, has to meet cache requirements like cache size and mapping policy. As output the genetic algorithm determines if the system is schedulable, the worst case execution time and worst response time for all the tasks, and the list of selected main memory blocks for each task to be loaded and locked in the cache. This list of blocks has been used in this work to calculate the number of required LUTs to implement the LSG.

Figure 5 shows the number of LUTs needed to build the mini-terms of each one of the 24 systems, as a function of the cache size using 4-input LUTs. Graph shows the maximum value, the minimum value, and the average number of LUTs for the 24 systems. Figures 6, 7, and 8 show the same information than Figure 5, using LUTs of 5, 6, and 7 inputs, respectively.

The four figures are identical in shape and tendency, but present some differences in their values. As expected the most noticeable is the effect of cache size. There is a clear and positive relationship between the cache size and the number of required LUTs. And regarding average values, this increment is very close to a lineal increase.

But there are two exceptions, both for the same reason. For the curve of minimum values, it presents a zero slope when cache size is larger than 256 lines. This is because the tasks in set 2 have a size lower than 256 main memory blocks (in average, size of tasks is 158; see Table II), but none of the tasks is larger than 256 blocks. This means that for each task, the genetic algorithm will select no more than 158 blocks, so, no matter the cache size, a maximum of 158 blocks multiplied by 5 (number of tasks in this system) will be selected and, thus, implemented as mini-terms.

Since the largest task in all systems is close to 2000 blocks, when the cache reaches a size of 2048 lines or larger, it

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

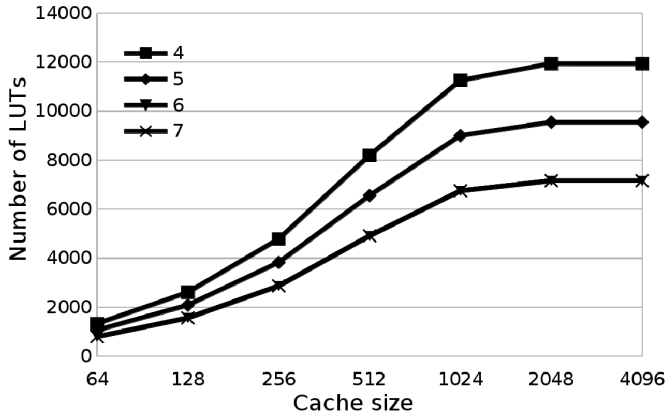


Fig. 9: Average LUTs required for LUTs of 4, 5, 6, and 7 inputs.

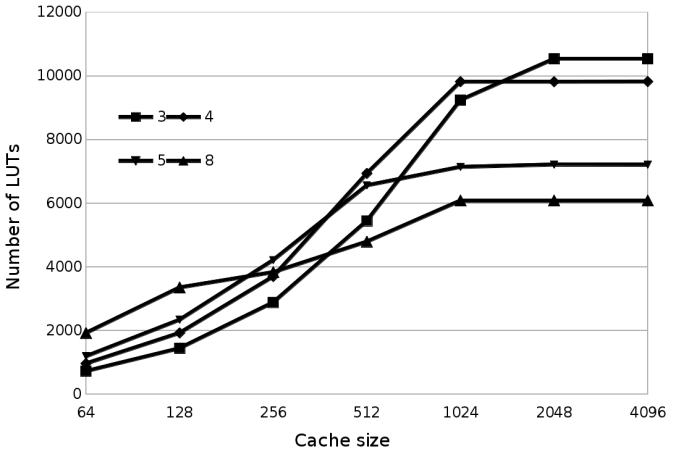


Fig. 10: Average LUTs required for number of system tasks (3, 4, 5 and 8 tasks).

does not affect the number of LUTs needed, because the number of blocks selected, and thus the number of mini-terms to be implemented, cannot be larger. Numerical differences between maximum and minimum values maybe explained by differences in tasks structures or genetic algorithm executions, but most probably differences come from the number of tasks in each system. However, the effect of cache size and the existence of tasks with sizes smaller than the largest caches prevent to clearly state this idea. Regarding the effect of the number of LUT inputs, there are significant differences in the number of needed LUTs to implement the LSG when using LUTs of 4, 5, 6, and 7 inputs.

This effect is more important as cache size increases. For small cache sizes, the difference in the number of LUTs related to the number of LUT inputs is about some hundreds. But for large cache sizes, this difference is around five thousand LUTs. This effect is better appreciated in Figure 9, where average of needed LUTs for all systems and total number of LUT sizes is shown.

Figure 10 shows the average number of LUTs needed to implement the LSG, in front of cache size and number of tasks, for the 24 systems analysed. This figure shows that both cache size and number of tasks are important characteristics regarding the number of LUTs needed, but no one is more important than the other. When the cache size is small, and thus individual task sizes are larger than the cache size, the number of tasks in the system becomes a significant parameter regarding the number of needed LUTs, as shown for cache sizes of 64, 128, and 256 lines. However, when the cache becomes larger, the effect of the number of tasks seems to be the inverse. This is not completely true. Curves arrange in inverse order for small cache sizes than for large cache sizes, but this is because all systems must fit into the limit of 64KB of code, so systems with more tasks have smaller tasks while systems with fewer tasks have larger tasks. The conclusion is that the most important factor is neither the cache size nor the number of tasks, but the relationship between cache size and size of the tasks in the system. This factor, called System Size Ratio (SSR), was identified as one of the main factors deciding cache locking performance in [16].

VI. Reducing complexity

There is a way of reducing LSG circuit complexity without affecting the number of tasks in the system or the cache size. This simplification comes from the way each mini-term is implemented. As explained before, the number of inputs of a LUT is not enough to implement a whole mini-term, so the associative property is used to decompose the mini-term in smaller parts, each implemented using a LUT that are then combined using again a LUT performing the function of an AND gate, as shown in Figure 3.

As an example, consider two mini-terms of six variables implemented with 3-input LUTs. In order to implement the two mini-terms each one is decomposed in two parts, and each part is implemented by a LUT, using four LUTs to build what may be called half-mini-terms. Finally, two LUTs implementing an independent AND logic function each are used to combine these parts to finally implement both mini-terms. Consider now that both mini-terms have one of its part equal. In this case, implementing the same half-mini-term twice it is not mandatory, because the output of a LUT may be routed to two different AND gates, so mini-terms with some parts equal may share the implementation of that part. Figure 11 shows an example with two mini-terms sharing one of their parts.

Profiting from the limited number of inputs of the LUTs previous experiments have been repeated, but in this case an exhaustive search have been carried out to count the number of mini-terms that share some of their parts. The number of parts a mini-term is divided into depends on the number of LUT inputs, and four sizes have been used like in previous experiments: 4, 5, 6, and 7 inputs. This way, and considering a 16 bits address bus, a mini-term may be divided in three or four parts. In some cases, some inputs of some LUTs will

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

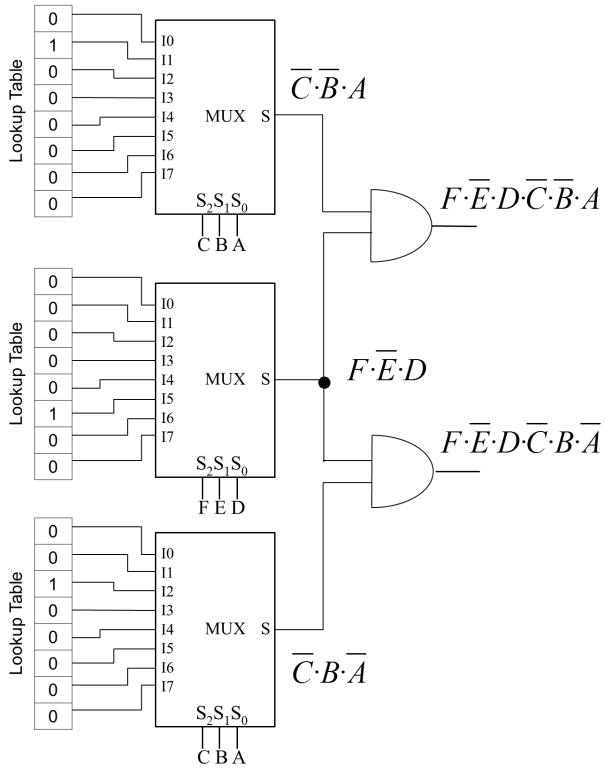


Fig. 11: Example of reducing LUTs needed to implement mini-terms.

not be used, being the worst case when using 7-inputs LUTs, because each mini-term requires 3 LUTs so there are 21 inputs available to implement mini-terms of arity 16.

A simple algorithm divides mini-terms in parts as a function of LUT size, and detects common parts between mini-terms. This exhaustive search is performed for the whole system, that is, it is not applied to mini-terms of the selected blocks of individual tasks but applied for all selected blocks of all tasks in the system.

Figure 12 shows the number of LUTs needed to build the mini-terms of each one of the 24 systems as a function of the cache size and using LUTs with 4 inputs, after applying the algorithm to search and reduce the LUTs needed due to the fact the implementation of common parts may be shared by the corresponding mini-terms. Graph shows the maximum value, the minimum value, and the average number of LUTs for the 24 systems. Figures 13, 14, and 15 show the same information than Figure 12 when the number of LUT inputs are 5, 6, and 7, respectively.

Figures 12, 13, 14, and 15 all present the same shape and the same values for minimum, average, and maximum curves, respectively. Numerical values show differences, but they are not significant, so it can be said that the number of LUT inputs does not affect the number of needed LUTs to implement the LSG when shared LUTs implementing common parts of mini-terms are used to reduce the total number of LUTs needed.

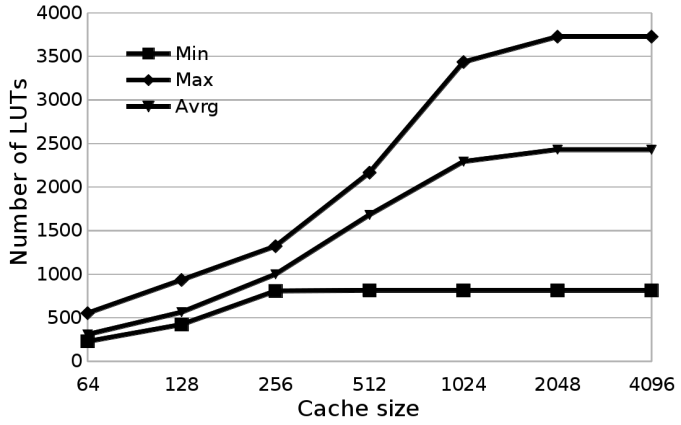


Fig. 12: Number of 4-inputs LUTs required after reduction.

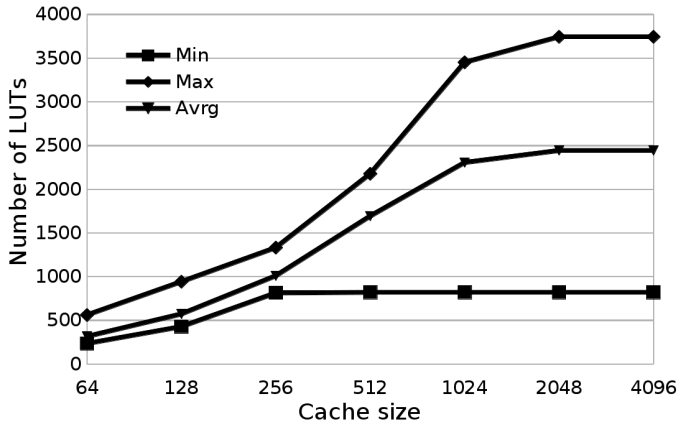


Fig. 13: Number of 5-inputs LUTs required after reduction.

This can be explained because when the number of LUT inputs is small the probability to find common parts among mini-terms increases. Figure 16 shows the average values of needed LUTs after reduction for the four LUT sizes considered. No significant differences appears in this graph. In front of average values for non-reduced implementation of the LSG, the number of required LUTs is between a 50% and a 80% when reducing the LSG implementation using common parts of mini-terms.

Regarding saving LUTs, shapes and tendencies of figures 12 to 15 are very similar to those in figures 5 to 8, so the effect of cache size, number of tasks in the system, and other parameters (except the LUT size) are similar for non-reduced and reduced implementation of the LSG.

Figure 17 shows the percentage of reduction in the number of LUTs regarding cache size and LUT size. The minimum reduction is 55% for a 64 lines cache size and 6-input LUTs, and a maximum reduction is close to 80% for 4-inputs LUTs and a cache of 512 lines or larger. The effect of cache size over reduction is more acute when using LUTs with six and

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

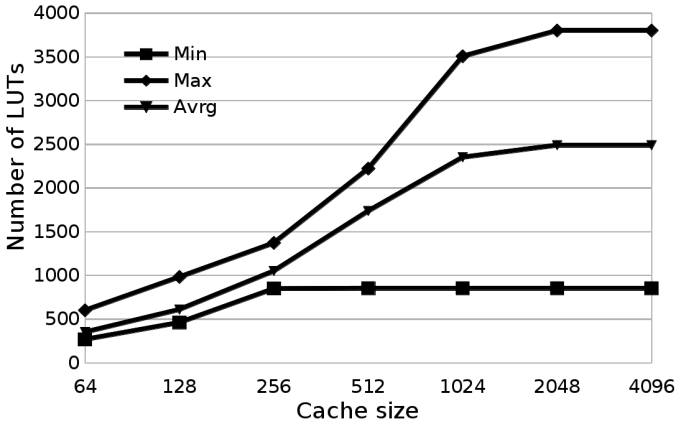


Fig. 14: Number of 6-inputs LUTs required after reduction.

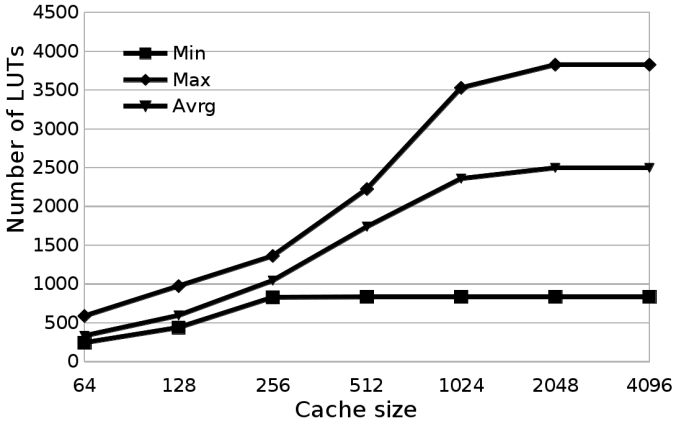


Fig. 15: Number of 7-inputs LUTs required after reduction.

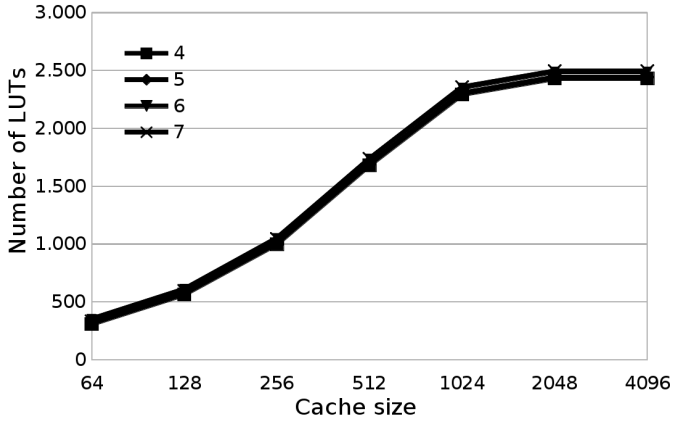


Fig. 16: Average LUTs required after reduction for LUTs of 4, 5, 6, and 7 inputs.

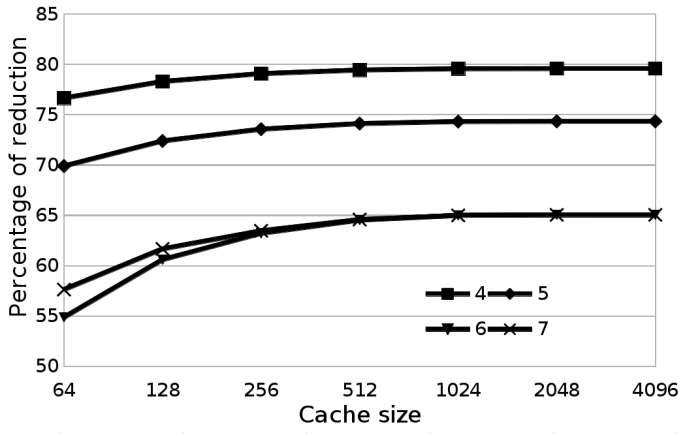


Fig. 17: Percentage of reduction as a function of cache size and number of LUT inputs.

seven inputs than when four and five inputs are used. However, the main effect over the percentage of reduction comes, as stated before, from the size of the LUTs. In absolute values, from the worst case of 14 000 LUTs needed to build the LSG (maximum for a cache size of 1024 lines in Figure 5), the reduced implementation of LSG lowers this number to 3500 (maximum for a cache size of 1024 lines in Figure 12).

VII. Ongoing work

The previous simplification may be improved by the selection algorithm, e.g., the genetic algorithm used to determine the main memory blocks that have to be loaded and locked into the cache. Usually, the goal of these algorithms is to provide predictable execution times and an overall performance improvement of the system and its schedulability, for example reducing global system utilisation or enlarging the slack of tasks to allow scheduling non-critical tasks. However, new algorithms may be developed that take into account not only this main goals, but that also that try to select blocks with common parts in their mini-terms, enhancing LUT reusing and reducing the complexity of the final LSG circuit. This is more than just wish or hope: for example, considering a loop with a sequence of forty machine instructions —10 main-memory blocks— the resulting performance is the same if the selected blocks are the five first ones or the last five ones, or even if the selected blocks are alternate blocks. Previous research show that genetic algorithms applied to this problem may produce different solutions, that is, different sets of selected main memory blocks, all with the same results regarding performance and predictability. So, next step in this research is the development of a selection algorithm that simultaneously tries to improve system performance and reduce the LSG circuit complexity.

What is performance and circuit complexity need to be carefully defined in order to include both goals in the selection algorithm. Once the algorithm works, the evaluation of

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

implementation complexity will be accomplished.

VIII. Conclusion

This work presents a new way of implementing the dynamic use of a dynamically locked cache for preemptive, real-time systems. The proposal benefits from recent devices coupling a processor with an FPGA, a programmable logic device, allowing the implementation of a logic function to signal the cache controller whether to load a main memory block in cache or not. This logic function is called a Locking State Generator (LSG) and replaces the work performed by the Locking State Memory (LSM) in previous proposals.

As the FPGA is already included in the same die or package of the processor, no additional hardware is needed as in the case of the LSM. Also, regarding circuit complexity, the LSG adapts better to the actual system as its complexity is related to both hardware and software characteristics of the system, an advantage in front of the LSM architecture, where the LSM size depends on the size of main memory exclusively. Results from experiments state than final LSG complexity is mainly related to cache size, and not main memory size as LSM is.

Implementation details described in this work show that it is possible to build the LSG logic function with commercial hardware actually found in the market.

Moreover, a way to reduce hardware requirements by means of reusing LUTs has been developed and experimented. Sharing LUTs among mini-terms allows a reduction in the number of LUTs needed to implement the LSG between 50% and 80%, and makes negligible the effect of LUT size over the number of LUTs needed.

Ongoing research steps about the selection algorithm of main memory blocks in order to reduce circuit complexity.

Acknowledgments

This work has been partially supported by PAID-06-11/2055 of Universitat Politècnica de València and TIN2011-28435-C03-01 of Ministerio de Ciencia e Innovación.

References

[1] A. M. Campoy, F. Rodríguez-Ballester, and R. Ors, "Using embedded fpga for cache locking in real-time systems," in *Proceedings of The Second International Conference on Advanced Communications and Computation*, INFOCOMP 2012, pp. 26–30, Oct 2012.

[2] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach, 4th Edition*. Morgan Kaufmann, 4 ed., 2006.

[3] A. M. Campoy, A. P. Ivars, and J. V. B. Mataix, "Dynamic use of locking caches in multitask, preemptive real-time systems," in *Proceedings of the 15th World Congress of the International Federation of Automatic Control*, 2002.

[4] J. B.-M. E. Tamura and A. M. Campoy, "Towards predictable, high-performance memory hierarchies in fixed-priority preemptive multitasking real-time systems," in *Proceedings of the 15th International Conference on Real-Time and Network Systems (RTNS-2007)*, pp. 75–84, 2007.

[5] J. C. K. Sascha Plazar and P. Marwedel, "Wcet-aware static locking of instruction caches," in *Proceedings of the 2012 International Symposium on Code Generation and Optimization*, pp. 44–52, 2012.

[6] L. C. Aparicio, J. Segarra, C. Rodríguez, and V. Vials, "Improving the wcet computation in the presence of a lockable instruction cache in multitasking real-time systems," *Journal of Systems Architecture*, vol. 57, no. 7, pp. 695 – 706, 2011. Special Issue on Worst-Case Execution-Time Analysis.

[7] X. Vera, B. Lisper, and J. Xue, "Data cache locking for tight timing calculations," *ACM Trans. Embed. Comput. Syst.*, vol. 7, pp. 4:1–4:38, Dec. 2007.

[8] M. Campoy, A. P. Ivars, and J. Busquets-Mataix, "Static use of locking caches in multitask preemptive real-time systems," in *Proceedings of IEEE/IEE Real-Time Embedded Systems Workshop (Satellite of the IEEE Real-Time Systems Symposium)*, IEEE, 2001.

[9] I. Puaut and D. Decotigny, "Low-complexity algorithms for static cache locking in multitasking hard real-time systems," in *Real-Time Systems Symposium, 2002. RTSS 2002. 23rd IEEE*, pp. 114–123, IEEE, 2002.

[10] I. Corp., "Intel atom processor e6x5c series-based platform for embedded computing." http://download.intel.com/embedded/processors/prodbrief/324535.pdf, 2013. [Online; accessed 15-March-2013].

[11] X. Inc., "Zynq-7000 extensible processing platform." http://www.xilinx.com/products/silicon-devices/epp/zynq-7000/index.htm, 2012. [Online; accessed 15-March-2013].

[12] M. Kumm, K. Mller, and P. Zipf:, "Partial lut size analysis in distributed arithmetic fir filters on fpgas," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, 2013.

[13] A. M. Campoy, F. Rodríguez-Ballester, R. Ors, and J. Serrano, "Saving cache memory using a locking cache in real-time systems," in *Proceedings of the 2009 International Conference on Computer Design*, pp. 184–189, jul 2009.

[14] A. M. Campoy, I. Puaut, A. P. Ivars, and J. V. B. Mataix, "Cache contents selection for statically-locked instruction caches: An algorithm comparison," in *Proceedings of the 17th Euromicro Conference on Real-Time Systems*, (Washington, DC, USA), pp. 49–56, IEEE Computer Society, 2005.

[15] A. Shaw, "Reasoning about time in higher-level language software," *Software Engineering, IEEE Transactions on*, vol. 15, pp. 875–889, July 1989.

[16] A. Martí Campoy, A. Perles, F. Rodríguez, and J. V. Busquets-Mataix, "Static use of locking caches vs. dynamic use of locking caches for real-time systems," in *Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on*, vol. 2, pp. 1283–1286 vol.2, May.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Archaeological and Geoscientific Objects used with Integrated Systems and Scientific Supercomputing Resources

Claus-Peter Rückemann
Westfälische Wilhelms-Universität Münster (WWU),
Leibniz Universität Hannover,
North-German Supercomputing Alliance (HLRN), Germany
Email: ruckema@uni-muenster.de

***Abstract*—This paper presents the methods and results from combining Integrated Information and Computing System components with classification for the purpose of enabling multi-disciplinary and dynamical use of information systems and supercomputing resources for Archaeological Information Systems. Focus is on soft criteria, structures, and classification for knowledge discovery for sustainable, long-term knowledge resources. The essential base are a flexible collaboration framework, suitable long-term documentation, structuring and classification of objects, computational algorithms, object representations, and workflows as well as portable application components like Active Source. Case studies of the successful implementation of integration of archaeology and geosciences information and facilitation for dynamical use of High End Computing resources are discussed. The implementation shows how the goal of integrating information and systems resources and advanced scientific computing for multi-disciplinary applications from natural sciences and humanities can be achieved by creating and using long-term knowledge resources.**

Keywords–Integrated Systems; Scientific Supercomputing; Knowledge Resources; Archaeology; Geosciences; Information Systems; Phonetic Algorithms; High Performance Computing.

I. INTRODUCTION

The target of this development is sustainable long-term knowledge resources providing information found by necessarily sophisticated workflows considering content and context. This has to go along with systematically structuring system components and information and describing content and context of objects. With archaeology the objects are commonly handled in a data collection different from the natural sciences objects. The collection and description normally shares no geoscientific, physical, and secondary data, e.g., from natural sciences.

Technology and components are used for digital library components, classification of objects, and realia. Nevertheless, it is important for many use cases in geosciences and archaeology to enable a dynamical use of Integrated Systems and computing resources [1]. In order to overcome many of the complex scientific impediments in prominent disciplines we do need mighty information systems but the more they are used for interactive use they show up needing capabilities for the state-of-the-art in dynamical computing. The studies and implementations of Integrated Information and Computing Systems (IICS) have shown a number of queuing aspects and challenges [2], [3]. In the case if archaeological information systems needed for multi-disciplinary investigation the motivation is the huge potential of integrative benefits and even more pressing that archives are needed for multi-disciplinary records of prehistorical and historical sites while context is often being changed or destroyed by time and development. Besides the academic, industrial, and business application scenarios in focus of the Geo Exploration and Information collaborations (GEXI) [4] in order to integrate the necessary computing facilities with these systems, on the technical side the recent implementations for spatial control problems, e.g., for wildfire control [5], integrating GIS, and parallel computing are promising candidates for future support. This research paper especially contributes to the most important aspect of soft criteria in creating knowledge resources and implementing effective knowledge discovery.

This paper is organised as follows. Sections II and III introduce the basic knowledge resources and the necessary long-term investments. Section IV shows the essential prerequisites of information and structure for the information and computing systems. Sections V and VI describe the results from the development of “silken criteria” and presents examples from phonetic support. Sections VII and VIII show a workflow from these developments and explain the importance of these criteria for the context. Section IX presents the high-level results for the computation and parallelisation with these workflows. Sections X to XIV describe and evaluate the resulting implementation for an Archaeological Integrated Information and Computing Systems and computation results from the components, based on the knowledge resources and digital library examples. Section XV summarises the conclusions and future work.

II. KNOWLEDGE RESOURCES

Knowledge resources provide the universal base for using information and computing resources for a multitude of purposes. They contain systematically gathered, structured and documented content and context on any kind of information, object, sources, and tools.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

This includes systematically structuring system components and information and describing content and context of objects. The architecture and structure enables to use any kind of workflow, e.g., filter stacks using flexible algorithms on different type of content and context. Information and data can be data-mined, analysed, retrieved, and used, e.g., for processing, computing or typesetting by sophisticated workflows considering any qualities or properties of the material. Examples for the material are data sets from natural sciences, documentation texts on multi-disciplinary topics, descriptive texts on humanities, media data, photo documentation on objects, e.g., from digital libraries, and visualised data.

III. Sustainable long-term investments

Although there is some overlap, investments can be categorized in investments for disciplines, services, and resources. For long-term scientific goals, the most significant investment is in knowledge resources. As disciplines have to care for their content and results these may be the investments being closest to the work within disciplines.

Nevertheless, there will be a number of components, e.g., algorithms and applications, which will directly be cared for by disciplines. Services are regulary provided by specialised groups. Computing and storage resources can be provided by various groups, as long as the necessary size and performance is not at the top edge.

All the developments presented in this paper can be considered to be tightly coupled to the knowledge resources, therefore being of close interest for participating disciplines: Silken criteria, parallelisation, workflow and context, information structure, classification, integrated information and computing systems. The investments in the knowledge resources have proved to provide highest sustainability for over twenty-five years now.

IV. Information and structure

It must be emphasised that the complexity of the ecosystem of algorithms and disciplines necessary to achieve an integration of multi-disciplinary information and components is by nature very high so besides the system components we have not only to integrate unstructured but highly structured data with a very complex information structure.

The overall information is widely distributed and it is sometimes very difficult and a long lasting challenge even to get access to a few suitable information sources. The goal for these ambitions is an integrated knowledge base for archaeological geophysics. Example data resources and methods are [6], [7], [8], [9], [10], [11], [12]. For all components presented, the main information, data, and algorithms are provided by the LX Foundation Scientific Resources [13].

Structuring information requires a hierarchical, multilingual and already widely established classification implementing faceted analysis with enumerative scheme features, allowing to build new classes by using relations and grouping. This is synonym to the Universal Decimal Classification (UDC) [14]. In multi-disciplinary object context a faceted classification does provide advantages over enumerative concepts. Composition/decomposition and search strategies do benefit from faceted analysis. It is comprehensive, and flexible extendable. A classification like UDC is necessarily complex but it has proved to be the only means being able to cope with classifying and referring to any kind of object.

V. Silken criteria: Phonetic support

Common means of knowledge exploitation provide string search, precise mathematical algorithms for selections and so on. These are rather sharp with their precision. Even string searches based on regular expressions using advanced means of wildcards are limited in terms of not simply matching the characters but the meaning or context.

For increasing the quality of exploiting knowledge resources we have to build sophisticated means of searching and filtering information objects. For example, if knowledge resources contain more features, these can be used in combination:

- Structure,
- Classification,
- Language distinction,
- Pattern recognition,
- “Sound” recognition, . . .

The entirety of the knowledge resources being part of the LX Foundation Scientific Resources [13] does provide unique means of collective use of features that can be used for knowledge based recognition.

Building applications based on the integrated features, this results in synergy on the one hand and in a much higher Quality of Data (QoD) on the other hand. For example, with search requests, the percentage of information used with the resulting matrix is much higher with integrated features. Standard search includes about 20–50 percent of the available first level information. Integrated search allows to gain up to over 90 percent of suggested information in the first level and about the same for the second and following levels. This does require much higher demands for computation, with most applications even in interactive time range, but this should not be a problem today.

In many of the applications built on knowledge resources, an uncertainty for various attributes is necessary. Algorithms solely being precise as well as those implementing an uncertainty have shown drawbacks when being used for increasing the quality of the results. The solution for many application is to implement sequences of those types of algorithms.

So, with the above mentioned features of the knowledge resources, the individual strengths are that, for example:

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

- Structure can integrate scientific names, e.g., botanical names, with commonly used names whatever they might be matching.
- Pattern recognition can be used to find matching objects on string basis, e.g., from exactly matching character strings.
- Classification can help to find context as well as choosing object or filtering besides any pattern or structure matching, e.g., with UDC codes.
- Language distinction can be used for supporting classification and pattern matching as well as typesetting and hyphenation support, mostly by improving the precision of meaning and context, e.g., generating publishing objects.
- Sound recognition can help find homophones and comparable objects, e.g., searching and selecting additional paths of knowledge discovery to follow in a workflow or filter process.

So, with these resources, even elementary modules for sound and pattern recognition can be of huge benefits when being integrated with the other methods.

VI. Supporting silken selection

The knowledge resources can be used by any algorithm suitable for a defined workflow. One of the available module implementing a silken selection based on the Soundex principle is the `knowledge_sndx_standard` application. The historical Soundex [15] is a phonetic algorithm for indexing names by sound. The goal with this algorithm is to encode homophones so that they can be represented by the same resulting code in order to match persons' name despite differences in writing and spelling [16]. The basic algorithm mainly encodes consonants. Vowels are not encoded unless being a first letter. The U.S. Government is using a modified modern rule set [17] for purposes in census and archives. The original intention was to catch the English pronunciation, anyhow there are many different implementations in use today.

Listing 1 shows a Perl source code used in the `knowledge_sndx_standard` module, modelled after the standard Perl implementation [18], for computing LX Soundex codes [19], being available based on different programming concepts [20], [21]. The various workflows can define and integrate their own Soundex codes for different purposes and topics.

```
#!/usr/bin/perl
#
# knowledge_sndx_standard -- (c) LX Project -- CPR
 1992, 2012
#

$string=$ARGV[0];
$sndx_nocode = undef;

sub knowledge_sndx_standard
{
  local (@s, $f, $fc, $_) = @_;
  push @s, '' unless @s;

  foreach (@s)
  {
    $_ = uc $_;
    tr/A-Z//cd;

    if ($_ eq '')
    {
      $_ = $sndx_nocode;
    }
    else
    {
      ($f) = /^(.)/;
      tr/AEHIOUWYBFPVCGJKQSXZDTLMNR
       /00000000111122222222334556/;
      ($fc) = /^(.)/;
      s/^$fc+//;
      tr///cs;
      tr/0//d;
      $_ = $f . $_ . '000';
      s/^(.{4}).*/$1/;
    }
  }

  wantarray ? @s : shift @s;
}

$code = knowledge_sndx_standard $string;
print ("SNDX-standard:$code:$string\n");

##EOF:
```

Listing 1. LX Soundex SNDX-standard module Perl source code.

The next examples are multi-disciplinary objects from one context, linked by the references in the knowledge resources. If the `SNDX-standard:` prefix is left out in the following examples, the code refers to this standard code.

A. Geology and volcanology

Listing 2 shows some computed LX Soundex codes for the La Soufrière volcano and the reference-internal comparable sound occurrences. The code unifies a number of different versions primarily linked by the prefix but classified by the object classification.

```
L216:La_Soufriere
L216:La_Soufri{`e}re
L216:La_Soufrière
```

Listing 2. SNDX-standard codes for La Soufrière.

The same is true for the following related object. Listing 3 shows computed LX Soundex codes for the Vesuvius and the reference-internal comparable sound occurrences.

```
V210:Vesuv
V210:Vesuvio
V212:Vesuvius
```

Listing 3. SNDX-standard codes for Vesuvius volcano.

Both of these object examples are referring to the volcano database.

B. Archaeology

The archaeological objects, too, very well fit with these algorithms. This is true for a large number of more than 95 percent of classified entries. Listing 4 shows some computed LX Soundex codes for Yucatán and the reference-internal comparable sound occurrences.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

```
Y235:Yucatan
Y235:Yucat'an
Y235:Yucatán
```

Listing 4. SNDX-standard codes for Yucatán.

Listing 5 shows a number of computed LX Soundex codes for Chichén Itzá and the reference-internal comparable sound occurrences.

```
C250:Chichén
C253:Chich'en_Itz'a
C253:Chichen_Itza
C253:Chichén_Itzá
```

Listing 5. SNDX-standard codes for Chichén Itzá.

Listing 6 shows computed LX Soundex codes for Cobá and the reference-internal comparable sound occurrences.

```
C100:Coba
C100:Cob'a
C100:Cobá
```

Listing 6. SNDX-standard codes for Cobá.

C. Biology and botanics

For any of the objects there may be different spellings or even different terms. This means that there are, e.g., botanical names, which are not homophonetically near to the other terms. Listing 7 shows some computed LX Soundex codes for the Chiricote and the reference-internal comparable sound occurrences.

```
G260:Geiger
C623:Chiricote
C623:Ciricote
Z623:Ziricote
C630:Cordia
```

Listing 7. SNDX-standard codes for Chiricote.

The higher variability of codes from the knowledge resources is a good source for calculating new trees for the knowledge discovery workflow.

D. Names and sources

Searching the knowledge resources for "geology, volcanology, and earthquake" delivers a person "Leibniz" in the result Matrix, referring to one of the early statements that volcano activity can result in earthquakes. As the Leibniz object carries a large number of pseudonyms, it can be interesting to follow these as non-explicit references.

An algorithm supports building groups of pseudonyms. Listing 8 shows a computed LX Soundex code for a selection of names used in context with Gottfried Wilhelm Leibniz (1646–1716) and their reference-internal comparable sound occurrences, as computed for the result matrix.

```
SNDX-standard:C260:Caesar
SNDX-standard:C262:Caesarius
SNDX-standard:C265:Caesarinus
SNDX-standard:F612:Freybach
SNDX-standard:F623:Fuerstenerius
SNDX-standard:F623:Fürsteneer
SNDX-standard:F623:Furstenerius
SNDX-standard:F623:Fürstenerius
SNDX-standard:G163:Goffredo
SNDX-standard:G244:Guglielmo
SNDX-standard:G316:Godefridus
SNDX-standard:G316:Godefroy-Guillaume
SNDX-standard:G316:Godfridus
SNDX-standard:G316:Godofredus
SNDX-standard:G316:Godofridus
SNDX-standard:G316:Gotfrid
SNDX-standard:G316:Gotfrids
SNDX-standard:G316:Gothofredus
SNDX-standard:G316:Gotofredus
SNDX-standard:G316:Gottefridus
SNDX-standard:G316:Gottfredus
SNDX-standard:G316:Gottfrid
SNDX-standard:G316:Gottfried
SNDX-standard:G316:Gottofredus
SNDX-standard:G426:Gallo-Graecus
SNDX-standard:G445:Guilelmus
SNDX-standard:G445:Guilielmus
SNDX-standard:G445:Guillielmus
SNDX-standard:G445:Gulielmus
SNDX-standard:G620:Georg
SNDX-standard:G622:Georgius
SNDX-standard:G622:Graecus
SNDX-standard:G655:Germano
SNDX-standard:G655:Germanus
SNDX-standard:J235:Justiniano
SNDX-standard:L152:Leibnics
SNDX-standard:L152:Leibniz
SNDX-standard:L152:Leibnizius
SNDX-standard:L152:Leibnüz
SNDX-standard:L152:Leibnuzius
SNDX-standard:L152:Leibnüzius
SNDX-standard:L153:Laipunitsu
SNDX-standard:L153:Leibnitio
SNDX-standard:L153:Leibnitius
SNDX-standard:L153:Leibnits
SNDX-standard:L153:Leibnitz
SNDX-standard:L153:Leibnitzius
SNDX-standard:L153:Leibnütz
SNDX-standard:L215:Lajbnic
SNDX-standard:L215:Lejbnic
SNDX-standard:L315:Lithvanus
SNDX-standard:L352:Lithuanus
SNDX-standard:R114:Republicanus
SNDX-standard:R153:Raibunittsu
SNDX-standard:S125:Sibisimilis
SNDX-standard:S516:Semper
SNDX-standard:U421:Ulicovius
SNDX-standard:V445:Vilhelm
SNDX-standard:V632:Veridicus
SNDX-standard:W445:Wilhelm
```

Listing 8. LX SNDX-standard codes for "Leibniz" pseudonym parts.

The result shows that the name-Soundex algorithm delivers several phonetical groups. Distinction criteria for modelling the results can be based on considering knowledge resources' structure, attributes, and features, e.g., language, topic context, and name-string order.

Here, the most frequent groups are G316, G244, G163, G445, W445, L152, L153, L215. On the one hand, these obviously correspond with different spellings of the real name. On the other hand, pseudonym name parts are especially carrying codes as C260, C262, C265, F612, F623, G622, G426, G655, J235, L315, L352, R114, R153, S125, S516, U421, V632. Further, if necessary for a workflow, it is as well possible to handle phonetical variances and pseudonym names separately, oeven with separate phonetical algorithms.

Listing 9 shows some essential modifications for the SNDX-latin module `knowledge_sndx_latin` com-

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

pared to the SNDX-standard (Listing 1), to be used with these groups of objects.

```
1       tr/AEHIOUWYBFPVCGJKQSXZDTLMNR
         /00000000111122022222324556/;
```

Listing 9. LX Soundex SNDX-latin modification for SNDX-standard.

Listing 10 shows a computed LX Soundex code for an excerpt selection as above but with the SNDX-latin module.

```
1  SNDX-latin:L152:Laipunitsu
2  SNDX-latin:L152:Lajbnic
3  SNDX-latin:L152:Leibnics
4  SNDX-latin:L152:Leibnitio
5  SNDX-latin:L152:Leibnitius
6  SNDX-latin:L152:Leibnits
7  SNDX-latin:L152:Leibnitz
8  SNDX-latin:L152:Leibnitzius
9  SNDX-latin:L152:Leibniz
10 SNDX-latin:L152:Leibnizius
11 SNDX-latin:L152:Leibnütz
12 SNDX-latin:L152:Leibnüz
13 SNDX-latin:L152:Leibnuzius
14 SNDX-latin:L152:Leibnüzius
15 SNDX-latin:L152:Lejbnic
```

Listing 10. LX SNDX-latin codes for "Leibniz" pseudonym name parts (excerpt) showing the harmonised codes.

The newly created algorithm has harmonised the codes L152, L153, L215 for the "Leibniz"-object regarding 'z' and 't' as well as 'i' and 'j' to become SNDX-latin:L152. In order to benefit from the improvements with algorithms, objects can carry any references to these algorithms. For the disciplines creating the content and references it is important not only to see the result matrix but also the reasons for the codes and ranking and to be able the modify the source codes with any objects.

VII. Workflow and silken criteria

The workflow for applying these algorithms for an enriched result matrix is as follows:

1) Object search using string and classification criteria on the knowledge resources and references results in primary result matrix.
2) Object search using smooth, silken criteria, e.g., Soundex, on attribute- selected content in the primary result matrix results in secondary result matrix.
3) References to object from the secondary result matrix are used to search objects from the knowledge base and references in order to create a tertiary result matrix.
4) The tertiary result matrix is integrated with objects from all steps and a defined ranking is used to create the final result matrix.

Methods include the structure of objects, language attribute, transliterations, transcriptions, synonyms, references and so on. In most cases these features are precisely defined. The silken support is provided by an algorithm defined for and by the user application within the scenario. This algorithm, by concept, is designed to enable a use case specific implementation.

VIII. Workflow and context

In the regular expression - knowledge resources workflow (workflow 1), the result will be based on the chain "*Volcano - Vesuvius - Vesuv*" (workflow 1 result chain). In the context regular expression - knowledge resources - phonetic algorithms - language attributes - context categorisation - references - sources/material (workflow 2), the knowledge resources workflow resembles results based on a chain of "*Volcano - Vesuvius - Vesuv - Leibniz - terrae motus - letter/communication - Vesuvium - Fumarole - Solfatara*" (workflow 2 result chain).

The first connections can be found by structure, references, and regular expressions. The various Leibniz information and references in the second workflow have solely been found by phonetic algorithms. The references from English to Latin or German content has solely been possible by language attributes. In order to find further information for the result matrix even these methods would not be sufficient. Thus, the terrae motus path has been recognised using context categorisation, e.g., context keywords. With a sophisticated combination of these methods new references and new links to sources and material could be found for an improved result matrix. In this example, the term "terrae motus" has been one of the keys opening up a multitude of further information.

Material in specialised collections, for example in the European Cultural Heritage Online [22] would not be accessible due to the type and context of the material.

In the above workflow, within the chain from the stage "Leibniz" on, the content of archaeology and geosciences will not be accessible, for example the communication regarding volcanoes, earthquakes, and caves in manuscripts and letters or content of pictorial objects are not available via search engines. In this example, there is a rich contribution for the result matrix on volcanism, volcanology, and geology by various historical objects, references, and sources, especially for volcanism, Vesuvius [23], as well as earthquake related context [24], even from concept glossaries [25], manuscript collections and catalogues [26], [27] as, e.g., [28], [29], or Leibniz related copperplates [30]. For example, the "praehistoric unicorn" reconstruction [31], as well as material on geological context has not been referenced before from the objects of the knowledge resources and is not freely and publicly available as a direct reference, media or verification [32].

Therefore, with conventional search concepts, the content and any information from it will be missed within the workflow and any information will not contribute to the result matrix. Reasons for these misses can, e.g., be historical language, type of material, licensing, property and access rights. All of these being at least as important as the technical issues. Using the available features, e.g., the context categorisation from the knowledge resources it is

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

possible to catch this information and to drastically increase the spectrum of gathering information and complementing the result matrix. The workflows and algorithms presented here can be used in order to overcome missing links in between different information pools.

Listing 11 shows an excerpt from the keyword context data of an 'Leibniz'-object.

```
...
keyword-Context: KYW :: Leibniz, Korrespondent, Tschirnhaus
keyword-Context: TXT :: Venedig, Neapolis, Puzzolo, Grotta del Cane
keyword-Context: TXT :: Neapolis, welches nach Rom und Venedig eine der schönsten städten Italiae ist
keyword-Context: TXT :: schwöfel bäder, schweffel
keyword-Context: KYW :: Schwefel, Solfatara, Fumarole
keyword-Context: TXT :: Neapolis, den brennenden Berg Vesuvium
keyword-Context: TXT :: Grotta del Cane
keyword-Context: TXT :: Neapolis, den brennenden Berg Vesuvium
keyword-Context: KYW DE :: Vulkanismus, Vulkanologie, Vesuv, Vesuvius, Vesuvium, Erdbeben, Beben
keyword-Context: KYW EN :: volcanism, volcanology, Vesuvius, Vesuvium, earthquake, quake
...
link-Context: LNK :: http://www.gwlb.de/Leibniz/Leibnizarchiv/Veroeffentlichungen/III7B.pdf
keyword-Context: TXT :: terrae motu, Sicilien
keyword-Context: KYW :: Erdbewegungen, Erdbeben, Vulkane, terrae motu, terra motus, Sicilien, Sizilien
...
link-Context: LNK :: http://echo.mpiwg-berlin.mpg.de
keyword-Context: KYW DE :: Nicolaus Seelaender, Nicolaus Seeländer, Kupferplatten, Leibniz, Leibniz Einhorn, Einhornhöhle b. Scharzfeld im Harz
...
link-Context: LNK :: http://www.leibnizcentral.de/CiXbase/gwlbhss/
keyword-Context: TXT :: 1631/1632 16xx, terra motus, fogelius
keyword-Context: KYW DE :: Erdbeben, Seismologie, Seismik, Fogel, Fogelius, Vulkan, Vesuvius, CiXbase, cixbase
keyword-Context: KYW EN :: earthquake, seismology, seismics, Fogel, Fogelius, volcano, Vesuvius, CiXbase, cixbase
...
link-Context: LNK :: http://www.leibnizcentral.com
keyword-Context: KYW DE :: Vulkan, Erdbeben, Seismologie
keyword-Context: KYW EN :: volcano, earthquake, seismology
```

Listing 11. Keyword context data from a 'Leibniz'-object (excerpt).

Listing 12 shows an excerpt from the keyword context data of two cave objects, which are referenced from the above object.

```
link-Context: LNK :: http://echo.mpiwg-berlin.mpg.de/content/copperplates
keywords-Context: KYW :: Leibniz, Nicolaus Seeländer, Kupferstichplatte, Copperplate, Baumannshöhle
link-Context: LNK :: http://echo.mpiwg-berlin.mpg.de/content/copperplates
keywords-Context: KYW :: Leibniz, Nicolaus Seeländer, Kupferstichplatte, Copperplate, Einhornhöhle, Harz
```

Listing 12. Keyword context data from cave objects (excerpt).

For finding these, the context descriptions have been evaluated [30], [22]. An example for the context description for one of these is shown in Listing 13.

```
Kupferstichplatten
Titel:  K 220 Einhorn und versteinerter Zahn
Beschriftung:   Tab. XII; Dens animalis marini Tidae prope Stederburgum e colle limoso effossi. Figura Sceleti prope Qvedlinburgum effossi.
Stecher:        Seeländer [signiert: N. Seelaender sc.]
Format:         318x196 mm
Bemerkung:      Abzug unter cua stark beschädigt. - Liste 1727, Nr. 23; Liste 1729a, Nr. 10. Abzug (ohne Tafelnummer) auch in Noviss. 56: IV,3, Bl. 12. Lt. Manuskript XXIII, 23b, Bl. 57' u. 57a, sollte dies ursprünglich Tafel X sein.
Abdruck:        Leibniz, Protogaea, Taf. XII, Text dazu S. 64 [über den Fund bei Quedlinburg]:Testis rei est Otto Gerikius, Magdeburgensis Consul, qui nostram aetatatem novis inventis illustravit [...] Gerikius igitur libro de vacuo edito, per occasionem narrat, repertum Sceleton unicornis in posteriore corporis parte, ut bruta solent, reclinatum, capite vero sursum levato, ante frontem gerens longe extensum cornu quinque fere ulnarum, crassitie cruris humani, sed proportione quadam decrescens. Ignorantia fossorum contritum particulatimque extractum est, postremo cornu cum capite et aliquibus costis, et spina dorsi atque ossibus Principi Abbatissae loci allata fuere. Eadem ad me perscripta sunt; additaque est figura, quam subiicere non alienum erit. [Zusatz im Manuskript, nicht im Druck:] Simile ingens animal Tidae prope Stederburgum nuper repertum est in monte a limo de cujus quodam immania ossa apud me sunt.  ;
Nachgestaltung:         Nachstich in Leibniz, Opera omnia, studio L. Dutens, 1768. - Wallmann, Abhandlung von den schätzbaren Alterthümern zu Quedlinburg, 1776, Tafel S. 39.  ;
Literatur:      Achim Rost, Das fabelhafte Einhorn. In: Die Welt im leeren Raum, 2002, S. 120-132. Vgl. dort auch S. 376 u. 378.
Signatur:       cup 4048
Signatur(Abzug):        cua 3203
```

Listing 13. Example for evaluated context description.

IX. Computation and parallelisation

The computation time for about 100000 objects is about 20 seconds on one processor. As per request it is necessary to have several runs, for several references, this add up to about 10 minutes even for a simple object if done linear. Most of these processes can be done in parallel but due to the complexity of the knowledge content and the flexibility implemented thereof implemented for the knowledge resources it is not possible to have a general algorithm and type of parallelisation. The basic types of workflows used with object extraction are:

1) Linear workflows do not benefit from parallelisation inside the workflow. However, if a large number of comparable operations have to be executed, the overall application will benefit from a more or less loosely coupled parallelisation of these operations. The efficiency depends on the application using the results and triggering the events for the operations.
2) Parallel workflows can benefit from a parallelisation inside the workflow. This can, for example, result from operations inside the workflow that have to use persistent as well as volatile information processing. A simple case is a workflow based on a regular pattern expression on classified object groups using

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

homophones for finding additional object identities. In this case, the phonetic calculations can be done "on the fly", for finding the homophones in parallel for all objects as soon as they are delivered by the regular expression pattern search.

3) Partially parallel workflows will combine both linear and parallel sequences in their workflow.

Therefore, the degree of parallelisation depends on the height of the level of the implementation. The integration of knowledge resource structure, classification, and algorithms does provide large benefits on the result matrix:

- Long-term sustainable knowledge base,
- Improved Quality of Results,
- Improved Quality of Data,
- Maximum flexibility.

X. Integrated information and computing

The integration issues of information, communication, and computing are well understood [2], [33], [34] from the "collaboration house" framework [1] integrating information and scientific computing.

A. Collaboration and multi-disciplinary workflow

Based on the collaboration framework the IICS enables to collaborate on disciplines, services, and resources and operational level. It allows disciplines to participate on multi-disciplinary topics for building Information Systems and to use scientific supercomputing resources for computing, processing, and storage, even with interactive and dynamical components [35]. The screenshot (Figure 1) illustrates some features, as with Active Source, computed and filtered views, LX information, and aerial site photographs, e.g., from Google Maps. Many general aspects of dynamical use of information systems and scientific computing have been analysed with the collaboration house case studies.

B. Integrative and synergetic effects

With IICS we do have integrative as well as synergetic effects from the participating disciplines. For example, the Roman city of Altinum, next to Venice, Italy, would not have been re-discovered without the combination of archaeological information, aerial photographs, satellite images, and digital terrain models [36]. Even in unorganised circumstances, like with this discovery, the multi-disciplinary cooperation can lead to success. The more we need an integrated information system approach for "disciplines on demand" in order to improve the collaboration and the sustainability of results.

On the other hand we have synergetic effects with the same scenario of archaeology and geosciences, too, the research does have benefits for archaeology and geosciences as the collection of information from archaeological probing will help to describe the underground, which is of immense importance for the future of the area [37] and its attractiveness [38].

XI. Archaeological Information Systems

Anyway, there should be a principle solution, considering the hardware and software if so individually available, without restructuring complex data all the time when migrating to different architectures or to be prepared for future resources.

A. Archaeology and geosciences

So, in case of Archaeological Information Systems (AIS), for advanced Archaeological IICS, cultural heritage, and geoscientific information, and computing systems, there is a strong need for integration and documentation of different data and information with advanced scientific computing, e.g., but not limited to:

- Object, site, artifact, spatial, multi-medial, photographical, textual, properties, sources, referencial information.
- Landscape and environmental information, spatial, photographical information.
- Geophysical information, geological information.
- Event information.

Important aspects with all this information are the distribution analysis and spatial mapping. With dynamical information systems for this scenario the components must enable to weave n-dimensional topics in time, use archaeological information in education, implement n-dimensional documentation, integrate sketch mapping, provide support by multi-disciplinary referencing and documentation, discovery planning, structural analysis, multi-medial referencing.

B. Creating metadata for documentation and computing

It will need a number of metadata types, depending from the variable type of content, describing all kind of relevant information regarding the data and the use of this data [39]. Some important groups are category, source, batch-System, OS version and implementation, libraries, information on conversion, virtualisation environment, and automation.

Currently only a few projects in some disciplines have worked on long-term content issues [40], [41], [42], [43], [44]. Commonly only three categories are relevant to archaeological projects, project level metadata (e.g., keywords, site, dates, project information, geodata), descriptive and resource level metadata (e.g., comprehensive description, documents, databases, geo-data), and file level metadata (software, hardware, accompanying files). As we saw above, from information science point of view this is by far not sufficient as there are, e.g., licensing and archiving restrictions, precision restrictions, network limitations, context of environment, hardware, and software, hardware restrictions, tools and library limitations and implementation specifics.

The long-term aspects for big heterogeneous data hold very difficult and complex challenges as big data storage facilities [45], for users there are, e.g., free public access and long-term operational issues, for context provisioning huge amount of work have to be done, e.g., handling licensing, archiving, context, hardware availability and many more.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

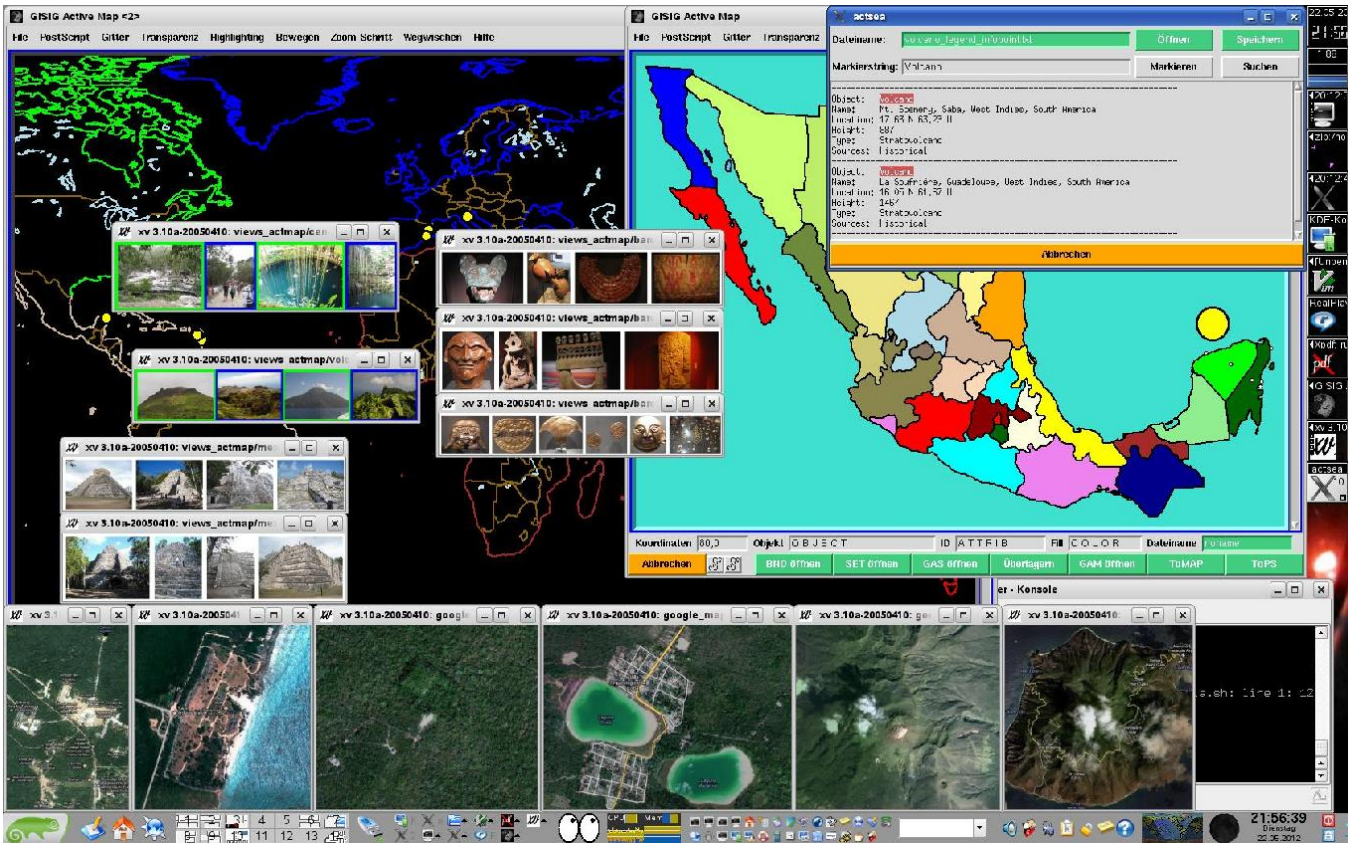


Figure 1. Dynamical use of information systems and scientific computing with multi-disciplinary and universal knowledge resources [1].

XII. IMPLEMENTATION OF COMPONENTS

A. Targets and means

The main target categories and means of information to be addressed are interdisciplinary, multi-disciplinary, intercultural, functional, application, and context information. The main functional targets with IICS are integrative knowledge, education, technological glue, linking isolated samples and knowledge databases, language and transcription databases, classified Points on Interest (POI), InfoPoints, multimedial information. The organisational means are commonly grouped in disciplines, services, resources and operation.

B. Information sources

All media objects used here with components and views are provided via the Archaeology Planet and Geoscience Planet components [13]. The related information, all data, and algorithm objects presented are copyright the LX Foundation Scientific Resources [13]. It provides multi-disciplinary information and data with its knowledge resources, e.g., for archaeology, geophysics, geology, environmental sciences, geoscientific processing, geoprocessing, Information Systems, philology, informatics, computing, geoinformatics, cartography.

C. Information, structure and classification

The following examples illustrate the retrieved object information, media, and sources with examples for their multi-disciplinary relations. The information is retrieved from the LX Foundation Scientific Resources [13], [2], [46] and categorised with means like UDC. Listing 14 shows an excerpt of a LX object entry used with IICS.

```
1 Cenote Sagrado [Geology, Spelaeology, Archaeology]:
2                Cenote, Yucatán, México.
3                Holy cenote in the area of Chichén Itzá.
4                ...
5                %%UDC:[55+56+911.2]:[902+903+904]:
                  [25+930.85]"63"(7+23+24)=84/=88
6                %%Location: 20.687652,-88.567674
7                Syn.: Cenote Sagrada
8                s. also Cenote, Chichén Itzá
```

Listing 14. Structure of object entry (LX Resources, excerpt).

Listing 15 shows a classification set of UDC samples used with the knowledge resources and IICS.

```
1 UDC:[902+903+904]:[25+930.85]"63"(7)(093)=84/=88
2 UDC:[902+903+904]:[930.85]"63"(23)(7):(4)=84/=88
3 UDC:[55+56+911.2]:[902+903+904]:[25+930.85]"63"
   (7+23+24)=84/=88
4 UDC:[25+930.85]:[902]"63"(7)(093)=84/=88
5 UDC:[911.2+55+56]:[57+930.85]:[902+903+904]"63"
   (7+23+24)=84/=88
6 UDC:[911.2+55]:[57+930.85]:[902]"63"(7+23+24)=84/=88
```

Listing 15. Classification set (UDC samples, excerpt).

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The classification deployed for documentation [47] must be able to describe any object with any relation, structure, and level of detail. Objects include any media, textual documents, illustrations, photos, maps, videos, sound recordings, as well as realia, physical objects such as museum objects. A suitable background classification is, e.g., the UDC. The objects use preliminary classifications for multi-disciplinary content. Standardised operations with UDC are, e.g., addition ("+"), consecutive extension ("/"), relation (":"), subgrouping ("[]"), non-UDC notation ("*"), alphabetic extension ("A-Z"), besides place, time, nationality, language, form, and characteristics.

D. Communication and computing

The central component groups for bringing multi-disciplinary information systems into practice are IICS and documentation of objects, structure, and references. Listing 16 shows an example of a dynamical dataset from an Active Source [35] component provisioning information services.

```
#BCMT-------------------------------------------------
###EN \gisigsnip{Object Data: Country Mexico}
#ECMT-------------------------------------------------
proc create_country_mexico {} {
global w
$w create polygon 0.938583i 0.354331i 2.055118i ...
...
proc create_country_mexico_autoevents {} {
global w
$w bind legend_infopoint <Any-Enter> {set killatleave [
 exec ./mexico_legend_infopoint_viewall.sh $op_parallel
  ] }
$w bind legend_infopoint <Any-Leave> {exec ./
 mexico_legend_infopoint_kaxv.sh }
$w bind tulum <Any-Enter> {set killatleave [exec
 $appl_image_viewer -geometry +800+400 ./
 mexico_site_name_tulum_temple.jpg $op_parallel ] }
$w bind tulum <Any-Leave> {exec kill -9 $killatleave }
} ...
```

Listing 16. Dynamical data set of Active Source component.

Batch and interactive features are integrated with Active Source event management [35], e.g., allowing structure and UDC based filtering. Computing interfaces can carry any interactive or batch job description. Taking a look onto different batch and scheduling environments one can see large differences in capabilities, handling environments and architectures. In the last years, experiences have been gained in simple features for different environments for High Throughput Computing like Condor, workload schedulers like LoadLeveler and Grid Engine, and batch environments like Moab / Torque.

XIII. Resulting implementation in practice

A. Scientific documentation

Scientific documentation is an essential part of a Universal IICS (UIICS), revealing associations and relations and gaining new insight. Handling the available information does provide transparent how puzzle pieces of a scientific context do fit, e.g., not only that terms like Bronze Age, Ice Age, Stone Age are only regional but in quantity and quality how the transitions and distributions in space and time are. Information on objects, archiving, analysis, documentation, sources and so on will be provided as available with the dimension space. Besides the dynamical features the objects carry information, e.g., references, links, tags, and activities.

B. Dimension space

The information matrix spans a multi-dimensional space (Table I). It illustrates the multi-faceted topic dimension containing important cognitive information for disciplines and applications. Examples of multi-disciplinary information in archaeological context are stony and mineral composition, e.g., of dead freight or ballast in ship wrecks, mineral material in teeth, fingerprints of metals used in artifacts, and genetic material of biological remains. Further there exists a "vertical" multi-dimensional space to this information matrix, carrying complementary information, e.g., color, pattern, material, form, sound, letters, characters, writing, and so on. The documentation can handle the holistic multi-dimensional space, so we can flatten the views with available interfaces to three or four dimensional representations.

Table I
Dimensions of the information matrix (excerpt).

Dimension	*Meaning, Examples*
Time	Chronology
Topic	Disciplines
	Purpose (tools, pottery, weapons, technology, architecture, inscriptions, sculpture, jewellery)
	Culture (civilisation, ethnology, groups, etymology)
	Infrastructure (streets, pathways, routes)
	Environment (land, sea, geology, volcanology, speleology, hydrogeology, astronomy, physics, climatology)
	Genealogy (historical, mythological documentation)
	Genetics (relationship, migration, human, plants)
	Biology (plants, agriculture, microorganisms)
	Trade (mobility, cultural contacts, travel)
Depth	Underground, subterranean
Site	Areal distribution, region
...	...
Data	Resources level, virtualisation

The dimensions are not layers in any way so it would contradict to percept their documentation with integrated systems in data or software layers. With these IICS we are facing a multi-dimensional volume, like multi-dimensional "potato shapes" of knowledge objects. Layer concepts are often used with cartographic or mapping applications but these products are infeasible for handling complex cognitive context.

C. IICS dimension view

As with the structure the communication and compute processes are getting resource intensive, the available storage and compute resources are used with the IICS. The following small example shows an excerpt of a tabulated dimension view (Table II). The last column shows if an object is deposited on site (O) or distributed (D) and if additional media

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

is available and referenced. The table shows if a storage or and additional compute request has been necessary for the resulting object or media. Information is given if primarily a storage request (S) for persistant media or a compute request (C) deploying High End Computing resources is dynamically used for creating the appropriate information.

Table II
DIMENSION VIEW WITH ARCHAEOLOGICAL IICS (EXCERPT).

Topic	*Purpose / Environment / Infrastructure*	*Ref.*
Egypt	Architecture	
Rome	Architecture	
Catalonia	Architecture	
	Monument de Colom, Port, Barcelona, Spain	OC
Maya	Architecture	
	Kukulkán Pyramid, Chichén Itzá, Yucatán, México	OC
	Nohoch Mul Pyramid, Cobá, Yucatán, México	OC
	El Meco Pyramid, Yucatán, México	OC
	El Rey Pyramid, Cancún, Yucatán, México	OC
	Pelote area, Cobá, Yucatán, México	OS
	Pok ta Pok, Cancún, Yucatán, México	OS
	Templo del Alacran, Cancún, Yucatán, México	OS
	Port, Tulúm, Yucatán, México	OC
	Infrastructure	
	Sacbé, Chichén Itzá, Yucatán, México	OS
	Sculpture	
	Diving God & T. Pinturas, Tulúm, Yucatán, México	OC
	Diving God, Cobá, Yucatán, México	OC
Precolombian	Architecture	
Caribbean	Environment (volcanology, geology, hydrogeology)	
	La Soufrière volcano, Guadeloupe, F.W.I.	OC
	Mt. Scenery volcano, Saba, D.W.I.	OC
	Cenote Sagrado, Chichén Itzá, Yucatán, México	OC
	Ik Kil Cenote, Yucatán, México	OC
Arawak	Architecture	
Prehistory	Architecture	
Topic:	architecture mythology environment infrastructure	
Entity:	Object Location: O On site, D Distributed; Object Media: C Compute, S Storage.	
Compute:	CONNECT REFERTO-TOPIC REFERTO-SPATIAL VIEW-TO VIEW-FROM	

The following examples explain views from disciplines and topics (Figure 1) as computed and filtered with the IICS, using photo media samples (media samples © C.-P. Rückemann, 2012, 2013). It must be emphasised that the applications can provide any type of objects, high resolution media, and detailed information. The first view (Figure 2) is a simple example from the above table for an excerpt of the computed class of regional pyramid object representations (Yucatán Peninsula, provinces Yucatán and Quintana Roo).

Figure 2. Object SAMPLE – regional pyramid of Maya, Yucatán, México.

Figure 3 illustrates the computed objects for the above REFERTO-TOPIC and REFERTO-SPACE chain classification, e.g., here via UDC "(7):(4)" relation.

Figure 3. Cross-purpose REFERTO – Diving god, Tulúm, Colom.

Besides that, viewing directions can be referred, e.g., "view to", "view from", "detail" as shown with a VIEW example (Figure 4) for the above selection with UDC "(23)", "(24)".

Figure 4. In-purpose: VIEW-TO VIEW-FROM – Volcanoes and Cenotes.

D. Topic view and object representation

The following sample excerpt tabulates a topic view (Table III) and shows the computed object representation (Figure 5) for an in-topic CONNECT example. From the eight samples of Chichén Itzá shown, the Sacbé pathway connects the Kukulkán Pyramid with the Cenote Sagrado. The table shows a sample of referred (Geo) information.

Table III
TOPIC VIEW WITH ARCHAEOLOGICAL IICS (EXAMPLE, CHICHÉN ITZÁ).

Site	*Topic / Purpose*	*Selected: Geo*	*Ref*
Chichén Itzá			
	Kukulkán Pyramid, El Castillo	Limestone	OC
	Sacbé	Limestone	OC
	Cenote Sagrado	Doline, hydrology	OC
	Jaguar temple		OS
	Tzompantli		OS
	Temple of the warriors		OS
	Caracol		OS
	Chac temple		OS

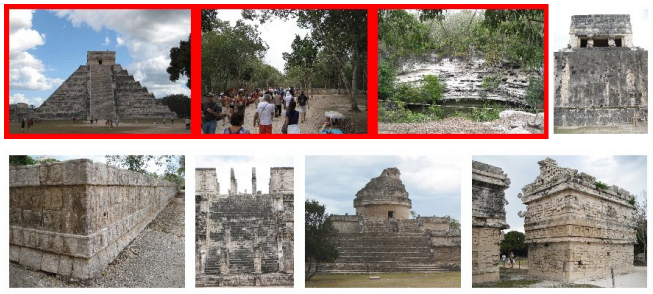


Figure 5. In-topic CONNECT – Kukulkán, Cenote, connected by Sacbé.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

As Figure 1 showed, the objects resulting from the computation can contain any additional attributes, e.g., geo-referenced relations for further application within spatial context or multi-disciplinary analysis and evaluation.

E. Object space grouping

The objects are linked by relations in the n-dimensional object space. The slices with a selected number of dimensions carry the common information, e.g., "Stone Age flint arrow heads" in a specific area. It is essential not to sort objects into layers within a database-like structure. So vectors and relations can help to represent their nature in a more natural way. The views, even traditional layered ones, are created from these by appropriate components. The following figures illustrate structure and references for collections, context, and integration of multi-disciplinary information: museum topical collection (Figure 6), context of amphores (Figure 7), and geology information (Figure 8).

Figure 6. Sample COLLECTION – Precolombian Museum.

Figure 7. Sample CONTEXT – Pottery (amphores).

Figure 8. Sample DISCIPLINE – Geology (Caribbean limestone and tuff).

XIV. Digital Archaeological Library Examples

In combination with the above shown features, objects in digital archaeological libraries have been enriched with various information, e.g., on museum, library information, archives, network information, mapping services, locations and Points Of Interest (POI).

Due to the knowledge resources organisation, the objects can be used in references as well as in the cache for interactive components at any stage within the workflow process. Combining the structure and classification with the silken selection algorithms leads to very flexible, multi-disciplinary interfaces.

Each group of digital images shows the result matrix from a selection process. The following figures illustrate resulting objects from the digital library of the LX knowledge resources with multi-disciplinary background, in these examples regarding material, function, and model or reconstruction purposes.

Selecting archaeological objects from Central and Southern America, ancient art, and consisting of Gold (`UDC:902+(7),(8)+700.32+546.59`) results in a subset from the gold objects collection (Figure 9).

Figure 9. Sample COLLECTION – Jewelery + material: gold.

Selecting archaeological objects, ancient art, and being part of the collier collection (`UDC:902+700.32`) results in a subset from the jewelery collection (Figure 10).

Figure 10. Sample COLLECTION – Jewelery + function: collier.

The result shows, that objects can be member of any number of collections and result matrices, as compared to the result from the museum topical collection (Figure 6). Selecting archaeological objects, watercraft engineering, marine engineering, boats, ships, boat building, ship building, and being models having origin from ancient Egypt and the Mediterranean (`UDC:902+629.5+(32),(37),(38)`) results in a subset from the ship model collection (Figure 11).

Figure 11. Sample COLLECTION – Ship + type: model.

Any silken criteria can be used for the transliteration, transcriptions and other content and context. So if not limited to fixed criteria, the resulting matrix is highly dynamical and supports a flexible modelling of the relations and explicit and implicit references within the available material.

XV. Conclusion and Future Work

It has been shown how long-term knowledge resources have been created and used for more than twenty-five years considering content and context with sophisticated workflows implementing various technology over the years.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The knowledge resources have proven to provide a universal way of describing multi-disciplinary objects, expressing relations between any kind of objects and data, e.g., from archaeology, geosciences, and natural sciences as well as defining workflows for calculation and computation for application components. Systematically structuring, classification, as well as soft 'silken' criteria with LX and UDC support have provided efficient and economic means for using Information System components and supercomputing resources. With these, the solution scales, e.g., regarding references, resolution, and view arrangements even with big data scenarios and parallel computing resources. The concept can be transferred to numerous applications in a very flexible way and has shown to be most sustainable.

The successful integration of IICS components and advanced scientific computing based on structured information and faceted classification of objects has provided a very flexible and extensible solution for the implementation of Archaeological Information Systems.

It has been demonstrated with the case studies that Archaeological IICS can provide advanced multi-disciplinary information as from archaeology and geosciences by means of High End Computing resources.

The basic architecture has been created using the collaboration house framework, long-term documentation and classification of objects, flexible algorithms, workflows and Active Source components. As shown with the examples, any kind of computing request, e.g., discovery, data retrieval, visualisation, and processing, can be done from the application components accessing the knowledge resources. Computing interfaces can carry any interactive or batch job description. Anyhow, the hardware and system resources have to be configured appropriately for a use with the workflow. For future applications a kind of "tooth system" for long-term documentation and algorithms for use with IICS and the exploitation of supercomputing resources will be developed. Besides this, it is intended to further extend the content spectrum of the knowledge resources.

Acknowledgements

We are grateful to all national and international academic, industry, and business partners in the GEXI cooperations and the Science and High Performance Supercomputing Centre (SHPSC) for long-term support of collaborative research and the LX Project for providing suitable resources. Many thanks to the scientific colleagues at the Leibniz Universität Hannover, the Institute for Legal Informatics (IRI), and the Westfälische Wilhelms-Universität (WWU), sharing experiences on ZIV, HLRN, Grid, and Cloud resources and for participating in fruitful case studies as well as the participants of the INFOCOMP and DigitalWorld conferences as well as the postgraduate European Legal Informatics Study Programme (EULISP) for prolific discussion of scientific, legal, and technical aspects over the last years.

Thanks for excellent inspiration, support, and photo scenery go to the Saba Conservation Foundation, Saba Marine Park, and National Heritage Foundation St. Maarten (D.W.I.), National Park Guadeloupe and Museum St. Martin (F.W.I.), Instituto Nacional de Antroplogía e Historia (I.N.A.H.), Mexicó for providing access to the sites of Chichén Itzá, Cobá, Tulúm, El Meco, El Rey, and many more as well as to the Eco-Parc Xel Há, México, and especially to Ms. Maureen Felix (Consejo de Promoción Turística de México, CPTM) for her excellent support, the Museu Barbier-Mueller d'Art Precolombí and Museu Egipci Barcelona, Museu Urbana, Valencia, Spain, as well as Canon for the photo equipment.

References

[1] C.-P. Rückemann, "Enabling Dynamical Use of Integrated Systems and Scientific Supercomputing Resources for Archaeological Information Systems," in *Proceedings of the International Conference on Advanced Communications and Computation (INFOCOMP 2012), October 21–26, 2012, Venice, Italy*. XPS, Xpert Publishing Services, 2012, pp. 36–41, Rückemann, C.-P. and Dini, P. and Hommel, W. and Pankowska, M. and Schubert, L. (eds.), ISBN: 978-1-61208-226-4, URL: http://www.thinkmind.org/download.php?articleid=infocomp_2012_3_10_10012 [accessed: 2013-06-09].

[2] C.-P. Rückemann, *Queueing Aspects of Integrated Information and Computing Systems in Geosciences and Natural Sciences*. InTech, 2011, pp. 1–26, Chapter 1, in: Advances in Data, Methods, Models and Their Applications in Geoscience, 336 pages, ISBN-13: 978-953-307-737-6, DOI: 10.5772/29337, OCLC: 793915638, DOI: http://dx.doi.org/10.5772/29337 [accessed: 2013-05-26].

[3] C.-P. Rückemann, "Implementation of Integrated Systems and Resources for Information and Computing," in *Proceedings of the International Conference on Advanced Communications and Computation (INFOCOMP 2011), October 23–29, 2011, Barcelona, Spain*, 2011, pp. 1–7, ISBN: 978-1-61208-009-3, URL: http://www.thinkmind.org/download.php?articleid=infocomp_2011_1_10_10002 [accessed: 2013-05-26].

[4] "Geo Exploration and Information (GEXI)," 1996, 1999, 2010, 2013, URL: http://www.user.uni-hannover.de/cpr/x/rprojs/en/index.html#GEXI (Information) [accessed: 2013-05-26].

[5] L. Yin, S.-L. Shaw, D. Wang, E. A. Carr, M. W. Berry, L. J. Gross, and E. J. Comiskey, "A framework of integrating GIS and parallel computing for spatial control problems - a case study of wildfire control," *IJGIS,* ISSN: 1365-8816, DOI: 10.1080/13658816.2011.609487, pp. 1–21, 2011.

[6] National Park Service, "National Register of Historic Places Official Website, Part of the National Park Service (NPS)," 2013, NPS, URL: http://www.nps.gov/nr [accessed: 2013-05-26].

[7] "North American Database of Archaeological Geophysics (NADAG)," 2013, University of Arkansas, URL: http://www.cast.uark.edu/nadag/ [accessed: 2013-05-26].

[8] "Center for Advanced Spatial Technologies (CAST)," 2013, University of Arkansas, URL: http://www.cast.uark.edu/ [accessed: 2013-05-26].

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[9] "Archaeology Data Service (ADS)," 2013, URL: http://archaeologydataservice.ac.uk/ [accessed: 2013-05-26].

[10] "Center for Digital Antiquity," 2013, Arizona State Univ., URL: http://www.digitalantiquity.org/ [accessed: 2013-05-26].

[11] "The Digital Archaeological Record (tDAR)," 2013, URL: http://www.tdar.org [accessed: 2013-05-26].

[12] IBM, "City Government and IBM Close Partnership to Make Rio de Janeiro a Smarter City," *IBM News room - 2010-12-27, USA*, 2012, URL: http://www-03.ibm.com/press/us/en/pressrelease/33303.wss [accessed: 2012-03-18].

[13] "LX-Project," 2013, URL: http://www.user.uni-hannover.de/cpr/x/rprojs/en/#LX (Information) [accessed: 2013-05-26].

[14] "Universal Decimal Classification Consortium (UDCC)," 2013, URL: http://www.udcc.org [accessed: 2013-02-10].

[15] R. C. Russel and M. K. Odell, "U.S. patent 1261167," 1918, (Soundex algorithm), patent issued 1918-04-02.

[16] D. E. Knuth, *The Art of Computer Programming: Sorting and Searching*. Addison-Wesley, 1973, vol. 3, ISBN: 978-0-201-03803-3, OCLC: 39472999.

[17] National Archives and Records Administration, "The Soundex Indexing System," 2007, 2007-05-30, URL: http://www.archives.gov/research/census/soundex.html [accessed: 2013-05-26].

[18] M. Stok, "Perl, Soundex.pm, Soundex Perl Port," 1994, (code after Donald E. Knuth).

[19] "LX SNDX, a Soundex Module Concept for Knowledge Resources," *LX-Project Consortium Technical Report*, 2013, URL: http://www.user.uni-hannover.de/cpr/x/rprojs/en/#LX (Information) [accessed: 2013-05-26].

[20] E. Rempel, "tcllib, soundex.tcl, Soundex Tcl Port," 1998, (code after Donald E. Knuth).

[21] A. Kupries, "tcllib, soundex.tcl, Soundex Tcl Port Documentation," 2003, (code after Donald E. Knuth).

[22] Max Planck Institute for the History of Science, Max-Planck Institut für Wissenschaftsgeschichte, "European Cultural Heritage Online (ECHO)," 2013, Berlin, URL: http://echo.mpiwg-berlin.mpg.de/ [accessed: 2013-05-26].

[23] E. W. von Tschirnhaus, "Brief (Letter), Ehrenfried Walther von Tschirnhaus an Leibniz 17.IV.1677," pp. 59–73, 1987, Gottfried Wilhelm Leibniz, Sämtliche Schriften und Briefe, Mathematischer, naturwissenschaftlicher und technischer Briefwechsel dritte Reihe, zweiter Band, 1667 – 1679, Leibniz-Archiv der Niedersächsischen Landesbibliothek Hannover, Akademie-Verlag Berlin, 1987, herausgegeben unter Aufsicht der Akademie der Wissenschaften in Göttingen; Akademie der Wissenschaften der DDR.

[24] G. F. von Franckenau, "Brief (Letter), Georg Franck von Franckenau an Leibniz 18. (28.) September 1697, Schloss Frederiksborg, 18. (28.) September 1697," pp. 568–569, Gottfried Wilhelm Leibniz Bibliothek (GWLB), Leibniz-Archiv der Niedersächsischen Landesbibliothek Hannover, URL: http://www.gwlb.de/Leibniz/Leibnizarchiv/Veroeffentlichungen/III7B.pdf [accessed: 2013-05-26] .

[25] Berlin-Brandenburgische Akademie der Wissenschaften, "Leibniz Reihe VIII," 2013, Glossary, Concepts, BBAW, Berlin, URL: http://leibnizviii.bbaw.de/glossary/concepts/ [accessed: 2013-05-26] (concepts glossary), URL: http://leibnizviii.bbaw.de/Leibniz_Reihe_8/Aus+Otto+von+Guericke,+Experimenta+nova/LH035,14,02_091v/index.html [accessed: 2013-05-26] (transcription), URL: http://leibnizviii.bbaw.de/pdf/Aus+Otto+von+Guericke,+Experimenta+nova/LH035.14,02_091v/LH035,14!02_091+va.png [accessed: 2013-05-26] (scan).

[26] Gottfried Wilhelm Leibniz Bibliothek (GWLB), Niedersächsische Landesbibliothek, "GWLB Handschriften," 2013, hannover, URL: http://www.leibnizcentral.de/CiXbase/gwlbhss/ [accessed: 2013-05-26].

[27] "LeibnizCentral," 2013, URL: http://www.leibnizcentral.com/ [accessed: 2013-02-10].

[28] M. Fogel, "Brieffragmente (Letter fragments) about 16xx, Historici Pragmatici universal, Terrae motus, Physica," manuscript ID: 00016293, Source: Gottfried Wilhelm Leibniz Bibliothek (GWLB), Niedersächsische Landesbibliothek, GWLB Handschriften, Hannover, URL: http://www.leibnizcentral.de/CiXbase/gwlbhss/ [accessed: 2013-05-26].

[29] M. Fogel, "Brieffragmente (Letter fragments) about 16xx, Terrae Motus in Nova Francia," manuscript ID: 00016278, Source: Gottfried Wilhelm Leibniz Bibliothek (GWLB), Niedersächsische Landesbibliothek, GWLB Handschriften, Hannover, URL: http://www.leibnizcentral.de/CiXbase/gwlbhss/ [accessed: 2013-05-26].

[30] Gottfried Wilhelm Leibniz Bibliothek Hannover, "Collection of Copperplates," 2013, URL: http://echo.mpiwg-berlin.mpg.de/content/copperplates [accessed: 2013-05-26].

[31] N. Seeländer, "Dens animalis marini Tidae prope Stederburgum e colle limoso effossi, Figura Sceleti prope Qvedlinburgum effossi," about 1716, Copperplate, (Kupferstichplatten), printed in "Leibniz, Protogaea, Tab. XII", re-printed in "Leibniz, Opera omnia, studio L. Dutens, 1768 – Wallmann, Abhandlung von den schätzbaren Alterthümern zu Quedlinburg, 1776, Tafel S. 39", URL: http://echo.mpiwg-berlin.mpg.de/ECHOdocuView?url=/mpiwg/online/permanent/echo/copperplates/Leibniz_cup4/pageimg&start=41&pn=73&mode=imagepath [accessed: 2013-05-26].

[32] C.-P. Rückemann and B. F. S. Gersbeck-Schierholz, "Object Security and Verification for Integrated Information and Computing Systems," in *Proceedings of the Fifth International Conference on Digital Society (ICDS 2011), Proceedings of the International Conference on Technical and Legal Aspects of the e-Society (CYBERLAWS 2011), February 23–28, 2011, Gosier, Guadeloupe, France / DigitalWorld 2011*. XPS, 2011, pp. 1–6, ISBN: 978-1-61208-003-1, URL: http://www.thinkmind.org/download.php?articleid=cyberlaws_2011_1_10_70008 [accessed: 2013-05-26].

[33] C.-P. Rückemann, "Dynamical Parallel Applications on Distributed and HPC Systems," *International Journal on Advances in Software*, vol. 2, no. 2&3, pp. 172–187, 2009, ISSN: 1942-2628, LCCN: 2008212462 (Library of Congress), URL: http://www.thinkmind.org/index.php?view=article&articleid=soft_v2_n23_2009_1/ [accessed: 2013-05-26] (ThinkMind(TM) Digital Library), URL: http://www.iariajournals.org/software/soft_v2_n23_2009_paged.pdf [accessed: 2013-05-26].

[34] C.-P. Rückemann, "Legal Issues Regarding Distributed and High Performance Computing in Geosciences and Exploration," in *Proceedings of the International Conference*

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

on Digital Society (ICDS 2010 / CYBERLAWS 2010), February 10–16, 2010, St. Maarten, Netherlands Antilles, D.W.I. IEEE Computer Society Press, IEEE Xplore Digital Library, 2010, pp. 339–344, ISBN: 978-0-7695-3953-9, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5432414 [accessed: 2013-05-26].

[35] C.-P. Rückemann, "Beitrag zur Realisierung portabler Komponenten für Geoinformationssysteme. Ein Konzept zur ereignisgesteuerten und dynamischen Visualisierung und Aufbereitung geowissenschaftlicher Daten," Dissertation, WWU, Münster, Deutschland, 2001, 161 (xxii+139) S., Deutsche Nationalbibliothek H 2002 A 103 (German National Library), urn:nbn:de:swb:14-1011014626187-99999 (persistent URN), oai:d-nb.de/dnb/964146754 (German National Library), OCLC: 50979238, URL: http://www.user.uni-hannover.de/cpr/x/publ/2001/dissertation/wwwmath.uni-muenster.de/cs/u/ruckema/x/dis/download/dis3acro.pdf [accessed: 2013-05-26].

[36] J. A. Lobell, "Roman Venice Discovered," *Archaeological Institute of America, November/December 2009*, vol. 62, no. 6, 1996, URL: http://www.archaeology.org/0911/trenches/roman_venice.html [accessed: 2013-05-26].

[37] A. J. Ammerman, "Probing the Depths of Venice," *Archaeological Institute of America, July/August 1996*, vol. 49, no. 4, 1996, URL: http://www.archaeology.org/9607/abstracts/venice.html [accessed: 2013-05-26].

[38] "Venice Mobility Project - Pedestrian Modeling," *Santa Fe Complex*, 2012, February, 2012, URL: http://sfcomplex.org/2012/02/venice-mobility-project-pedestrian-modeling [accessed: 2013-05-26].

[39] C.-P. Rückemann, "Advanced Scientific Computing and Multi-Disciplinary Documentation for Geosciences and Archaeology Information," in *Proceedings of The International Conference on Advanced Geographic Information Systems, Applications, and Services (GEOProcessing 2013), February 24 – March 1, 2013, Nice, Cote d'Azur, French Riviera, France*. XPS Press, 2013, pp. 81–88, Rückemann, C.-P. (ed.), ISSN: 2308-393X, ISBN: 978-1-61208-251-6, URL: http://www.thinkmind.org/download.php?articleid=geoprocessing_2013_4_10_30035 [accessed: 2013-05-26], URL: http://www.iaria.org/conferences2013/ProgramGEOProcessing13.html (Program) [accessed: 2013-05-26].

[40] K. Perrin, "Archaeological Archives: Documentation, Access and Deposition. A Way Forward," *English Heritage*, 2002.

[41] D. H. Brown, "Safeguarding Archaeological Information: Procedures for minimising risk to undeposited archaeological archives," *English Heritage*, 2011, URL: http://www.english-heritage.org.uk/publications/safeguarding-archaeological-information/ [accessed: 2012-04-08].

[42] "Guides to Good Practice," 2013, ADS, URL: http://guides.archaeologydataservice.ac.uk/ [accessed: 2013-05-26].

[43] H. Eiteljorg II, K. Fernie, J. Huggett, and D. Robinson, *CAD: A Guide to Good Practice*. Archaeology Data Service, 2002, ISSN: 1463-5194, URL: http://ads.ahds.ac.uk/project/goodguides/cad/ [accessed: 2013-05-26].

[44] "Archaeological Archives Forum (AAF)," 2013, URL: http://www.britarch.ac.uk/archives/ [accessed: 2013-05-26].

[45] *DigitalWorld 2012 / GEOProcessing International Expert Panel on Challenges in Handling Large Data Volume for GEO Processing, January 31, 2012, Valencia, Spain The International Conference on Advanced Geographic Information Systems, Applications, and Services (GEOProcessing 2012), Polytechnic University of Valencia, January 30 – February 4, 2012, Valencia, Spain*, 2012, URL: http://www.iaria.org/conferences2012/filesGEOProcessing12/GEO_2012_PANEL.pdf [2013-05-26], URL: http://www.iaria.org/conferences2012/ProgramGEOProcessing12.html (Program) [accessed: 2013-05-26].

[46] C.-P. Rückemann, *Integrated Information and Computing Systems for Advanced Cognition with Natural Sciences*. Premier Reference Source, Information Science Reference, IGI Global, 701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA, Oct. 2012, pp. 1–26, chapter I, in: Rückemann, C.-P. (ed.), Integrated Information and Computing Systems for Natural, Spatial, and Social Sciences, 543 (xxiv + 519) pages, 21 chapters, ill., ISBN-13: 978-1-4666-2190-9 (hardcover), EISBN: 978-1-4666-2191-6 (e-book), ISBN: 978-1-4666-2192-3 (print & perpetual access), DOI: 10.4018/978-1-4666-2190-9, LCCN: 2012019278 (Library of Congress), OCLC: 798809710, URL: http://www.igi-global.com/chapter/integrated-information-computing-systems-advanced/70601 [accessed: 2013-05-26], DOI: http://dx.doi.org/10.4018/978-1-4666-2190-9.ch001 [accessed: 2013-05-26].

[47] C.-P. Rückemann, "Integrating Information Systems and Scientific Computing," *International Journal on Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 113–127, 2012, ISSN: 1942-261x, LCCN: 2008212470 (Library of Congress), URL: http://www.thinkmind.org/index.php?view=article&articleid=sysmea_v5_n34_2012_3/ [accessed: 2013-05-26] (ThinkMind(TM) Digital Library), URL: http://www.iariajournals.org/systems_and_measurements/sysmea_v5_n34_2012_paged.pdf [accessed: 2013-06-09].

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Quantifying Network Heterogeneity by Using Mutual Information of the Remaining Degree Distribution

Lu Chen, Shin'ichi Arakawa, and Masayuki Murata
Graduate School of Information Science and Technology
Osaka University
1-5 Yamadaoka, Suita Osaka 565-0871, Japan
{*l-chen, arakawa, murata*}*@ist.osaka-u.ac.jp*

***Abstract*—As the Internet becomes a social infrastructure, a network design method that has adaptability against the failure of network equipment and has sustainability against changes of traffic demand is becoming important. Since we do not know in advance when the environmental changes occur and how large the changes are, it is preferable to have heterogeneity in topological structures so that the network can evolve more easily. In this paper, we investigate the heterogeneity of topological structures by using mutual information of remaining degree distribution. We discuss and show that the mutual information represents the heterogeneity of topological structure through illustrative examples. Our results show that the mutual information is high at most of router-level topologies, which indicate that the route-level topologies are highly designed by, e.g., the network operators. We also compared topologies with different mutual information, and show that, when node failures occur, the alternative paths will less converge on some of the links in topology having low mutual information.**

Keywords-power-law network; router-level topology; topological structure; mutual information; network heterogeneity; degree distribution; node failure.

I. INTRODUCTION

As the Internet becomes the social infrastructure, it is important to design the Internet that has adaptability and sustainability against environmental changes [1], [2]. However, dynamic interactions of various network-related protocols make the Internet into a complicated system. For example, it is shown that interactions between routing at the network layer and overlay routing at the application layer degrade the network performance [2]. Therefore, a new network design method which has the adaptability against the failure of network equipment and has the sustainability against changes of traffic demand is becoming important. Since complex networks display heterogeneous structures that result from different mechanisms of evolution [3], one of the key properties to focus on is the network heterogeneity where, for example, the network is structured heterogeneous rather than homogeneous by some design principles of information networks.

Recent measurement studies on the Internet topology show that the degree distribution exhibits a power-law attribute [4]. That is, the probability P_x, that a node is connected to x other nodes, follows $P_x \propto x^{-\gamma}$, where γ is a constant value called scaling exponent. Generating methods of models that obey power-law degree distribution are studied widely, and Barabáshi-Albert (BA) model is one of it [5]. In BA model, nodes are added incrementally and links are placed based on the connectivity of topologies in order to form power-law degree distribution. The resulting topology has a large number of nodes connected with a few links, while a small number of nodes connected with numerous links. Topologies generated by BA model are used to evaluate various kinds of network performance [6], [7].

However, it is not enough to explain topological characteristics of router-level topologies by such models. It is because topological characteristics are hardly determined only by degree distribution [8], [9]. Li et al. [8] enumerated several different topologies with power-law, but identical degree distribution, and showed the relation between their structural properties and performance. They pointed out that, even though topologies have a same degree distribution, the network throughput highly depends on the structure of a topology. The lessons from this work suggest us that the heterogeneity of the degree distribution is insufficient to discuss the topological characteristics and the network performance of router-level topologies.

In this paper, we focus on the property, diversity. It is a property studied in biological systems. Biological systems are systems that evolve robustly under many kinds of environmental changes. They often studied with information networks in complex system field [10]–[13]. Many of their networks also exhibit power-law attribute. A study of a key mechanism for adapting to environment changes in biological systems [10] explained that, because the system components can contribute to required traits diversely, the system can getting traits required in a new environment by changing their contribution adaptively. Prokopenko et al. [14] considered the diversity changes in growing process of some complex systems. They said that an organized system, which we consider as a less diverse system here, with effectively less configurations available. They also said that the system configurations may be have and look more complex than a disorganized system, a diverse system, to which

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

more configurations are available. From their words, we considered that a diverse system which more configurations are available to is easy to adapt to different environment. Therefore, we think that diversity is an interesting property to focus on in router-level topologies.

In [14], they used mutual information to measure the complexity, which we consider as diversity here. Inspired from their work, we investigate the topological diversity of router-level topologies by using mutual information. Here, the topological diversity means how diverse the interconnections are in any sub graphs chosen from the topology. Mutual information yields the amount of information that can obtain about one random variable X by observing another variable Y. The topological diversity can be measured by considering Y as some random variable of a part of the topology and X as the rest of it. Solé et al. [3] studied complex networks by using remaining degree distribution as the random variable. They calculated the mutual information of remaining degree distribution of biological networks and artificial networks such as software networks and electronic networks, and shown that both of them have higher mutual information than randomly connected networks. In this paper, we evaluate the mutual information of some router-level topologies, and show that the mutual information represents the topological diversity.

Heterogeneity of structures have also been studied by Milo et al. [15]. They have introduced a concept called Network Motif. The basic idea is to find several simple sub graphs in complex networks. Arakawa et al. [16] shows the characteristic of router-level topologies by counting the number of each kind of sub graph which consists of 4 nodes respectively. They conclude that router-level topologies have more sub graphs called "sector", that is removing one link from 4 nodes complete graph, than other networks. However, Network Motif is expected to evaluate the frequency of appearance of simple structure in a topology, and is not expected to measure the diversity of topology.

The rest of this paper is organized as follows. The definition of remaining degree and mutual information is explained in Section II. We investigate the topological characteristic and give some illustrative examples by changing the mutual information through a rewiring process in Section III. In Section IV, mutual information of several router-level topologies are calculated, and shown. Another topological characteristic, which is from the information network aspect, is shown in there too. Finally, we conclude this paper in Section V.

II. Definitions

Information theory was originally developed by Shannon for reliable information transmission from a source to a receiver. Mutual information measures the amount of information that can be obtained about one random variable by observing another. Solé et al. [3] used remaining degree distribution as the random variable to analysis complex networks. In this section, we explain the definitions of the mutual information of remaining degree with some example topologies shown in Table I.

Remaining degree k is defined as the number of edges leaving the vertex other than the one we arrived along, so that it is one less than the ordinary degree. The example is shown in Figure 1, where the remaining degree is set to two for the left node and three for the right node.

The distribution of remaining degree $q(k)$ is obtained from:

$$q(k) = \frac{(k+1)P_{k+1}}{\Sigma_k k P_k}, \tag{1}$$

where $P(P_1, \ldots, P_x, \ldots, P_K)$ is the ordinary degree distribution, and K is the maximum degree.

The mutual information of remaining degree distribution, $I(q)$, is

$$I(\mathrm{q}) = H(\mathrm{q}) - H_c(\mathrm{q}|\mathrm{q}'), \tag{2}$$

where q=($q(1)$, ... , $q(i)$, ... , $q(N)$) is the remaining degree distribution, and N is the number of nodes.

The first term $H(\mathrm{q})$ is entropy of remaining degree distribution:

$$H(\mathrm{q}) = -\sum_{k=1}^{N} q(k)\log(q(k)), \tag{3}$$

and the range of entropy is $0 \leq H(\mathrm{q})$. Within the context of complex networks, it provides an average measure of network's heterogeneity, since it measures the dispersion of the degree distribution of nodes attached to every link. H is 0 in homogeneous networks such as ring topologies. As a network become more heterogeneous, the entropy H gets higher. Abilene inspired topology [8], that is shown in Figure 2, is heterogeneous in its degree distribution, as shown in Figure 3. Therefore, it has higher entropy as shown in Table I.

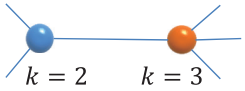


Figure 1. Example of remaining degree

Table I
Mutual information of example topologies

Topology	H	H_c	I
Ring topologies	0	0	0
Star topologies	1	0	1
Abilene-inspired topology	3.27	2.25	1.02
A random topology	3.22	3.15	0.07

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The second term $H_c(\mathrm{q}|\mathrm{q'})$ is the conditional entropy of the remaining degree distribution:

$$H_c(\mathrm{q}|\mathrm{q'}) = -\sum_{k=1}^{N}\sum_{k'=1}^{N} q(k')\pi(k|k')\log\pi(k|k'), \quad (4)$$

where $\pi(k|k')$ are conditional probability:

$$\pi(k|k') = \frac{q_c(k,k')}{q(k')}. \quad (5)$$

$\pi(k|k')$ give the probability of observing a vertex with k' edges leaving it provided that the vertex at the other end of the chosen edge has k leaving edges. Here, $q_c(k,k')$ is the joint probability, which gives the probability of existence of a link that connects a node with k edges and a node with k' edges, and it is normalized as:

$$\sum_{k=1}^{N}\sum_{k'=1}^{N} q_c(k,k') = 1. \quad (6)$$

The range of conditional entropy is $0 \leq H_c(\mathrm{q}|\mathrm{q'}) \leq H(\mathrm{q})$. Ring topologies and star topologies have the lowest H_c, because, when knowing the degree of one side of a link, the degree of the node on the other side is always determined. For Abilene inspired topology, because of its heterogeneous degree distribution, it is hard to determine the degree of the other side of a link than ring topologies or star topologies. Therefore, the conditional entropy $H_c(\mathrm{q}|\mathrm{q'})$ is higher than them. However, to compare with a random topology that have almost the same $H(\mathrm{q})$ as Abilene-inspired topology, the $H_c(\mathrm{q}|\mathrm{q'})$ of Abilene-inspired topology is lower than that of the random topology. That means the degree combination of a pair of nodes connected to a link is more biased in Abilene-inspired topology than in the random topology.

Finally, using the distribution and probability explained above, mutual information of the remaining degree distribution can also be expressed as follow:

$$I(\mathrm{q}) = -\sum_{k=1}^{N}\sum_{k'=1}^{N} q_c(k,k')\log\frac{q_c(k,k')}{q(k)q(k')}. \quad (7)$$

The range of mutual information is $0 \leq I(\mathrm{q}) \leq H(\mathrm{q})$. It is higher in star topologies and Abilene-inspired topology, since it can get more information about the degree of a node by observing the node connected to it. And $I(\mathrm{q})$ of ring topologies and the random topology is low, but the reason is different because of the difference in their H. In ring topologies, because of the homogeneous degree distribution, no information can be obtained. On the contrast, in the random topology, though the degree distribution is heterogeneous, because of the random connections, less information can be obtained. As we can see from these example topologies, $I(\mathrm{q})$ is hard to discuss without considering about $H(\mathrm{q})$. Hereafter in this paper, we mainly use $H(\mathrm{q})$ and $I(\mathrm{q})$ to discuss topologies.

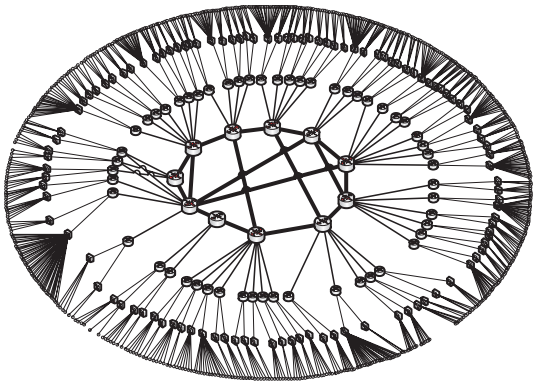


Figure 2. Abilene-inspired topology [8]

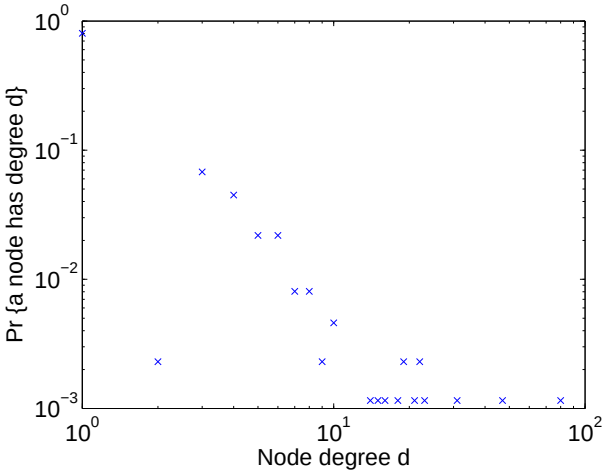


Figure 3. Degree distribution of Abilene-inspired topology

III. Mutual information and the characteristic of topologies

In this section, we explore the relationship between entropy and average hop distance. Then, we show some illustrative examples of some topologies with different mutual information.

A. Entropy H and average hop distance

To show the relationship between entropy and the characteristic of topologies, we generate topologies having different entropy, and compared their average hop distance and degree distribution.

Topologies are generated by simulated annealing that looks for a candidate network that minimize the potential function $U(G)$. Here, the temperature is set to 0.01, and the cooling rate is set to 0.0001. The simulation searched 450000 steps. The initial topology is set to a topology obtained by BA model which has 523 nodes and 1304 links, that is as same as AT&T explained in Section IV. Topologies

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

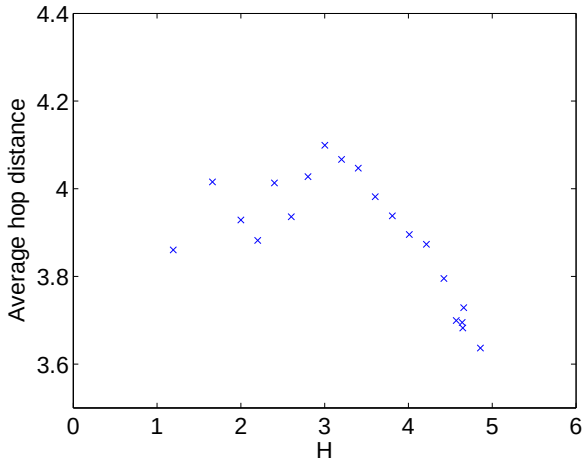


Figure 4. Average hop distance

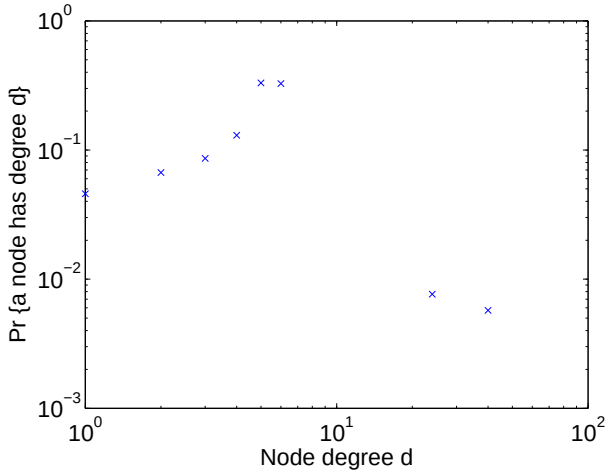


Figure 5. Degree distribution ($H = H_c = 2.2$)

are changed by random rewiring, and try to minimize the following potential function:

$$U(G) = \sqrt{(H - H(G))^2 + (H_c - H_c(G))^2}. \quad (8)$$

Here H and H_c are pre-specified value of entropy and conditional entropy respectively. $H(G)$ and $H_c(G)$ are entropy and conditional entropy calculated by the topology G generated in the optimizing search process. We generated topologies by setting H, H_c as $H = H_c$ from 1 to 5. Every time in the search process, $U(G)$ converge to approximately 0. Therefore, entropy and conditional entropy of the generated topologies are almost equal, and their I are approximately 0.

Figure 4 shows the average hop distance of topologies we generated. Degree distribution of a topology generated by setting $H = H_c = 2.2$ is shown in Figure 5, $H = H_c = 4.2$

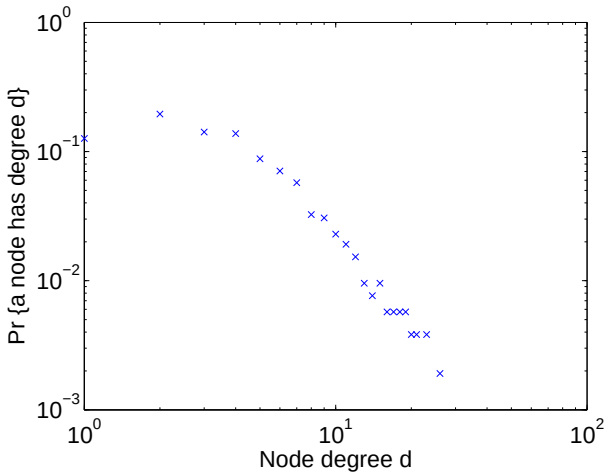


Figure 6. Degree distribution ($H = H_c = 4.2$)

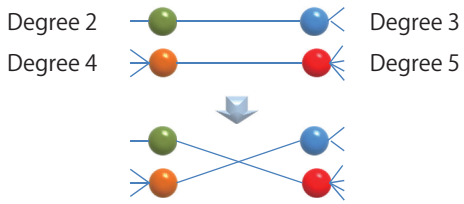


Figure 7. Rewiring method to leave the degree distribution unchanged

is shown in Figure 6. Here, average hop distance is defined as the average of hop distance between every node pairs. We calculate the hop distance by assuming the minimum hop routing. From the result, we can see that, when H increases higher than 3, the average hop distance decreases. This is because, as H increases, the degree distribution become biased, and it gets close to power-law around $H = 4$.

B. Mutual information I and topological diversity

Next, we show some illustrative examples of topologies with different mutual information. Because router-level topologies obey power-law, we compare topologies having high H.

Topologies are again generated by the simulated annealing. We set the same parameter and the same initial topology as we have used in the previous section. The different points are the way to rewire the topology and the potential function $U^I(G)$. For the first point, topology is changed by a rewiring method [17] that leaves the degree distribution unchanged, i.e., by exchanging the nodes attached to any randomly selected two links (Figure 7). For the second point, the potential function we used to minimize is $U^I(G)$ defined

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

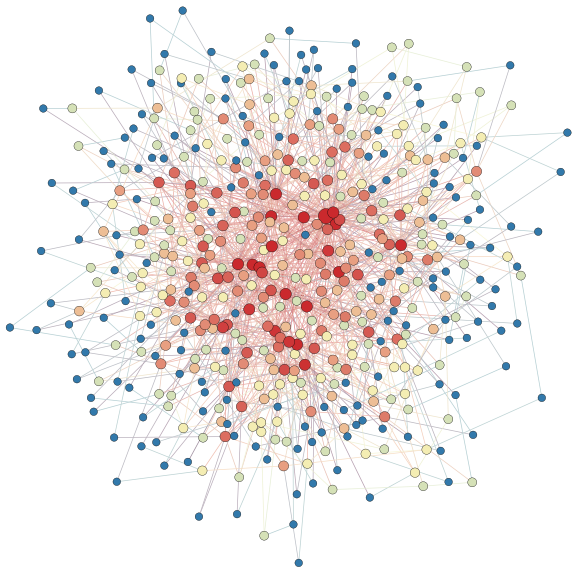


Figure 8. T_{Imin} with minimum mutual information

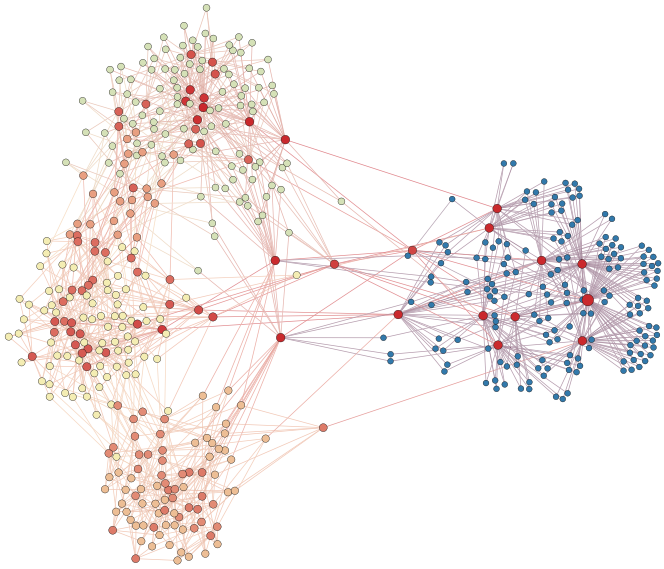


Figure 9. T_{Imax} with maximum mutual information

Table II
TOPOLOGIES OBTAINED BY SIMULATED ANNEALING

Topology	Nodes	Links	$H(G)$	$H_c(G)$	$I(G)$
BA	523	1304	4.24	3.98	0.26
T_{Imin}	523	1304	4.24	4.13	0.12
T_{Imax}	523	1304	4.24	1.54	2.70

as,

$$U^I(G) = |I - I(G)|, \tag{9}$$

where I is pre-specified mutual information, and $I(G)$ is mutual information calculated by the topology G generated in the optimizing search process. Note that looking for a pre-specified mutual information I is as the same as looking for a pre-specified conditional entropy H_c under the same entropy H. Because the entropy is same when the degree distribution unchanged, minimizing mutual entropy is identical to maximize conditional entropy.

To show the relationship between mutual information and topological diversity, we use two topologies: topology T_{Imin} with minimum mutual information and topology T_{Imax} with maximum mutual information. T_{Imin} is generated by setting $I = 0.0$ for simulated annealing, and the resulting mutual information is 0.12. The topology is shown in Figure 8. T_{Imax} is generated by setting $I = 3.0$ for simulated annealing, and the resulting mutual information is 2.70. The topology is shown in Figure 9. In both figures, colors represent node degrees. Nodes which have the same color have the same node degree. Topological characteristics of the initial topology, T_{Imin} and T_{Imax} are summarized in Table II.

From Figures 8 and 9, we can see that topology with high mutual information is less diverse, and have more regularity than the one with low mutual information. From Figure 10 to Figure 13, we show $\pi(k|k')$ dependent on remaining degree k. $\pi(k|k')$ is defined as the probability that observing a vertex with k' edges leaving it provided that the vertex at the other end of the chosen edge has k leaving edges. Figures 10 and 11 show $\pi(k|k')$ of nodes with the largest remaining degree and nodes with the smallest remaining degree in T_{Imin}, respectively. Figures 12 and 13 show $\pi(k|k')$ of nodes with the largest remaining degree and nodes with the smallest remaining degree in T_{Imax}, respectively. We can see that $\pi(k|k')$ of T_{Imax} is more biased than that of T_{Imin}. This also represents that the topology with high mutual information is less diverse than the one with low mutual information.

IV. TOPOLOGICAL DIVERSITY IN ROUTER-LEVEL TOPOLOGIES

In this section, we calculate the measurement for some router-level topologies. According to those measurements, we discuss the topological diversity of the router-level topologies. Next, we evaluate topologies with different mutual information from an information network aspect. We evaluate the amount of increment of the edge betweenness centrality under some node failure occurring situation, and evaluate the link capacity needed to deal with it.

A. Mutual information of router-level topologies

In this section, we show the mutual information of some router-level topologies. We calculated mutual information for topologies: Level3, Verio, AT&T, Sprint and Telstra. The

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

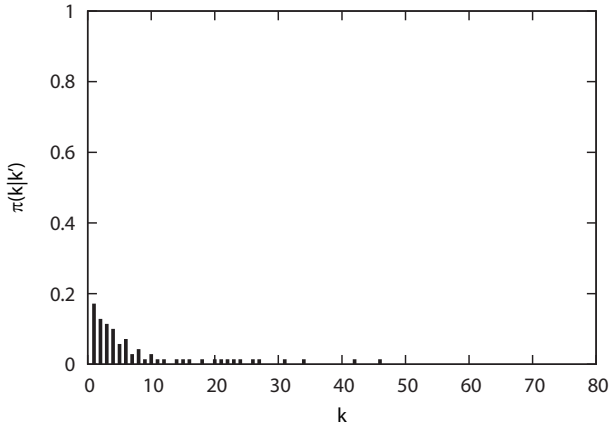


Figure 10. $\pi(k|k')$ of nodes with the largest remaining degree in T_{Imin}

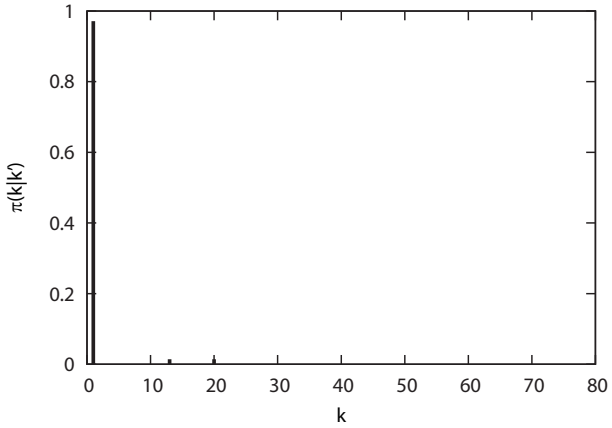


Figure 12. $\pi(k|k')$ of nodes with the largest remaining degree in T_{Imax}

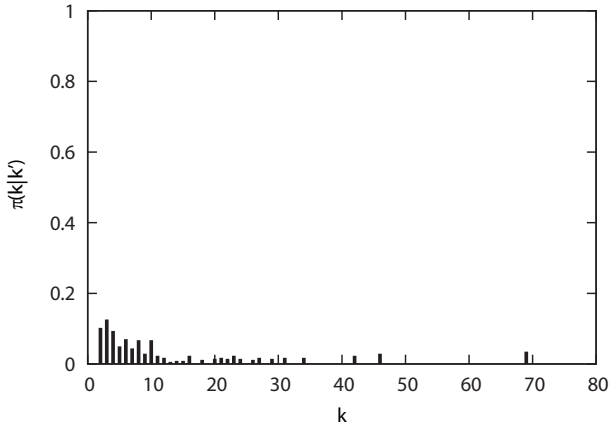


Figure 11. $\pi(k|k')$ of nodes with the smallest remaining degree in T_{Imin}

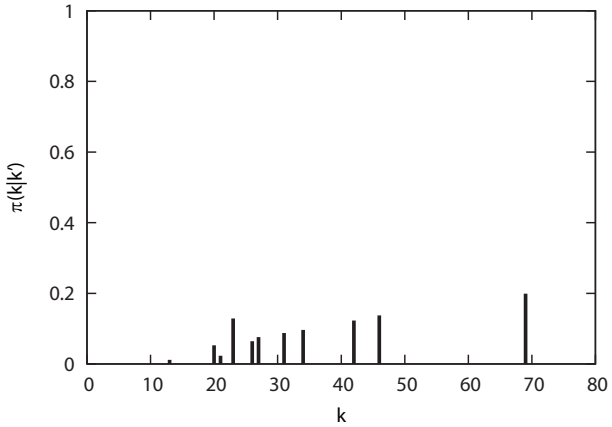


Figure 13. $\pi(k|k')$ of nodes with the smallest remaining degree in T_{Imax}

Table III
MUTUAL INFORMATION OF ROUTER-LEVEL TOPOLOGIES

Topology	Nodes	Links	$H(G)$	$H_c(G)$	$I(G)$
Level3	623	5298	6.04	5.42	0.61
Verio	839	1885	4.65	4.32	0.33
ATT	523	1304	4.46	3.58	0.88
Sprint	467	1280	4.74	3.84	0.90
Telstra	329	615	4.24	3.11	1.13
BA	523	1304	4.24	3.98	0.26

router-level topologies are measured by Rocketfuel tool [18]. To compare with those router-level topologies, a topology made by BA model [5] which has the same number of nodes and links with AT&T is also calculated. The results are summarized in Table III and Figure 14.

From Table III, we can see that, all the router-level topologies have high H, which means they have heterogeneous degree distribution. Level3 topology has higher H than others. This is because the measured topology includes many MPLS paths. These paths made the topology having high heterogeneity in degree distribution. Except Level3 topology, other router-level topologies shown in Table III has almost the same H.

Comparing those topologies with BA topology that also have almost the same H, we can see that, the mutual information of router-level topologies are higher than that of the model-based topology. This can be explained by a design principle of router-level topologies. Because router-level topologies are designed under the physical and technological constraints such as the number of switching ports and/or maximum switching capacity of routers, there are some restrictions and a kind of regulations on constructing the topologies, so that they are less diverse. Note, however, that of Verio topology is low. This can be explained by its growing history. Because Verio grows big with small ISPs [19], it contains various kinds of design principles conducted in each ISP. Therefore, Verio topology is more diverse than other router-level topologies.

B. Link capacity needed for topologies with different mutual information

In this section, we generated several topologies with different mutual information, but having the same entropy, and compared their characteristics in an information network

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

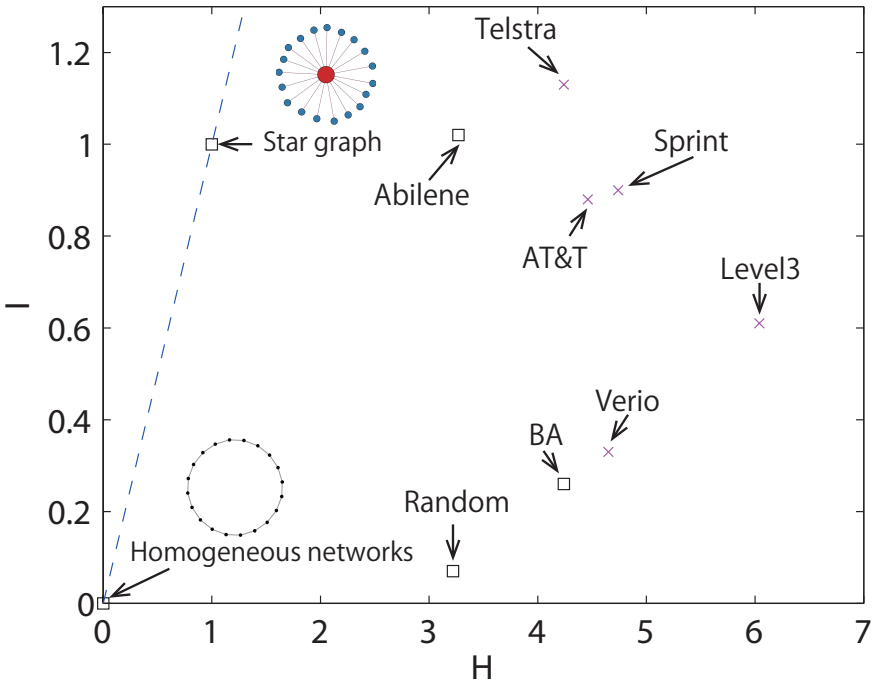


Figure 14. Entropy and mutual information

Table IV
MUTUAL INFORMATION OF TOPOLOGIES REWIRED FROM AT&T

Topology	AT&T$_{0.3}$	AT&T$_{0.4}$	AT&T$_{0.5}$	AT&T$_{0.6}$	AT&T$_{0.7}$	AT&T$_{0.8}$	AT&T
H	4.45583	4.45583	4.455834	4.45583	4.45583	4.45583	4.45583
H_c	4.17594	4.07697	3.97701	3.87589	3.77558	3.67903	3.57515
I	0.27989	0.37886	0.47882	0.57994	0.68025	0.77680	0.88068
Average hop distance	3.57439	3.56669	3.64005	3.74615	3.92027	4.18759	5.06338

aspect. To investigate the adaptability against environmental changes, we evaluate changes in edge betweenness centrality under some node failures occurring situation. When considering about the information network, it is preferable to have fewer changes in load on links even when node failures occur, because the load increment would lead to high link usage, that would increase delay, or high link capacity cost, that is needed to deal with it. To evaluate it simply, we regard edge betweenness centrality as load on links, and evaluate the minimum link capacity needed to cover node failures. Note that the edge betweenness centrality does not reflect the actual load on links. Nevertheless, we use the edge betweenness centrality to characterize ISP topologies because it gives a fundamental characteristic to identify the amount of traffic flow on topologies.

Topologies we used to compare this time are generated by rewiring AT&T randomly. The rewiring method leaves the degree distribution unchanged, which is as same as explained in Section III-B. Because the topological diversity become lower as the rewiring proceed, we calculated mutual information for every topology, and pick out topologies every time when the mutual information decreases 1 than the previous picked out one. The entropy, conditional entropy and mutual information of all the selected topologies are summarized in Table IV. AT&T$_{0.3}$ is the last topology possible to generate by this method with a long time of simulation. The average hop distance of each topology is also shown in it.

The failure we consider here is a single node failure. First, we evaluate the minimum link capacity needed to cover every pattern of single node failures. The link capacity $C(i)$ on link i is calculated as follow:

- Step 0: For all links i, set the initial edge betweenness centrality $E(i)$ as the link capacity $C(i)$:

$$C(i) = E(i). \tag{10}$$

- Step 1: When node j fails, calculate the new edge betweenness centrality $E_j(i)$ for every link. Renew the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

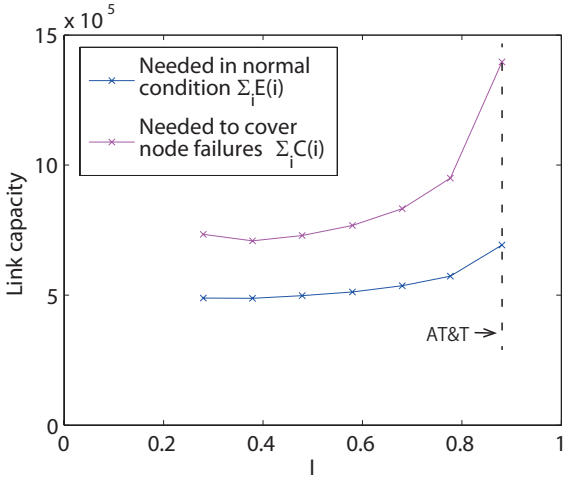


Figure 15. Link capacity

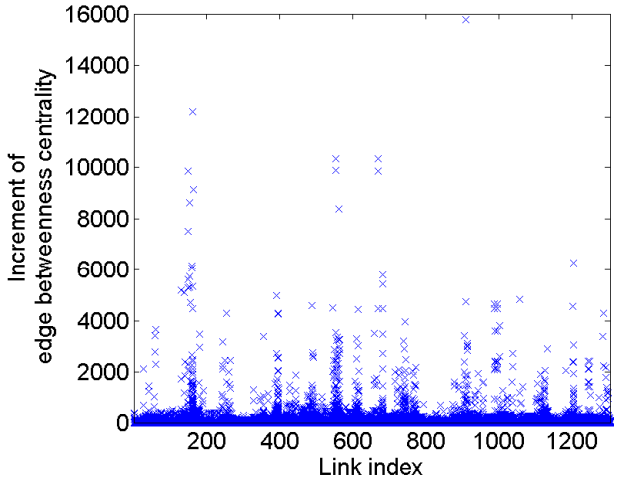


Figure 16. Increament of edge betweenness centrality (AT&T)

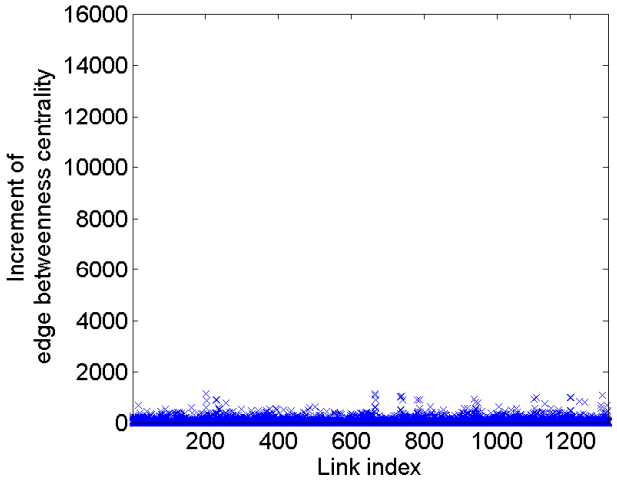


Figure 17. Increament of edge betweenness centrality (AT&T$_{0.3}$)

link capacity as (11) for every link:

$$\begin{cases} C(i) = E_j(i) & \text{if } (E_j(i) > C(i)) \\ C(i) = C(i) & \text{otherwise.} \end{cases} \quad (11)$$

- Step 2: Go back to Step1, select a new j until every node has been selected.

The total of edge betweenness centrality $\Sigma_i E(i)$ and the total of link capacity needed to cover every pattern of single node failure $\Sigma_i C(i)$ is shown in Figure 15. Because $\Sigma_i E(i)$ is directly affected by average hop distance, the difference of $\Sigma_i E(i)$ in each topology is not important. What we want to see from this figure is, the extra amount of link capacity needed to cover the node failures, which is not needed in normal condition. We can see that for the original AT&T, about twice as much as $\Sigma_i E(i)$ is needed for $\Sigma_i C(i)$. When mutual information of the topology decrease, $\Sigma_i C(i)$ tends to decrease.

We next evaluate the changes of edge betweenness centrality on each link. The increment in edge betweenness centrality is also calculated for every failure node j :

$$\begin{cases} A_j(i) = E_j(i) - E(i) & \text{if } (E_j(i) > E(i)) \\ C_i = 0 & \text{otherwise.} \end{cases} \quad (12)$$

$A_j(i)$ for all the j sorted by link index i is shown in Figures 16 and 17. Figure 16 is calculated for original AT&T, and Figure 17 is calculated for AT&T$_{0.3}$. We can see that, in AT&T, load in some of the links are highly increased compared to AT&T$_{0.3}$. This means many alternative paths tend to converge on some of the links when node failures occur. In the contrast, for AT&T$_{0.3}$, the variation of increment of edge betweenness centrality on every link is small. This can be considered because the alternative paths are balanced on

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

many links.

From these evaluations, we conclude that link capacity needed to deal with node failures decrease when the topology becomes diverse because alternative paths less convergences in such topology.

V. Conclusion and future work

In this paper, we investigated the network heterogeneity of router-level topologies by using mutual information. We mainly discussed topologies using entropy H and mutual information I.

In Section II, we used ring topologies, star topologies, Abilene-inspired topology and a random topology for examples to explain the measurements. H indicates the heterogeneity of degree distribution in complex networks, and I indicates the amount of information about the node degree that can be obtain by observing a node connected to it.

In Section III, we generated topologies between $(H, I) = (1, 0)$ and $(H, I) = (5, 0)$, and showed that, when H increases higher than 3, the average hop distance decreases. We also generated topologies that having the same H with BA model but with different I, and showed that the topology is diverse when mutual information is high, and the topology has regularity when mutual information is low.

In Section IV, from calculating mutual information of some router-level topologies, we found that most of the router-level topologies have higher mutual information than a model-based topology. From comparing the topology with different mutual information generated from AT&T, we find that link capacity needed to deal with node failures decrease when the topology becomes diverse because alternative paths less convergences in the topology with high topological diversity.

Our next work is to evaluate network performance of topologies with different mutual information also considering physical distance, and to apply this measure to designing information network that has adaptability and sustainability against environmental changes.

Acknowledgment

This research was supported in part by Grant-in-Aid for Scientific Research (A) 24240010 of the Japan Society for the Promotion of Science (JSPS) in Japan. Thanks to Hajime Nakamura, Shigehiro Ano, Nagao Ogino and Hideyuki Koto from KDDI for their helpful advice.

References

[1] L. Chen, S. Arakawa, and M. Murata, "Analysis of network heterogeneity by using entropy of the remaining degree distribution," in *Proceedings of The Second International Conference on Advanced Communications and Computation*, Oct. 2012, pp. 161–166.

[2] Y. Koizumi, T. Miyamura, S. Arakawa, E. Oki, K. Shiomoto, and M. Murata, "Stability of virtual network topology control for overlay routing services," *OSA Journal of Optical Networking*, no. 7, pp. 704–719, Jul. 2008.

[3] R. Solé and S. Valverde, "Information theory of complex networks: On evolution and architectural constraints," *Complex networks*, vol. 650, pp. 189–207, Aug. 2004.

[4] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the Internet topology," *ACM SIGCOMM Computer Communication Review*, vol. 29, no. 4, pp. 251–262, Oct. 1999.

[5] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct. 1999.

[6] R. Albert, H. Jeong, and A. Barabási, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, Jun. 2000.

[7] K. L. Goh, B. Kahng, and D. Kim, "Universal behavior of load distribution in scale–free networks," *Physical Review Letters*, vol. 87, no. 27, Dec. 2001.

[8] L. Li, D. Alderson, W. Willinger, and J. Doyle, "A first-principles approach to understanding the Internet's router-level topology," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 3–14, Oct. 2004.

[9] R. Fukumoto, S. Arakawa, and M. Murata, "On routing controls in ISP topologies: A structural perspective," in *Proceedings of Communications and Networking in China*, Oct. 2006, pp. 1–5.

[10] J. Whitacre and A. Bender, "Degeneracy: a design principle for achieving robustness and evolvability," *Journal of Theoretical Biology*, vol. 263, no. 1, pp. 143–153, Mar. 2010.

[11] N. Wakamiya and M. Murata, "Bio-inspired analysis of symbiotic networks," *Managing Traffic Performance in Converged Networks*, vol. 4516, pp. 204–213, Jun. 2007.

[12] K. Leibnitz, N. Wakamiya, and M. Murata, "Biologically inspired networking," *Cognitive Networks*, pp. 1–21, Jul. 2007.

[13] Y. Koizumi, T. Miyamura, S. Arakawa, E. Oki, K. Shiomoto, and M. Murata, "Adaptive virtual network topology control based on attractor selection," *Journal of Lightwave Technology*, vol. 28, no. 11, pp. 1720–1731, Jun. 2010.

[14] M. Prokopenko, F. Boschetti, and A. Ryan, "An information-theoretic primer on complexity, self-organization, and emergence," *Complexity*, vol. 15, no. 1, pp. 11–28, Sep. 2009.

[15] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: Simple building blocks of complex networks," *Science*, vol. 298, no. 5594, pp. 824–827, Oct. 2002.

[16] S. Arakawa, T. Takine, and M. Murata, "Analyzing and modeling router-level Internet topology and application to routing control," *Computer Communications*, vol. 35, no. 8, pp. 980–992, May 2012.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[17] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat, “Systematic topology analysis and generation using degree correlations,” *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4, pp. 135–146, Oct. 2006.

[18] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, “Measuring ISP topologies with rocketfuel,” *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, pp. 2–16, Feb. 2004.

[19] M. Pentz, “Verio grows big with small clients,” *Business Journals*, Feb. 1999.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

An FPGA Implementation of OFDM Transceiver for LTE Applications

Tiago Pereira, Manuel Violas, João Lourenço, Atílio Gameiro, and Adão Silva
Instituto de Telecomunicações
Universidade de Aveiro
Aveiro, Portugal
e-mail: targp@av.it.pt, manuelv@ua.pt, jlourenco@av.it.pt, amg@ua.pt, asilva@ua.pt

Carlos Ribeiro
Instituto de Telecomunicações
Escola Superior de Tecnologia e Gestão
Instituto Politécnico de Leiria
Leiria, Portugal
e-mail: cribeiro@av.it.pt

Abstract **– The paper presents a real-time transceiver using an Orthogonal Frequency-Division Multiplexing (OFDM) signaling scheme. The transceiver is implemented on a Field-Programmable Gate Array (FPGA) through Xilinx System Generator for DSP and includes all the blocks needed for the transmission path of OFDM. The transmitter frame can be reconfigured for different pilot and data schemes. In the receiver, time-domain synchronization is achieved through a joint maximum likelihood (ML) symbol arrival-time and carrier frequency offset (CFO) estimator through the redundant information contained in the cyclic prefix (CP). A least-squares channel estimation retrieves the channel state information and a simple zero-forcing scheme has been implemented for channel equalization. Results show that a rough implementation of the signal path can be implemented by using only Xilinx System Generator for DSP.**

Keywords – Software Defined Radio; OFDM; FPGA; time-domain synchronizatio; least square channel estimation.

I. INTRODUCTION

This paper is an extension of work originally reported in [1]. Software-Defined Radio (SDR) is both the popular research direction of the modern communication and the key technology of the 3rd generation mobile communication [2]. Ideally, in a receiver, it is an antenna connected to an Analog-to-Digital Converter (ADC) and a digital signal processing unit. However, Radio Frequency (RF) processing and down conversion is performed on the analog domain before the ADCs, see Figure 1. SDR is evolving towards the ideal and future SDRs might replace hardware with an intelligent software-controlled RF front-end (FE) [3]. Devised by Joseph Mitola in 1991 [4], it provides control over a range of modulation methods, filtering, frequency bands and bandwidths enabling its adaptability to several wireless standards in order to meet users necessities. Current home radio systems nowadays support at least 4 different radio standards (a/b/g/n) with dedicated circuits for filtering, modulating and processing each standard.

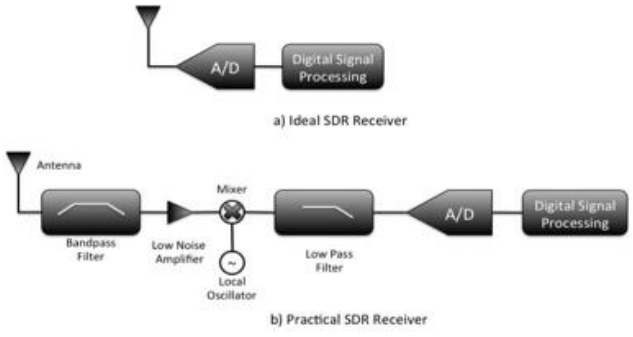


Figure 1. The Software-Defined Radio architecture

A SDR's *reconfigurability* allows the programming of the required standard instead of building extra hardware according to a standard's need. If multiple waveforms can be designed to run on a single platform, and that platform can be reconfigured at different times to host different waveforms depending on the operational needs of the user, it stands to reason that fewer platforms may be needed [6]. SDR is forcing a fundamental change in the business model by both platform and waveform developers, in that – although capability is still a key discriminator – the low cost solution wins [6].

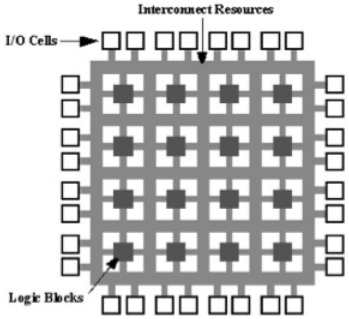


Figure 2. FPGA structure [5]

Field Programmable Gate Arrays (FPGAs) are mainly used in SDR RF FEs to improve the performance of digital signal processing chip-based systems [7]. Current FPGA vendors include Xilinx, Altera, Actel, Lattice, Tabula, among others. Each vendor has its architectural approach. A FPGA, see Figure 2, is a reconfigurable logical device

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

consisting of an array of small logic blocks and distributed interconnection resources and is characterized by a structure that allows a very high logic capacity. They provide a higher computing power when compared to Digital Signal Processors (DSPs) or General Purpose Processors (GPPs) due to their parallel processing nature, which are essentially serial in operation.

One of the peculiarities of FPGA is "number representation." Unlike GPPs who are typically equipped with Floating Point Units (FPUs), most DSPs and FPGAs are outfitted with highly parallel multiplier-accumulator cores dedicated to fixed-point precision operations, and even though the support for FPGAs floating-point operations has increased, there are no RF FEs that perform floating-point precisions. In signal processing, the additional range provided by floating point is uncalled for in most cases and fixed-point operations on DSPs and FPGAs provide you a large speed and cost benefit due to their dedicated cores. Still regarding operation speed, if you are running a program on a GPP that has multiple fixed-point multiply/accumulate cores then it will be far faster in fixed-point. On the other hand, on a standard x86 chip, it will actually probably be slower in fixed point. A floating-point representation will have a higher accuracy though and an example is given in [8]. Even though the embedding of FPUs in FPGAs is discouraged; encouragement to improve floating-point support is discussed in [9].

The development of wireless networks is a lasting process that includes many stages, but at some point, verification on a hardware testbed is needed to validate the theoretical and simulation work. Such testbeds are used not only for theory verification, but there are also some concepts that can only be seriously studied in practice (e.g., interference modeling). For instance, rarely a communication theory student needs to spend time understanding the impact of I/Q imbalance, while a student working on a testbed will have to consider such effects.

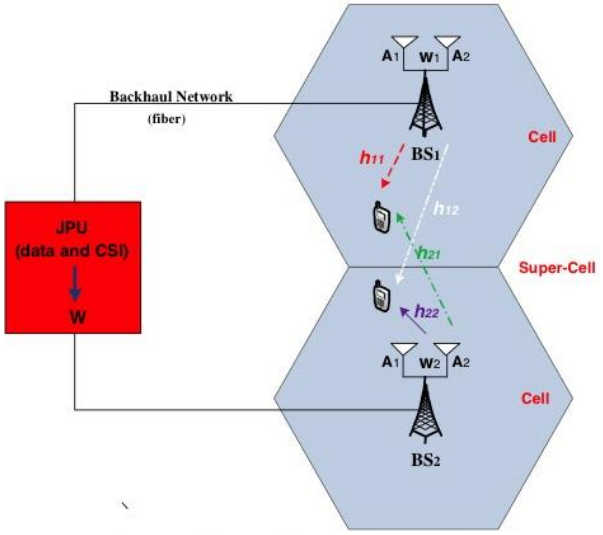


Figure 3. Multicell cooperative scenario

While theory and simulations typically show the corresponding gains under ideal conditions, hardware platforms and testbeds are essential in validating these gains in real channels and in the presence of implementation impairments [10].

In a distributed antennas system, see Figure 3, the radio signals are jointly processed at a central point, therefore enabling efficient interference mitigation, space diversity and uniform coverage inside the cell. Recently, some practical centralized precoding schemes that can be employed in the considered platform have been proposed [11]-[14]. Two centralized multicell precoding schemes based on the waterfilling technique have been proposed in [11]. It was shown that these techniques achieve a close to optimal weighted sum rate performance. A block diagonalization (BD) cooperative multicell scheme was proposed in [11], where the weighted sum-rate achievable for all the user terminals (UTs) is maximized. A promising centralized precoding scheme based on Zero-Forcing (ZF) criterion with several power allocation approaches, which minimize the average BER and sum of inverse of signal-to-noise ratio (SNR) was proposed in [13][14].

The aim of this article is to present the implementation of an FPGA-based Orthogonal Frequency-Division Multiplexing (OFDM) receiver with a ML time-domain synchronization and a frequency-domain Least-Squares (LS) Channel Estimator (ChEst) using Xilinx System Generator for DSP (SysGen) and Xilinx ISE Design Suite. SysGen is a high-level design "toolbox blockset" built into Matlab's Simulink providing the user with high-level abstractions of the system that can be automatically compiled into an FPGA. It provides the user a thin boundary between hardware and software, given that it enables hardware design by allowing the blocks to be synthesized into VHDL and compiling them into a FPGA with a single click. The FE for the platform we are using does require VHDL knowledge, although not all boards in the market do at this point. This allows the user to abstract himself from a time-consuming and knowledge-dependent programming language such as VHDL or Verilog, as well as thousands of lines of code. Even though some SysGen blocks need to be studied for timing and feature purposes, they are in many ways similar to Simulink blocks making them easier to work on.

We discuss some testbeds present on literature nowadays. We present some uncertainties present on the radio domain as well as a possible algorithm to correct them in higher detail along with its implementation. We show our testbed current architecture as well as our go-to deployment scenario. We "focus" on time-domain synchronization using the Beek algorithm and frequency domain LS channel estimation. We show some Bit Error Rate (BER) results with a ZF equalization as well as the simulation method (hardware co-simulation). To finish, we yield some conclusions.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

II. BACKGROUND AND RELATED WORK

Although multicarrier techniques can be traced back to 1966 [15], the first commercial application of OFDM occurred only in 1995 with the Digital Audio Broadcasting (DAB) standard [16]. OFDM is a multicarrier bandwidth efficiency scheme for digital communications, where the main difference to conventional Frequency Division Multiplexing (FDM) is that in the frequency domain the OFDM subcarriers overlap, providing spectrum efficiency. Given that OFDM implementations are carried out in the digital domain, there are a number of platforms able to implement an OFDM system suitable for SDR development.

SDR testbeds can be discerned between 2 main fields: hardware platforms and software architectures. The hardware features of an SDR consist of the RF parts and communication links to the software-based signal-processing component. The remaining parts can be composed of a DSP, a FPGA or a GPP.

The BEEcube Company is probably the best growing example on this field and has the Berkeley Emulation Engine 4 (BEE4) as its latest platform. It consists of a platform with 4 different modules, each one supporting a variety of 4 Xilinx Virtex-6, allowing the support of 20 million gate designs per module. Users can run logic up to 500 MHz and digital communication at 640 Gbps per module, along with flexible expansion options such has HDMI. It explores an FPGA capability of processing a large data amount in parallel very quickly. Similar to our system, it also implements its design flow in SysGen. BEE system tests include projects such as an emulation of a Time-Division Multiple Access (TDMA) receiver with an 806 kHz symbol rate using 3 processing FPGAs, 1 crossbar FPGA, and achieves a maximum operating frequency of 25 MHz [17]; a single-channel 2.4 GHz radio system capable of operating in real-time with a 32 MHz system clock rate; a video encoder; a complex iterative decoder design, and other DSP related component designs. Additional BEEcube models include the miniBEE "R&D in a box" platform aimed at smaller designs containing a single Virtex-6 FPGA and targets applications such as Wireless Digital Communications, High Performance Computing, and Video Prototyping, among others. The BEE7 will be introduced in 2013, and will be packaging the latest Xilinx Virtex-7 FPGA family.

Another well-known hardware platform is the Wireless open-Access Research Platform (WARP) from Rice University. One of its fundamental attributes is the central repository [18] dedicated to free distribution of hardware and software projects on the WARP website. It is an extensible reprogrammable platform built for prototyping wireless networks [19]. Their latest model, the WARP v3.0 has a Xilinx Virtex-6 FPGA, two 12-bit ADCs with a sampling rate of 100 MSPS, two 10-bit DACs with a sampling rate of 170 MSPS and comes by default with a 200 MHz Low-Voltage Differential Signaling (LVDS) oscillator. Its capability enables the programmability of both physical and network layer protocols. For design flow implementation on the WARP hardware platforms, Rice developed two dedicated software architectures, WARPnet and WARPLab. WARPLab is a non-real-time system that brings together WARP and Matlab through an Ethernet switch. One can interact with WARP nodes directly from the Matlab workspace and signals generated in Matlab can be transmitted in real-time over-the-air using the nodes, facilitating rapid prototyping of physical layer (PHY) algorithms directly in Matlab M-Code [20]. Transmitter and receiver processing is performed offline in Matlab. WARPnet is a SDR measurement framework for real-time designs built around client-server architecture in Python [21][22] and it uses a packet capture (PCAP) application-programming interface (API) to communicate with the WARP nodes directly. The PHY layer is implemented on SysGen and VHDL while the Medium Access Control (MAC) layer is implemented in C/C++ code using Xilinx Platform Studio (XPS). Hardware Co-Simulation, see Section 5, is also supported [21][23]. A real-time cooperative OFDM transceiver is presented on [22][23] [24][25]to explore the utility of PHY layer cooperation in real-world wireless systems and early performance results are performed using WARP. An architecture for MAC protocol development and performance evaluation entitled WARPMAC is presented in [22]. A similar work in [26] uses this testbed to present an OFDM-based cooperative system using Alamouti's block code to study its capability versus a 2 x 1 multiple input single output (MISO) system. It is a suite of software routines that sits above the PHY layer and allows for flexible abstraction of hardware interactions [24][27]. On [25][28] a flexible architecture of a high data rate LTE uplink receiver with multiple antennas is implemented in a single FPGA using SysGen and then verified with WARPLab on a real over-the-air indoor channel supporting data rates up to 220 Mbps.

As for software architectures, the open-source GNU Radio [29] is a development toolkit distributed under the GNU General Public License that provides a set of signal processing libraries for the implementation of the processing blocks required by a transmission system. The GNU Radio project has started in 2001 and now has a large community worldwide devoted to the use of the platform for different applications: OFDM systems, GSM communications, GPS receivers, HDTV receivers, RF sensing, amateur radio applications, FM radio, etc.

The GNU Radio platform runs on Linux-based machines and processing blocks other than the ones given in the libraries are written in C++ language. The flow graph of the system is defined in Python language that defines the interaction among the different blocks.

This platform only implements the digital baseband processing and RF hardware is not part of GNU Radio. To implement the RF transmit and receive paths, off-the-shelf low-cost external hardware is readily available. Some of the boards that interface with the platform are Ettus Research

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

USRP Series [30], FlexRadio Systems hardware [31], open source HPSDR hardware [32], AMRAD Charleston SDR project board [33], etc. The equipment that stands-out as the most commonly used is the USRP family of devices. A USRP device is made-up of a baseband analog/digital processing motherboard and an RF FE daughterboard. The RF boards cover frequencies from DC to 6GHz with different bandwidths, gains and noise figures. The motherboards are able to process signals with bandwidths up to 50MHz with 100MSamples/s ADCs and 400MSamples/s DACs.

Smaller scale testbeds for OFDM systems based on GNU Radio have been reported in the literature. An OFDM modulator/demodulator with two synchronization options and two error-controlling techniques is reported in [27][34]. The work in [28][35] uses GNU radio to transfer OFDM signals with Quaternary Phase Shift Keying (PSK) and Binary PSK modulation to analyze the packet-received ratio for Quality of Service purposes. An implementation of superposition coding for OFDM systems using the GNU Radio is presented in [34][36]. FPGA implementations of standards 802.11a and 802.16-2004' modulators using Xilinx System Generator for DSP for high-level design can be found in [37][38].

III. THE ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING TRANSCEIVER

A. Testbed Architecture

Figure 4 depicts the transceiver architecture of the system discussed in this paper. On the transmitter, data is generated randomly by making an inverse fast Fourier transform (IFFT) of quadrature amplitude modulated (QAM) symbol sets with 1024 subcarriers. The CP is added after the IFFT and the symbols are turned into frames. An up-conversion of 4 is performed on the digital up conversion (DUC) block by a set of two interpolation filters: a square-root-raised-cosine and a halfband.

The mixer and direct digital synthesizer (DDS) block performs frequency translation to an intermediate frequency (IF) and is achieved by mixing the frame with a DDS. On the receiver side, another DDS translates the IF back to baseband on the mixer block. Down-conversion and matched filtering is performed by a similar set of filters as the ones used on the transmitter by the digital down conversion (DDC) block.

Once the estimations for the offsets are performed, the frame to symbol and CFO correction blocks performs the compensations. A fast Fourier transform (FFT) shifts the data back into the frequency domain. A LS channel estimator is implemented to retrieve the channel state information (CSI) and a ZF equalizer applies the estimations. Once pilots and DC subcarriers are removed, the data is demodulated back into bits. Several parameters along the system are reconfigurable at users need. Such parameters include number of symbols per frame, CP length, carrier frequency (limited by the system's frequency), modulation (QPSK, 16-QAM, 64-QAM, etc.) and the system's main clock frequency, among some others.

Two critical parts of the receiver are the time-domain synchronization and channel estimation subsystems. On the time-domain synchronization, we should estimate the frame arrival time and the frequency offset between the local oscillators and RF carriers. Compensation can then be applied to the received signal. On the channel estimation subsystem on the frequency domain, the CSI will be estimated by a channel estimator and then corrected by an equalizer. In the following subsections, we will detail these two algorithms.

B. Time-domain synchronization - Beek

Receiver and transmitter operate with independent local reference oscillators. In order to perform an efficient demodulation, the receiver should be able to perform frame and carrier synchronization. The first operation defines the starting / ending points of the frame while the latter synchronizes the phase / frequency between transmitter and receiver. Erroneous frame detection is projected into the symbol constellation with a circular rotation, whereas the carrier frequency offset (CFO) causes all the subcarriers to shift and is projected as dispersion in the constellation points. Both ambiguities yield the received signal:

$$r(k) = \mathrm{s}(k-\tau)\, e^{\,j2\pi\varepsilon k/N} + n(k) \qquad (1)$$

where ε is the normalized CFO, τ is the unknown arrival time of a frame, $\mathrm{s}(k)$ is the transmitted signal, N is the

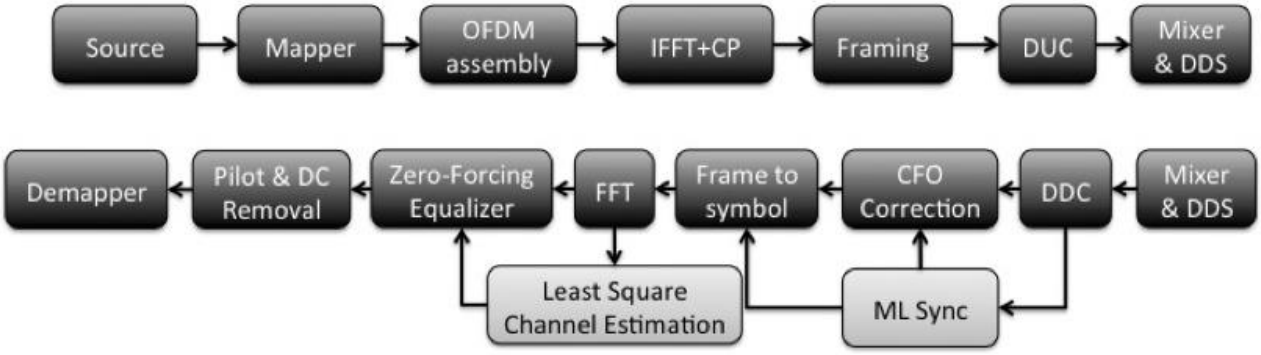


Figure 4. Transceiver architecture

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

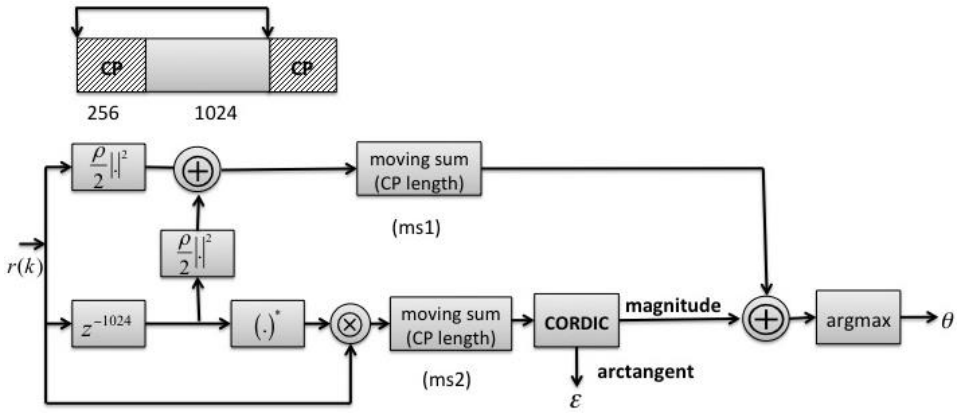


Figure 5. Beek estimation algorithm architecture

number of samples per symbol, $n(k)$ is the additive white gaussian noise (AWGN) and k is the sample index of each symbol ranging [0,1023].

Moose [39] presented a simple method using the CP just like Beek [40]. Schmidl and Cox [41] use the repetition on the preamble, providing a more robust algorithm for symbol formats where the CP is short.

We do not make use of preamble repetition on our system, although we use Zadoff-Chu (ZC) sequences at the beginning of each frame for time-domain synchronization due to its good autocorrelation properties and given that they are a part of 3GPP Long Term Evolution (LTE) air interface. Beek's algorithm, see Figure 5, was the chosen one due its mediocre complexity and it can be easily adapted to take advantage of our ZC sequences.

C. *Frequency-domain estimation – Least-Squares Channel Estimation*

Channel estimation has always been present in wireless communications systems to assist the receiver in mitigating the effects of the wireless multipath channel on the received signal. In OFDM systems, the acquisition of accurate (CSI) is crucial to achieve high spectral efficiency, with emphasis on the demodulation/decoding process, where the frequency response of the channel at the individual subcarrier frequencies needs to be accurately estimated to be used in the decoding process. Furthermore, the synchronization algorithm presents a phase offset ambiguity after frequency offset correction that must be estimated by the channel estimator and removed in the equalization process.

The system discussed in this paper uses the common rectangular pilot pattern adopted by the LTE standard with some adaptations, where a 12 symbol OFDM frame carries pilots in the 1st, 5th and 9th symbol. The pilot-carrying subcarriers are optimally equipowered and equidistant to

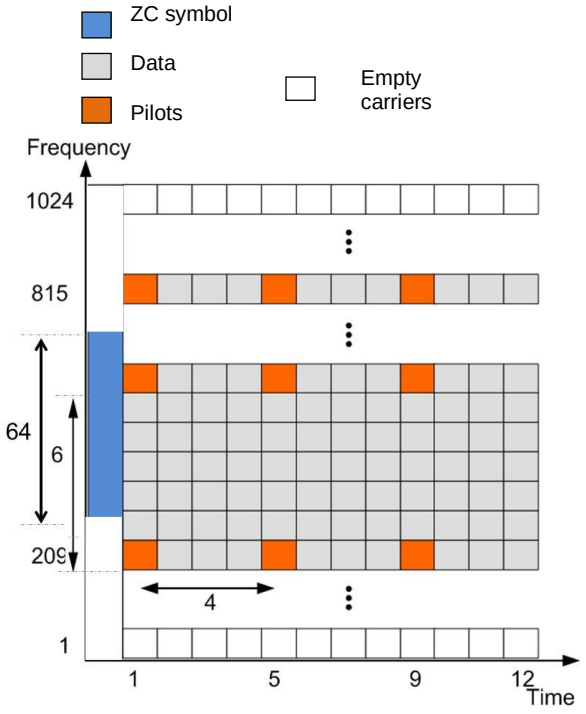


Figure 6. Frame structure

achieve the lowest mean square error (MSE) [42][43], considering that the transceiver uses LS channel estimation.

The distance between consecutive pilots is 6 subcarriers. The first and last 208 subcarriers are not loaded making-up the band guards on each end of the spectrum to contain the spectral leakage typical of OFDM systems. An initial ZC training symbol is appended to the frame for synchronization. The frame structure is depicted in Figure 6.

This pilot arrangement has been extensively used in the related literature. Some of the outstanding works on channel estimation that used it can be found in [44][45][46].

To overcome the issue of having to extrapolate the edge subcarriers [47][48], with the subsequent degradation of the

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

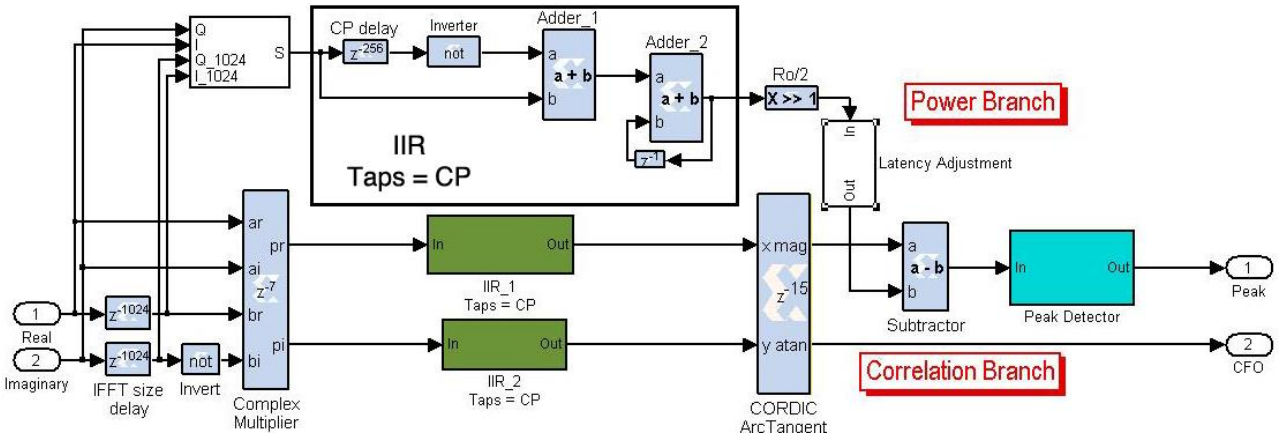


Figure 7. Beek estimation algorithm implementation on Xilinx System Generator for DSP

estimation accuracy, the adopted frame structure has pilots at both edge subcarriers.

In this work, the initial estimate in the pilot subcarriers used the well-known LS estimator [49]. This classical estimator does not take advantage of the correlation of the channel across the subcarriers in frequency and time domains nor does it use a-priori information on the channel statistics to obtain the estimate, but, on the other hand, presents a reduced implementation complexity, requiring only an inversion and a multiplication per pilot subcarrier. Considering that the value received in the *k*th pilot subcarrier $p(k)$ can be expressed by

$$u(k) = s(k)h(k) + n(k) \tag{2}$$

where $h(k)$ is the channel value affecting the *k*th pilot subcarrier. The LS estimation's output can be expressed as

$$\hat{h}(k) = \frac{p(k)}{s(k)} = h(k) + \frac{n(k)}{s(k)} \tag{3}$$

that can be interpreted as noisy samples of the wanted channel frequency response (CFR).

In the literature, some channel estimation schemes output the full channel estimate (for both data and pilot subcarriers) [44], but our initial estimation only outputs the channel values for the pilot subcarriers. It is now necessary to estimate the channel values for the data-carrying subcarriers. The simplest method would be to extend the current channel estimates to the closest pilots in both frequency and time domains [50]. This method only yields acceptable performance if the correlation of the CFR for neighboring pilots is significant. Therefore, it is only adequate for scenarios where the channel varies slowly and has a limited delay spread. The transceiver introduced in this paper adopted a linear interpolation method in the frequency domain, similar to the one found in [51][52], using a first order polynomial to define the line that connects two neighboring pilots, enhancing the performance of the previous scheme [53]. Higher order polynomials could be used [54]-[56] to achieve higher accuracy in estimating highly selective channels, at the cost of a higher implementation complexity. With the full CFR for the pilot-carrying symbols, and as the pilot separation is small in time domain (4 symbols), the transceiver extends each CFR estimate until next pilot-carrying symbol, to get the full frame CFR.

IV. BEEK ESTIMATION, FRAME SYNCHRONIZATION AND CFO COMPENSATION

The following subsections present the time-domain synchronization algorithm divided in three parts.

A. Estimation of frame arrival time and carrier frequency offset

The algorithm presented on this subsection is based on the algorithms developed by Beek and the subsystem created for its purpose and adapted to the frame pattern on Figure 6 is illustrated in Figure 7. Beek exploits the CP by correlating it with a delayed version of itself. When the repeated pattern is located, a peak is generated in order to detect the frame arrival and the phase between patterns gives the CFO.

The algorithm consists of two main branches. The top one calculates an energy term. While the bottom one calculates the correlation term required for estimating both symbol arrival time and phase offset. Equation (4) shows the calculation of the energy term and equation (5) shows the calculation of the correlation term.

$$ms1 \equiv \frac{\rho}{2} \sum_{k=m}^{m+L+1} |r(k)|^2 + |r(k+N)|^2 \tag{4}$$

$$ms2 \equiv \frac{\rho}{2} \sum_{k=m}^{m+L+1} r(k) r^*(k+N) \tag{5}$$

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

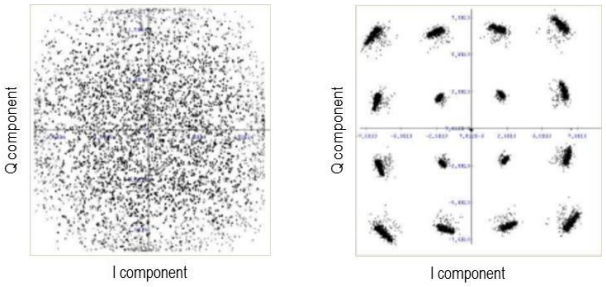


Figure 8. OFDM symbol constellation with a 6 kHz offset between oscillators. Before compensation (left) and after compensation (right)

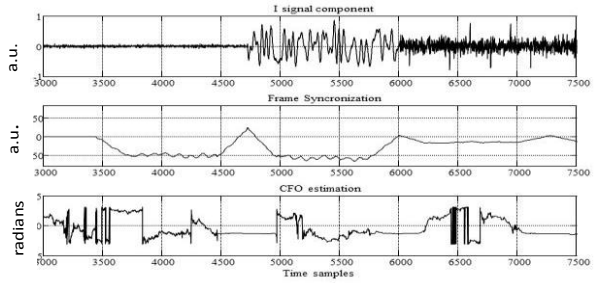


Figure 9. Estimation algorithm results for the 1st three symbols of a frame (Zadoff-Chu and two symbols) without AWGN: (a) signal, (b) peak estimation and (c) computed angle

The factor ρ is the magnitude of the correlation coefficient between $r(k)$ and $r(k+N)$; it depends on the signal-to-noise ratio but can be set to 1. Both moving sums were designed using infinite impulse response (IIR) filters.

The complex multiplier core present on the SysGen libraries performs multiplications throughout the subsystem. In order to proceed with both estimations, two operations must be performed on the bottom branch, a complex module to create the peak when the CP correlates with its delayed version and an arctangent to calculate the angle between both IQ signals to enable CFO estimation, see Figure 9. SysGen provides a CORDIC arctangent reference block that implements a rectangular-to-polar coordinate conversion using a CORDIC algorithm in circular vectoring mode, that given a complex-input <I,Q>, it computes a magnitude and an angle according to (6) and (7), respectively.

$$|\mathrm{I,Q}| = \sqrt{\mathrm{I}^2 + \mathrm{Q}^2} \tag{6}$$

$$\mathrm{ang} = 2\pi\varepsilon = \arctan(\mathrm{Q/I}) \tag{7}$$

It is assumed that the offset between oscillators is lower than a single subcarrier and so $|\varepsilon| < 1/2$. On [57], a division is performed to create the necessary peak for frame arrival detection, but such operation in hardware is more expensive and should be avoided. The only difference brought by the difference operation is how the peak is generated, since the argument to be detected will be close to 0 with a subtraction and to 1 with a division. Achieving a theoretical value of 0 when a signal is detected is not a realistic approach since the fixed-point logic used is subject to quantization errors and to contention of bit propagation along the system. The computed angle is only used when the peak is detected, ensuring the CFO is only used if the correlation is complete.

B. Data forwarding control

This subsystem uses the peak detected for each ZC to process the frame in order for each symbol to be processed by the FFT. Unlike a non-deterministic simulation such as the ones ran in Simulink, a FPGA simulation does not have the ability to hold the information on its own while the estimations described on the previous subsection are executed. Data must be contained in a memory and forwarded when a condition is met or delayed by a constant value if the process is continuous, which is the case. The processing time required for a peak to be detected and the accurate CFO to be estimated is known, constant and introduced as a delay before the FIFOs. The peak detected on subsection *A* triggers the frame writing into the FIFOs. The CP is not needed anymore so it is not stored. The FFT will require 3*N samples to process each symbol and output it.

This amount of samples needs to be created given that the symbols stored on the FIFOs are continuous. Reading the data stored on the FIFOs at a sampling rate four times higher as the symbols arrive creates that gap, breaking the frame back into separate symbols.

C. Carrier frequency offset correction

Correction of the CFO is achieved with a CORDIC implementing a rotate function [58]. The core rotates the vector (*I*,*Q*) by an angle ϕ yielding a new vector (*I'*,*Q'*) such that

$$\begin{aligned} I'(k) &= I(k)\times\cos\phi - Q(k)\times\sin\phi \\ Q'(k) &= Q(k)\times\cos\phi + I(k)\times\sin\phi \end{aligned} \tag{8}$$

where

$$\phi = 2\pi\varepsilon k / N \tag{9}$$

Taking the angle achieved at subsection *A*, the angle is first divided by *N* and then accumulated along each symbol nullifying the phase offset along each symbol, see Figure 8.

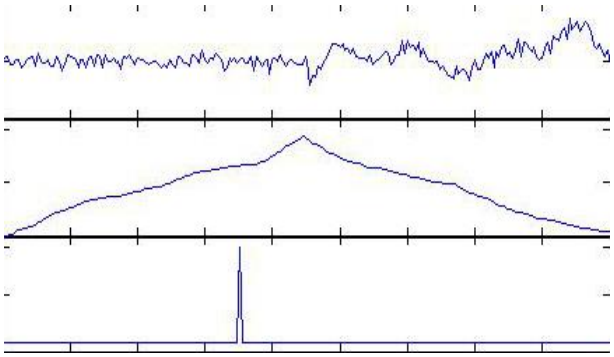


Figure 10. Erroneous peak detection

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Figure 11. FPGA hardware platform setup

D. Estimation Issues

On Figure 8, the received constellation is rotated due to two possible factors, an erroneous frame detection arrival time, which will be discussed shortly, and/or an offset between the oscillators starting time. Both errors are compensated on the channel equalization subsystem on the frequency-domain.

An issue brought by this algorithm is how noise affects the correlation algorithms as seen on Figure 10. Assuming a peak detection algorithm where the peak, that sets the frame start time, is defined on sample N when N+1>N after a given threshold, flawed detections may occur when noise is present. If the peak is detected before the actual peak occurs, a rotation is induced on the constellation and compensated by the channel equalizer. On the other hand, if the peak is detected after the actual peak occurs, random distortion is introduced due to intersymbol interference (ISI) and intercarrier interference (ICI). A peak detection algorithm based on maximum value would always perform a detection closer to the peak, but it would be more time-consuming and it would not be error-free either if the noise disturbed the correlation near the peak. The current algorithm will not avoid this problem either, so we shift the detected peak by three samples into the cyclic prefix to ensure that the frame start is not set inside the symbols useful time.

TABLE I. SYSTEM PARAMETERS

System Parameters	
Baseband frequency || Bandwidth	15.36 MHz || 10 MHz
FFT size || CP size	1024 || 256
Modulation	QPSK - 16QAM
Subcarrier separation	15 kHz
Symbol duration (Symbol+CP)	66.66 + 16.66 = 83.32 μs
IF sampling frequency	61.44 MHz
Oscillator frequency	15 MHz

V. TESTBED,SIMULATIONS AND RESULTS

Even though we are targeting the 3GPP LTE standard at this point, such implementation can be easily adapted to several other OFDM standards such as 802.11a, WiMAX, among others, given the reconfigurability of the parameters.

The design was compiled through hardware co-simulation. It is a methodology introduced by Xilinx on 2003 [59], which allows a system simulation to be run completely in hardware (FPGA), while showing the results in Simulink. This enables accurate hardware modeling along with faster simulation times due to the faster calculations and easier hardware verification by implementing the manufactured algorithm into the FPGA. Xilinx's block components behaviors are projected to Simulink, while at the same time; the behavior of each block's associated hardware component is performed on the FPGA. The objective is to get both hardware and software working before the prototyping stage by providing a better understanding of its behavior.

The targeted model for the simulation was the Xilinx ML605 development board, which contains a Virtex-6 LX240T FPGA, and a 4DSP FMC150 FMC daughter card with a dual 14-bit 250 MSPS ADC and a dual 16-bit 800 MSPS DAC, see Figure 11.

The tests were performed in a wired-channel and the system was run at a system clock of 61.44 MHz with an IF of 15 MHz. The BER hardware results were obtained using Simulink on a hardware co-simulation mode without the daughter card, with some parameters being shown on Table I. The theoretical results were obtained from an adapted Matlab OFDM chain that is used in [60]. Figure 8 was

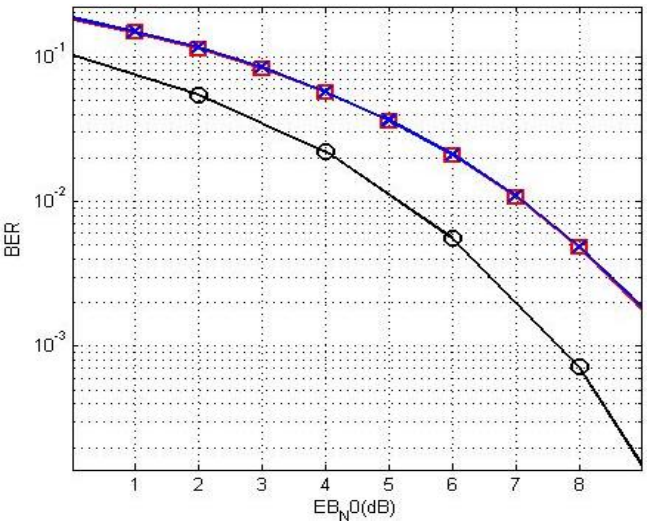


Figure 12. Baseband BER results for 3 simulations: Perfect CSI (black) and Zero-Forcing Equalization with/without time-domain synchronization (blue and red)

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

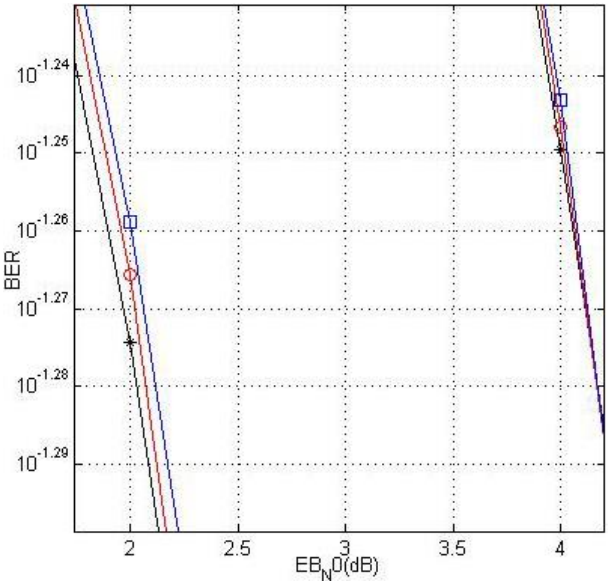


Figure 13. BER results for 2 simulations: Perfect CSI (left lines) and Zero-Forcing Equalization without time-domain synchronization (right lines). Theoretical (black*), baseband (red O) and IF (blue □)

obtained using Xilinx ChipScope Pro Tool with the daughter card attached. Because of the daughter card present on the testbed, a wrapper must be created with Xilinx ISE Design Suite in order to connect both the system presented here and the daughter card where the DACs/ADCs are present. Table II shows the resources used for the full transceiver, without the wrapper.

Figures 12 and 13 illustrate the BER theoretical and practical results for three different simulations: perfect CSI, no time-domain synchronization with a ZF equalizer and time-domain synchronization with a ZF equalizer.

VI. CONCLUSION AND FUTURE WORK

A full baseband + IF design was presented focused on the synchronization and channel estimation algorithms. The work presented was performed using Xilinx System Generator for DSP, the ChipScope Pro tool, ISE Design Suite, and validated with Matlab Simulink.

SysGen does not allow the user to replace hardware description language (HDL) completely but allows him to focus the attention on the critical parts of and optimizing paths, HDL is better suited. The amount of resources used for the design was never a priority and can certainly be reduced by optimizing the register transfer level (RTL) of the design to ensure maximum reuse and an efficient implementation [7].

In Figure 9, a rough estimation of the frame is presented, with a peak being generated at the beginning of each symbol and the respective CFO on the bottom, thus proving an accurate arrival time and CFO estimation of each symbol. It is also possible to perform a symbol-based estimation instead of a frame-based one, with no additional complexity brought by the change.

TABLE II. RESOURCE USAGE OF THE FULL SYSTEM

Full system resource usage for Virtex-6		
Parameter	***Used***	***%***
Slices	7693	19
Slice registers	29395	10
Slice LUTs	25684	17
Block RAMs	42	3
DSP48E	149	19

Figure 8 shows OFDM's sensitivity to frequency offsets, and even though the CORDIC Rotate corrects the phase along the symbol, the algorithm lacks the ability to compensate for an ambiguous phase offset present on the constellation, later corrected on the channel estimation.

The BER results for a QPSK modulated signal show that the obtained results are according to theory. No relevant differences can be perceived between a baseband and baseband with IF implementation. Our results show a degradation ranging from 1.8 (SNR=$10^{-3.2}$) to 2.3 (SNR = 0) when the Zero-Forcing equalizer is used which is a feasible result when compared to theory [60]. The time-domain algorithm is also validated; there are no relevant differences between a perfect synchronization simulation and a simulation with Beek's algorithm.

It is possible to do FPGA simulations with a floating-point representation, but not all blocks present on the SysGen libraries allow such precision and operations on floating point have a higher resource usage in hardware. Also, the FE only allows a fixed-point precision. One discrepancy between such precisions can be seen on Figure 9; when the correlation is occurring the angle should be expressed has a constant flat line. However, due to the lower precision brought by fixed-point, there are some inconsistencies on the line. Unlike the system found in [7], our BER results show no relevant degradation between the Matlab floating-point and FPGA fixed-point simulations, however, we are not limiting the registers bit width along the algorithm and the system parameters are different.

The next step is to direct the work presented here towards a 3GPP LTE MIMO-PHY 2x2 layer implementation along with channel encoding and decoding algorithms.

ACKNOWLEDGMENT

The Portuguese projects CROWN, PTDC/EEA-TEL/115828/2009, and CelCop Pest-OE/EEI/LA0008/2011 as well as the Mobile Network Research Group (MOBNET) from the Instituto de Telecomunicações in Aveiro supported the work presented in this paper.

REFERENCES

[1] T. Pereira, M. Violas. A. Gameiro, C. Ribeiro, and J. Lourenço, "Real time FPGA based testbed for OFDM development with ML synchronization", in The Seventh

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

International Conference on Systems an Network Communications (ICSNC 2012), pp. 197-200, Lisbon, Portugal, November 18-23, 2012.

[2] Bo Li, "Analysis and design of Software Defined Radio", in Proceedings of the 2011 International Conference on Internet Computing and Information Services (ICICIS), pp. 415-418, September 2011, doi:10.1109/ICICIS.2011.108.

[3] W. Tuttlebee, "Software Defined Radio: enabling technologies", John Wiley & Sons, 2002.

[4] J. Mitola, "Software radios survey, critical evaluation and future directions", in Telesystems Conference, NTC. National, pp. 13/15 –13/23, IEEE, 1992.

[5] T. Pereira, "Tx/Rx baseband implementation of 802.11-2007 on a FPGA", Masters Thesis, Universidade de Aveiro, 2010.

[6] Alan C. Tribble, "The Software Defined Radio: fact and fiction", in Proceedings of IEEE Radio and Wireless Symposium, pp. 5-8, January 2008, doi:

[7] A. A. Tabassam, F. A. Ali, S. Kalsait, and M. U. Suleman, "Building Software-Defined Radios in MATLAB Simulink – A step towards cognitive radios" in Proceedings of the 2011 UKSim 13th International Conference on Modelling and Simulation, pp. 492-497, 2011 ,doi: 10.1109/UKSIM.2011.100.

[8] W. Zhu, et al., "A real time MIMO OFDM testbed for cognitive radio & networking research", in Proceedings of the 1st International Workshop on Wireless Network Testbeds, experimental evaluation & characterization (WiNTECH), pp. 115-116, September 2006, doi: 10.1145/1160987.1161018.

[9] F. de Dinechin, J. Detrey, O. Cret, and R. Tudoran, "When FPGAs are better at floating-point that microprocessors", in Proceedings of the 16th International ACM/SIGDA symposium on Field programmable gate arrays, pp. 24-26, February 2008, doi: 10.1145/1344671.1344717.

[10] W. Zhu, et al., "Multi-antenna testbeds for research and education in wireless communications", IEEE Communications Magazine, pp. 72-81, vol. 42 issue 12, December 2004, doi: 10.1109/MCOM.20041367558.

[11] A. G. Armada, M. S. Fernández, and R. Corvaja, "Waterfilling schemes for zero-forcing coordinated base station transmission", in Proceedings of the 28th IEEE Conference on Global Telecommunications (GLOBECOM'09), pp. 213-217, 2009, doi: 10.1109/GLOCOM.2009.5425267.

[12] R. Holakouei, A. Silva, A. Gameiro, "Multiuser precoding techniques for a distributed broadband wireless system", Tellecommunication Systems Journal, Special Issue on Mobile Computing and Networking Technologies, Springer, online version published on June 2011, doi: 10.1007/s11235-011-9496-2.

[13] R. Holakouei, A. Silva, A. Gameiro, "Coordinated precoding techniques for multicell MISO-OFDM networks", Wireless Personal Communication (WPC) Journal, Springer, 2013, in press.

[14] R. Zhang, "Cooperative multi-cell block diagonalization with per-base-station power constraints", in IEEE Journal on Selected Areas in Telecommunications – Special Issue on cooperative communications in MIMI celular networks, pp. 1435-1445, vol. 28 issue 9, December 2010, doi: 10.1109/JSAC.2010.101205.

[15] R. W. Chang, "Synthesis of band-limited orthogonal signals for multi-channel data transmission", Bell System Technical Journal 45, 1966, pp. 1775-1796.

[16] ETS 300 401, "Radio broadcasting systems; Digital Audio Broadcasting (DAB) to mobile, portable and fixed receivers", ETSI, Feb. 1995.

[17] K. Kuusilinna, C. Chang, M. J. Ammer, B. C. Richards, R. W. Brodersen, "Designing BEE: a hardware emulation engine for signal processing in low-power wireless applications", in EURASIP Journal on Applied Signal Processing, pp. 502-513, Vol. 2003, January 2003, doi: 10.115/S1110865703212154.

[18] WARP repository, http://warp.rice.edu/trac/browser/ 13.03.2013.

[19] C. Clark, "Software Defined Radio: with GNU Radio and USRP", McGraw-Hill Professional, November 2008.

[20] WARPLab Framework Overview, http://warp.rice.edu/trac/wiki/WARPLab/ 13.03.2013.

[21] WARPnet Measurement Framework, http://warp.rice.edu/trac/wiki/WARPnet/ 13.03.2013.

[22] C. Hunter, J. Camp, P. Murphy, A. Sabharwal, and C. Dick, "A flexible framework for wireless medium access protocols", Invited Paper in Proceedings of IEEE Signals, Systems and Computers Conference, ASILOMAR, November 2006.

[23] K. Amiri, Y. Sun, P. Murphy, C. Hunter, J. R. Cavallaro, and A. Sabharwal, "WARP, a unified wireless network testbed for education and research", in Proceedings of the 2007 IEEE International Conference on Microelectronic Systems Education, pp. 53-54, June 2007, doi: 10.1109/MSE.2007.91.

[24] P. Murphy, C. Hunter, and A. Sabharwal, "Design of a cooperative OFDM transceiver", in Proceedings of the 43rd ASILOMAR Conference on Signals, Systems and Computers, pp. 1263-1267, November 2009.

[25] P. Murphy, and A. Sabharwal, "Design, implementation and characterization of a cooperative communications system", in IEEE Transactions on Vehicular Technology, vol. 60, July 2011, doi: 10.1109/TVT.2011.2158461.

[26] P. Murphy, A. Sabharwal, and B. Aazhang, "On building a cooperative communication system: testbed implementation and first results", EURASIP Journal on Wireless Communications and Networking, June 2009, doi:10.1155/2009/972739.

[27] WARPMAC, http://warp.rice.edu/trac/wiki/WARPMAC/

[28] G. Wang, B. Yin, K. Amiri, Y. sun, M. Wu, and J.R. Cavallaro, "FPGA prototyping of a high data rate LTE uplink baseband receiver", in Proceedings of the 43rd ASILOMAR Conference on Signals, Systems and Computers, pp. 248-252, November 2009.

[29] GNU Radio, http://gnuradio.org/, 2013.

[30] USRP - Universal Software Radio Peripheral, http://www.ettus.com, 13.03.2013.

[31] FlexRadio Systems, http://www.flex-radio.com/, 2013.

[32] HPSDR - High Performance Software Defined Radio, http://openhpsdr.org/index.php, 13.03.2013.

[33] AMRAD Charleston SDR project, http://www.amrad.org/projects/charleston_sdr, 13.03.2013.

[34] M. Majó, "Design and implementation of an OFDM-based communication system for the GNU radio platform", Master Thesis, Dec. 2009.

[35] A. Marwanto, M. A. Sarijari, N. Fisal, S. K. S. Yusof and R. A. Rashid, "Experimental study of OFDM implementation utilizing GNU Radio and USRP – SDR", Proc. of the IEEE 9th Malaysia International Conference on Communicatons, Dec. 2009, pp. 132-135.

[36] R. K. Ganti, et al., "Implementation and experimental results of OFDM-based superposition coding on software radio", IEEE International Conference on Communications (ICC'10), pp. 1-5, May 2010, doi: 10.1109/ICC.2010.5502330.

[37] J. Garcia and R. Cumplido, "On the design of an FPGA-based OFDM modulator for IEEE 802.11a", 2nd International

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Conference on Electrical and Electronics Engineering", Sept. 2005, pp. 114-117.

[38] E J. Garcia and R. Cumplido, "On the design of an FPGA-based OFDM modulator for IEEE 802.16-2004", 2005 International Conference on Reconfigurable Computing and FPGAs, 2005, pp. 22-25.

[39] P. Moose, "A technique for Orthogonal Frequency Division Multiplexing frequency offset correction", IEEE Transactions on Communications, vol. 42 no. 10, pp. 2908-2914, October 1984.

[40] Jan-Jaap van de Beek, M. Sandell, and P. O. Börjesson, "ML estimation of time and frequency offset in OFDM systems", IEEE Transactions on Signal Processing, vol. 45, no. 7, July 1997.

[41] T.M. Schmidl and Cox, and D. C. Cox, "Robust frequency and timing synchronization for OFDM", IEEE Transactions on Communications, vol. 45, pp. 1613-1621, December 1997.

[42] R. Negi, and J. Cioffi, "Pilot tone selection for channel estimation in a mobile OFDM system", Journal IEEE Transactions on Consumer Electronics, pp. 1122-1128, vol. 44 issue 3, August 1998, doi: 10.1109/30.713244.

[43] I. Barhumi, G. Leus, M. Moonen, "Optimal Training Design For Mimo–Ofdm Systems in Mobile Wireless Channels", IEEE Transactions on Signal Processing, vol. 51 no. 6, pp. 1615–1624, June 2003.

[44] P. Hoeher, S. Kaiser, P. Robertson, "Two-dimensional pilot-symbol-aided channel estimation by Wiener filtering," in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 1845-1848, April 1997.

[45] S. Kaiser, P. Hoeher, "Performance of multi-carrier CDMA systems with channel estimation in two dimensions," in Proc. IEEE Personal, Indoor and Mobile Radio Communications Symposium, pp. 115-119, Helsinki, Finland, September 1997.

[46] Y. Li, "Pilot-symbol-aided channel estimation for OFDM in wireless systems," IEEE Transactions on Vehicular Technology, Vol. 49, Issue 4, pp.1207-1215, July 2000.

[47] S. Boumard, A. Mammela, "Channel Estimation Versus Equalization in an OFDM WLAN System," in Proc. IEEE Vehicular Technology Conference, vol. 1, pp. 653–657, Rhodes, Greece, May 2001.

[48] M. Shin, H. Lee, C. Lee, "Enhanced Channel Estimation Technique for MIMO–OFDM Systems," IEEE Transactions on Vehicular Technology, vol. 53, no. 1, pp. 261–265, Jan. 2004.

[49] A. Chini, "Multicarrier modulation in frequency selective fading channels," Ph.D. dissertation, Carleton University, Canada, 1994.

[50] J. Rinne, M. Renfors, "Pilot spacing in Orthogonal Frequency Division Multiplexing systems on practical channels," IEEE Transactions on Consumer Electronics, vol. 42 no. 3, pp. 959 – 962, November 1996.

[51] S. Coleri, M. Ergen, A. Puri, A. Bahai, "Channel Estimation Techniques Based on Pilot Arrangement in OFDM Systems," IEEE Transactions on Broadcasting, vol. 48, no. 3, pp. 223–229, Sept. 2002.

[52] C. Athaudage, A. Jayalath, "Low–Complexity Channel Estimation for Wireless OFDM Systems," in Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 1, pp. 521 – 525, Beijing, China, Sept. 2003.

[53] S. Coleri, M. Ergen, A. Puri, A. Bahai, "A Study of Channel Estimation in OFDM Systems," in Proc. IEEE Vehicular Technology Conference, vol. 2, pp. 894 – 898, Vancouver, Canada, Sept. 2002.

[54] X. Wang, K. Liu, "OFDM Channel Estimation Based on Time-Frequency Polynomial Model of Fading Multipath Channel," in Proc. IEEE Vehicular Technology Conference, vol. 1, pp. 460 – 464, Atlantic City, USA, Oct. 2001.

[55] A. Dowler, A. Doufexi, A. Nix, "Performance Evaluation of Channel Estimation Techniques for a Mobile Fourth Generation Wide Area OFDM System," in Proc. IEEE Vehicular Technology Conference, vol. 4, pp. 2036 – 2040, Vancouver, Canada, Sept. 2002.

[56] S. Lee, D. Lee, H. Choi, "Performance Comparison of Space-Time Codes and Channel Estimation in OFDM Systems with Transmit Diversity for Wireless LANs," in Proc. Asia-Pacific Conference on Communications, vol. 1, pp. 406 – 410, Penang, Malaysia, Sept. 2003.

[57] O. Font-Bach, N. Bartzoudis, A. Pascual-Iserte, and D. L. Bueno, "A real-time MIMO-OFDM mobile WiMAX receiver: architecture, design and FPGA implementation", Computer Networks, 55 (16), pp. 3634-3647, 2011.

[58] Xilinx Inc., "DS249 LogiCORE IP CORDIC v4.0", http://www.xilinx.com/support/documentation/ip_documentation/cordic_ds249.pdf, 13.03.2013.

[59] Xilinx Inc., "Put hardware in the loop with Xilinx System Generator for DSP", Xcell Journal, Fall 2003.

[60] C. Ribeiro, "Channel and frequency offset estimation schemes for multicarrier systems", PhD Thesis, Universidade de Aveiro, 2010.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Comparison of Single-Speed GSHP Controllers with a Calibrated Semi-Virtual Test Bench

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

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

Abstract — With the recent development of new controllers for heat pump systems, there is a need to test and compare these controllers in a realistic and reproducible environment. This can be done using a semi-virtual test-bench with a simulation environment that is calibrated with in-situ measurements. A real ground source heat pump (GSHP) is connected to the test bench that emulates the building and the boreholes. The test can thus be carried out under dynamic conditions: dynamic weather conditions are used as well as simulated building, floor heating and boreholes. In this study, the developed neural network-based predictive controller is compared to a conventional controller during a one-week semi-virtual test. Test results showed that the predictive controller can provide up to 40% energy savings in comparison with a conventional controller.

Keywords - Artificial neural networks; Predictive control; Energy savings; Geothermal heat pump, Semi-virtual test-bench.

I. INTRODUCTION

Important research was conducted on predictive control strategies during the 1980s and 1990s. More recently, the use of artificial neural networks (ANN) has significantly increased the prediction performances of models. ANN models were successfully applied to the control of residential and small office buildings [1-4]. Other kinds of predictive controllers for radiant floor heating systems have also led to remarkable results [5-8].

Most of these smart controllers were validated by simulation, while some were tested on a real building or on a test cell. Each test technique has its advantages and its disadvantages. The simulation test is required to optimize the controller and to ensure its accurate behavior in various situations. Nevertheless, a simulated environment may not be realistic enough to produce reliable results. Besides, this procedure uses a simulated heat pump. To remedy that situation, the controller can be tested in-situ on a real building or on a test cell. These approaches allow the use of a real heat pump and deals with real noisy data. The main problem of these tests is the fact that two controllers can only be tested sequentially. Even if weather compensation techniques can be done, the comparison generally fails since the conditions (occupants' behavior, weather, etc.) are different. Another comparison technique, called cross-comparison, consists in testing two controllers at the same time but on separate blocks of the same building. Again, the comparison is not accurate since the two blocks can have different internal and external heat gains, orientation or wall composition.

For the purpose of comparing different controllers sequentially and under identical conditions, the semi-virtual test bench PEPSY-PAC [9] developed by the CSTB is used. A real GSHP is connected to a test bench that emulates the building and the boreholes. The test of the controllers can thus be carried out under dynamic conditions: dynamic weather conditions are used as input of a building simulation including floor heating and boreholes. This approach opens a large variety of possible test schedules since the simulated building, the emitter, weather conditions and occupancy can be changed easily. Moreover, the semi-virtual test allows the comparison of different controllers with the same solicitations.

In this paper, the developed ANN predictive controller is compared to a conventional controller during two sequential semi-virtual tests of one week. The simulation environment is designed to reproduce all characteristics (building, weather, boreholes, etc.) of an in-situ GSHP that was monitored during the 2011/2012 heating season in the north of France. The system components parameters (boreholes, GSHP, floor heating and building) are first identified separately then the global simulation with all the components is compared to in-situ measurements.

The paper also includes the description of the ANN controller. The training process including the determination of optimal input data, algorithm, and structure is detailed. The objective of the controller is to minimize the energy consumption of the GSHP system and maintain a good comfort level anticipating future disturbances (solar gains, outdoor temperature) and room temperature. ANN modules are used for the prediction of weather data, room temperature and temperatures in the floor heating and in the boreholes.

The paper is organized as follows. In Section II, the semi-virtual test bench is presented. Section III deals with the calibration of the simulated part with in-situ measurements. The ANN controller is detailed in Section IV. In Section V, the predictive controller is compared to a conventional controller on the bench. The last section presents the conclusions of this paper.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

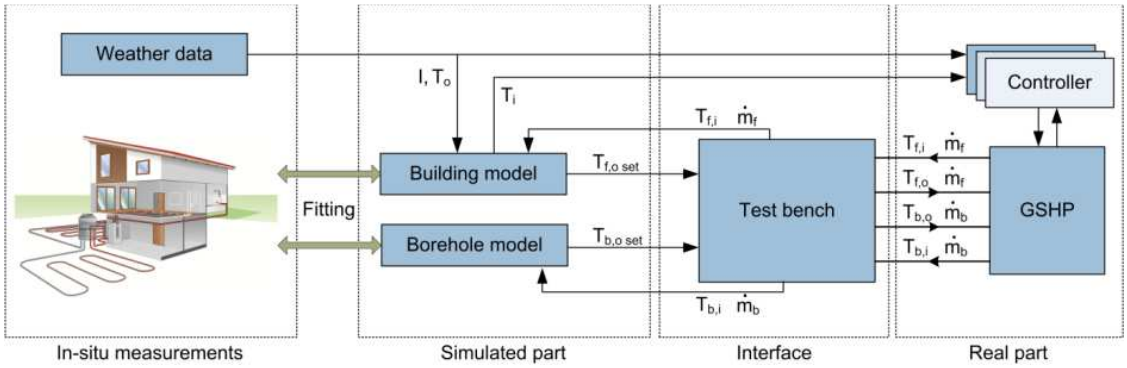


Figure 1: Flowchart of the semi-virtual test of a controller.

II. SEMI-VIRTUAL TEST BENCH

A. Concept of the test-bench

The semi-virtual platform PEPSY-PAC (Platform for the Evaluation of Performances of dynamic SYstems) has been developed for testing performances of GSHP systems or parts of the system [9]. It also allows the test of a controller connected to a real GSHP integrated in a simulated environment, as presented in this paper. This test bench allows the emulation of any water-based heat emitter integrated in a building as well as any kind of ground heat exchanger. The outlet temperature and flowrate of the test bench is controlled by the system simulation.

Matlab is used for the simulated part of the test bench. Simulation is therefore slowed down to real time and the simulation environment enables at the same time the test bench control, system simulation (emulator) and online monitoring of the test.

The operation of the test bench is detailed in Figure 1. Every thirty seconds, the simulated part sends model outputs (outlet temperatures of the floor heating $T_{f,o\text{-}set}$ and the boreholes $T_{b,o\text{-}set}$) to the test bench.

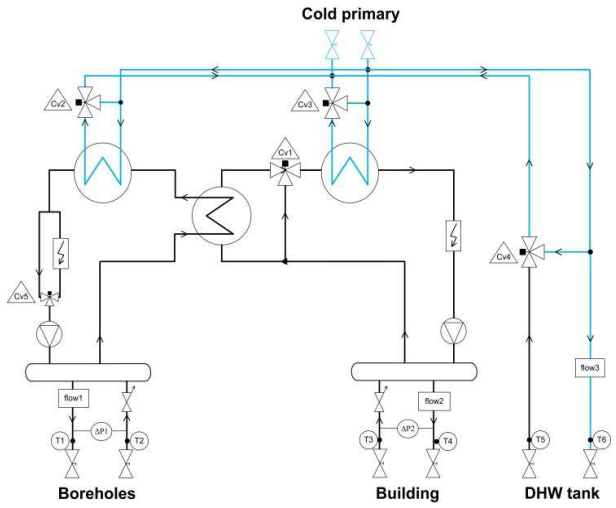


Figure 2: Test bench hydraulic circuit diagram.

The test bench controls the real outlet temperatures of the GSHP ($T_{f,o}$ and $T_{b,o}$) to reach these setpoints. At the same time, the GSHP inlet temperatures ($T_{f,i}$ and $T_{b,i}$) and flow rates ($\dot{m}_f$ and $\dot{m}_b$) are measured and sent to the simulation environment. Weather data like solar radiation I and outside temperature T_o as well as room temperature T_i are transmitted to the tested controller. In-situ measurements, detailed in the next section, are used to fit the simulated part.

B. Construction and control

The test bench integrates 6 hydraulic ports for testing (building, boreholes and Domestic Hot Water tank) as well as 2 hydraulic ports for the cold primary circuit. The DHW tank ports are not used for this test. The circuit diagram is presented in Figure 2.

Seven proportional-integral-derivative (PID) controllers ensure the continuous control of outlet temperatures through the action of hydraulic valves and electric heaters. Figure 3 shows the temperature step responses on the building side and in the boreholes. Inlet and outlet temperatures are measured every thirty seconds with a specific datalogger. The test bench was designed to consume the less possible energy: the heat extracted at the building side is used to heat up the boreholes side. Two hydraulic separators on the building side and on the borehole side allow the heat pump flowrate to be independent from the bench flowrate. The pressures losses of the heat pump circulators can thus be adjusted to correspond to real floor heating and boreholes.

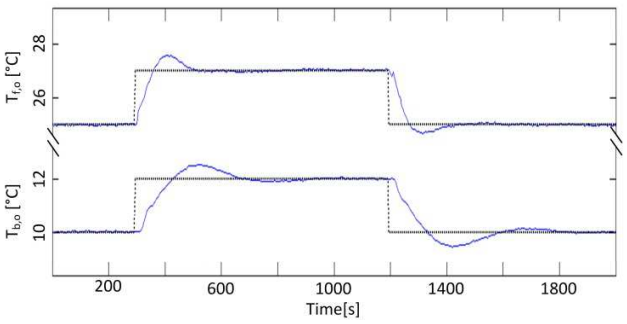


Figure 3: Test-bench response to setpoint step changes.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

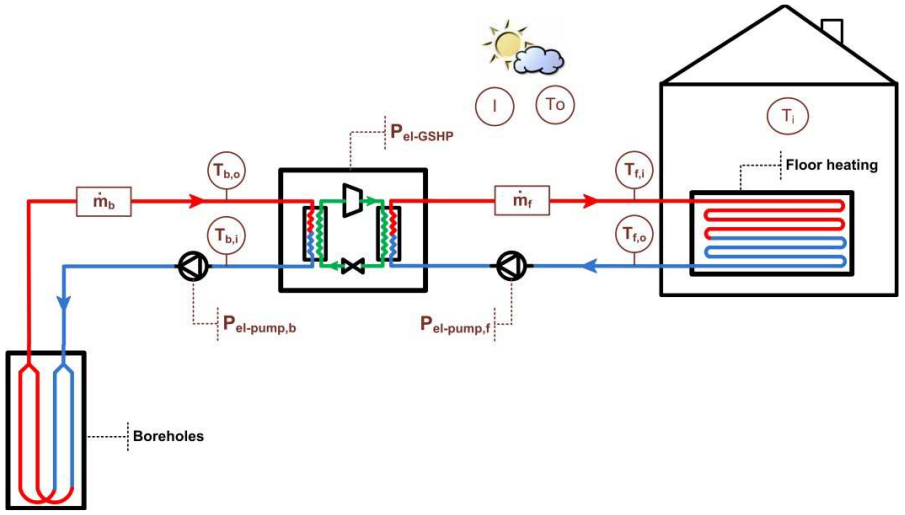


Figure 4 : In-situ monitoring of a GSHP system on a dwelling in Marck (France).

III. CALIBRATION OF SIMULATED PART

A. *In-situ measurements*

A single family-house located in Marck (France) has been monitored during the 2011/2012 heating period. The dwelling is conform to the 2005 French regulation (RT2005) and has the following characteristics:

- Surface area of 100 m^2.
- External walls: brick (11 cm), air layer, cellular concrete (11 cm), glass-wool (10 cm), air layer, plasterboard (1.3 cm). Global U-value of 0.18 W.m-2.K-1 ;
- Double glazing, U-value of 1.5 W.m-2.K-1 ;
- Windows distribution: North 7%, South 10%, East 17%, West 0% ;
- Single flow hygro-adjustable ventilation ;
- Equipped with a 8.5 kW GSHP connected to a floor heating;
- Double U-pipe vertical boreholes of 100m depth.

The renewable energy monitoring box (REMBO) developed by the CSTB acquires, treats and sends measured data every minute to a server. Flow rates and temperatures on the building side and on the borehole side are measured as well as electric consumptions of compressor and pumps. Outside and room temperature are also measured. Global horizontal solar radiation is obtained from satellite images thanks to the SODA service [10].

B. *Modeling of the GSHP system*

The whole system model is based on Matlab/Simulink environment using the SIMBAD toolbox (Simbad, 2004). The system includes the following components (

Figure 5):

- Building part (building, floor heating system, occupants, ventilation and equipment);
- GSHP;
- Borehole heat exchanger part.

The building was modeled with the Simbad multizone model [11] and designed with the associated SimBDI graphical interface. A simple monozone model has been chosen.

The floor heating model developed by Salque [12] is based on finite difference method. It consists in a 2D-grid of the slab coupled to a pipe model. The floor heating is made of four layers (floor covers, slab with pipes, insulation and concrete floor) with different thermal properties.

The heat pump model is based on experimental data. The coefficient of performance (COP), which is the ratio of the heat produced at the condenser to the electric energy consumed by the compressor, is determined with the method of least squares for a plane equation, depending of average temperatures at both condenser and evaporator side.

The boreholes model developed by Partenay [13] is based on finite difference method. It consists in a 3D-grid of the ground coupled to a pipe model, allowing the modeling of single or double U pipes. The heat conduction problem is solved with a state-space formulation.

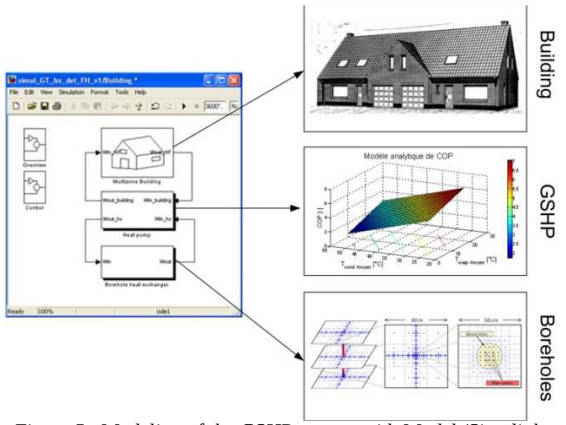


Figure 5 : Modeling of the GSHP system with Matlab/Simulink.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

C. Fitting of simulated part

The objective is to fit the simulated GSHP system to the measured data to obtain a realistic simulation environment. The system components parameters (boreholes, GSHP, floor heating and building) are identified separately. For each component, the physical parameters known a priori were fixed, while others were fitted by least square minimization. A step by step method for tuning the physical parameters of the different models was proposed by Salque [12]. A specific iterative process for parameters identification of building and floor heating was developed since these components are physically coupled. An overview of this method is detailed here, for more information please refer to [12].

- Boreholes parameters identification

Design parameters such as the radius of drilling, borehole length or pipe diameter are fixed since they are known from in-situ measurements. Modeling parameters such as the radius of domain and the number of nodes are also fixed to simplify the problem. The unknowns concern the thermal characteristics of the ground (ground conductivity and heat capacity) and the initial ground temperature. These variable parameters were adjusted in a physical range of values to best fit the measured data. The following values were found to be the optimal set of parameters:

- Ground conductivity : 2.2 W/(m.K)
- Ground heat capacity: 2180 kJ/(kg.K)
- Initial ground temperature : 12.2°C

The Root Mean Square (RMS) error on outlet temperature with the optimal set of parameter is 0.41°C. The error in terms of energy extracted from the ground during the month of March is lower than 1%.

- Floor heating and building parameters identification

The building was modeled with the Simbad multizone model [11] and designed with the associated SimBDI graphical interface. Geometry and wall compositions of the identified dwelling were read from plans. Due to a large number of unknowns related to the occupants' behavior (windows opening, internal gains, etc.) and the exact location of the room temperature sensor, a simple monozone model has been chosen. Design parameters such as building geometry, wall composition or floor heating surface are supposed to be perfectly known and fixed. The real hygro-adjustable ventilation is modeled by simple-flux ventilation with a constant air flow as humidity ratio of indoor air is unknown.

Since internal gains and ventilation parameters compensate when trying to fit the building model, internal gains were fixed to a typical value while the ventilation rate was estimated. A constant blinds position between 0 (closed) and 1 (open) was also estimated to fit the solar gains. The composition of floor heating layers is known in a range of uncertainty. It was found that the adjustment of the most influent layer (slab with pipes) is enough to make the model fit. Another crucial floor heating parameter that needs to be adjusted is the pipe spacing that is proportional to the heat-exchange surface between fluid and floor heating

Since there are no measurements of surface temperature, the identification of both floor heating and building models has to be made in parallel. The optimal set of parameters was found to be:

- Pipe spacing : 0.33 m ;
- Floor heating conductivity : 1.9 W/(m.K) ;
- Floor heating inertia : 8950 kJ/K ;
- Ventilation rate : 0.36 vol/h;
- Blinds position: 0.8 [-].

- GSHP parameters identification

The GSHP model is only required to verify that the global simulation still fits the measured data. The heat pump COP is modeled by the following function, developed by Partenay [13]:

$$COP = a * T_{evap} + b * T_{cond} \quad (1)$$

where T_{evap} and T_{cond} are the average temperatures at evaporator and condenser side. For a given temperature level in the heating floor, COP behaves as a linear function of the temperature level in the ground. Experimental tests revealed that electric power Pel was only a function of condenser temperature. The chosen model is expressed as follows:

$$P_{el} = d * {T_{cond}}^2 - e * T_{cond} + f \quad (2)$$

The coefficients *a, b, c, d, e, f* are identified using the least squares method (*a*=5.09, *b*=0.16, *c*=-0.05, *d*=-81.9, *e*=66.9, *f*=-0.55).

- Global simulation results

The identified models are now integrated in a global simulation in Matlab/Simulink. The month of March is simulated and compared to the measured data. The measured heat pump on/off control is applied to the simulated heat pump. This way the differences between simulation and measurements are only due to the modeling and cannot be attributed to an incorrect estimate of control logic. Besides, the action of the occupants on room temperature setpoint makes it very difficult to accurately estimate the control logic.

Figure 6 shows the comparison between simulated and real GSHP system. The first graph on top shows simulated and measured room temperatures. The identification of the thermal behavior of the building is satisfactory. Indeed, simulated and measured room temperature extremum are in phase. The RMS error on room temperature over the whole month is 0.63 °C. The RMS error is 26 W for condenser power and 18 W for evaporator power. Simulated heating energy consumption is 558 kWh while measured consumption is 541 kWh. .

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

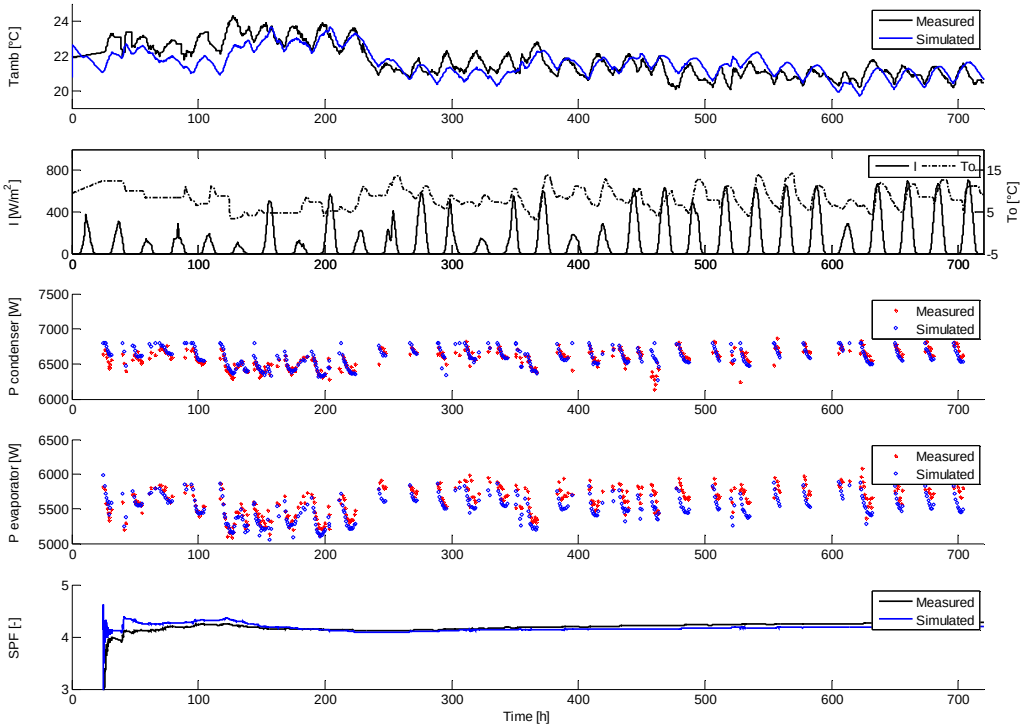


Figure 6 : Comparison of global simulation results and in-situ measurements – Month of March.

The last graph shows the SPF, which is the ratio between heating energy delivered to the building and electric energy consumed by the compressor. The SPF over the month of March obtained by simulation is 4.28, while the real SPF is 4.21.

IV. THE PREDICTIVE CONTROLLER

The objective of the controller is to minimize the energy consumption of the GSHP system and maintain a good temperature level anticipating future disturbances and room temperature. The controller is designed to be self-learning and easily adaptable in practice.

To be compatible with the developed controller, the GSHP system must fulfill the following conditions:

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

A. Controller strucutre

The modular structure of the controller is illustrated in Figure 7. The forecasting modules are all based on ANN. A weather module performs predictions of solar radiation (I) and outdoor temperature (T_o). The heating power produced (P_h) and the electric power consumed by the GSHP (P_{el}) are predicted by another module. The latter uses as inputs the supply and returns temperatures in the boreholes (T_b) and in the radiant floor (T_f), as well as all the possible trajectories of the GSHP on/off for the next 6 hours. Based on these predictions, another ANN makes predictions of room temperature T_i. The optimization block determines the optimal trajectory to be applied to the system according to the various trajectories of T_i and P_{el}.

B. Control strategy

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

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

$$\text{subject to} \quad T_{min} < \widehat{T_i}(k) < T_{max} \qquad (4)$$

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

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

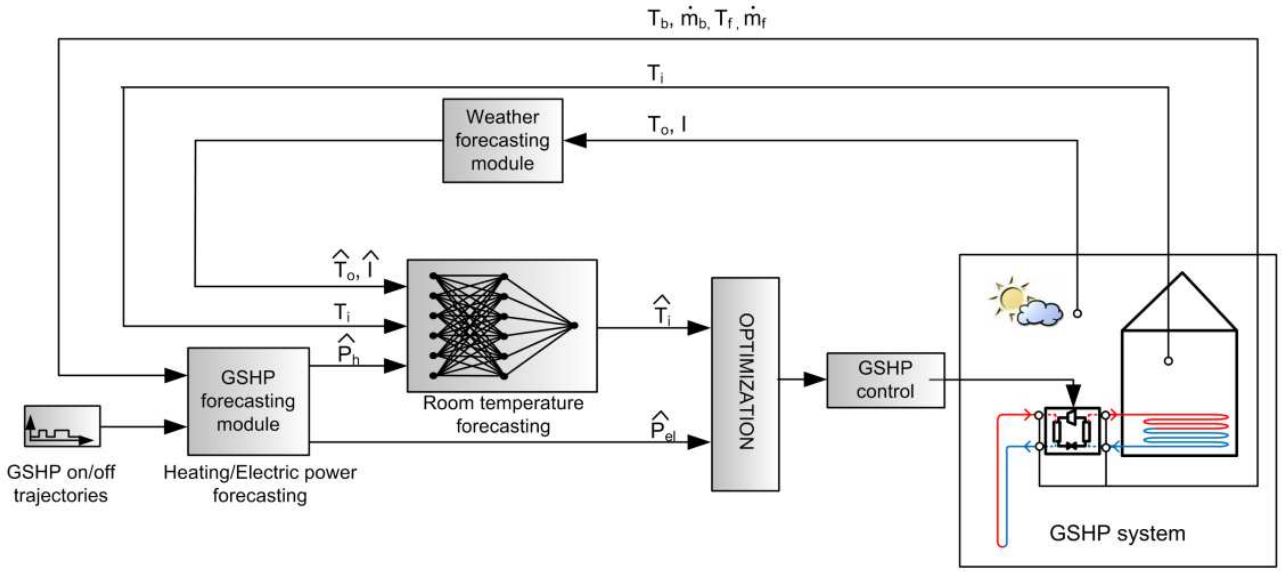


Figure 7: Flow chart of the ANN-based predictive controller. The symbol (^) is assigned to the predicted values.

C. Prediction horizon

The length of the prediction horizon depends on several factors. A large horizon is needed when large room temperature or electricity price changes are expected in the future [14]. It is the case in an intermittently occupied building. In practice, the horizon length is chosen as an equivalent of the room time constant corresponding to the first active layers of the walls. For the purpose of the present study, a 6 hours receding horizon is applied and the optimal control problem is repeated every 15 minutes.

D. Algorithm

At each time step, the optimal on/off trajectory for the next 6 hours is determined. The discrete nature of the input makes it possible to compute all the possible trajectories and chose the one that minimizes the cost function (3) subject to constraint (4). Moreover, it allows the use of non-linear models, such as ANN, that usually limit the possibilities of analytical problem solving [15].

E. ANNs training process

The various modules were first optimized via extensive off-line tests conducted with the neural network toolbox in Matlab [16]. The objective is to produce a network that fits the data as accurately as possible, but simple enough to train easily and generalize well. Optimization is an iterative process that consists in finding the ideal ANN structure, algorithm and set of input variables.

The ANNs architecture is a multilayer perceptron. In the present study, one hidden layer was always found to be the best solution. The number of neurons in the hidden layer was first chosen to be equal to 75% of the number of inputs [17] and then optimized by trial-and-error until no improvement could be seen.

Another key step in the process of ANN building is the choice of inputs and associated time delays. For nonlinear models such as ANN, there is no systematic approach [18] and the risk of dismissing relevant inputs is high. Statistical methods like auto-correlation criterion or cross correlation give a good insight into the relevance and the lag effect of an input variable on the output. The model has to be as simple as possible while taking into account the most relevant inputs. Again, optimal sets of inputs and time delays are obtained by trial-and-error. A hyperbolic tangent sigmoid function was used as the transfer function in the single hidden layer. The algorithm used for training was an optimized version of the Levenberg-Marquardt algorithm that included Bayesian regularization. This algorithm minimizes a combination of squared errors and weights, and then determines the correct combination so as to produce a network that generalizes well.

The generalization capability is also improved with the early stopping feature. With this technique, the collected data that was first normalized to the range [-1; 1] is divided into three subsets: training, validation, and test. Training stops when validation performance has increased more than 5 times since the last time it decreased. The test data set is used to estimate the generalization error of the ANNs but does not interfere during the training process.

For online applications, ANNs have to be trained regularly on new data set to adapt to changes in the system. For instance, during the heating season, the boreholes temperature will fall. To take into account this phenomenon, studies not presented here showed that the ANN for borehole temperature prediction has to be trained every 15 days on the last 30 days data.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

F. Room temperature prediction

ANN for room temperature prediction is here detailed as this module is of most interest. For more information on the other ANN modules, please refer to [19].

- *Choice of inputs*

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

- *Optimal structure*

The developed ANN provides room temperature T_i for the next time step from current weather data (T_o, I) as well as previous and current values of heating power P_h and room temperature T_i. This ANN making the link between the heating power delivered to the radiant floor and the impact on room temperature, it encapsulates both the thermal behavior of the building and the emitter. In particular, the thermal lag of the radiant floor is taken into account in the ANN using P_h(k-1). A wide range of current and previous values of these variables was tested as inputs. The optimal ANN structure and set of inputs for room temperature prediction of the studied building are presented in Figure 8.

Offline tests revealed that the mean value of the outdoor temperature on the last 24 hours $\overline{To_{24}}(k)$ contains enough information to describe the dynamic behavior of the tested building. For less insulated buildings or buildings with a higher ventilation rate, the impact of the outdoor temperature is higher and the current value of T_o is likely to be more appropriate. The ANN used in this module has 6 input neurons, one hidden layer of 6 neurons and one output neuron.

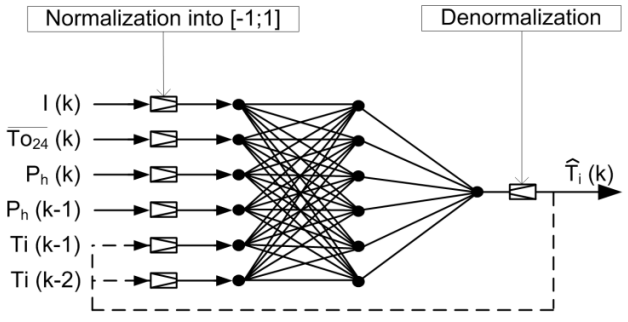


Figure 8: ANN architecture for room temperature prediction.

- *Comparison with ARX model*

ANN performances for room temperature prediction are compared to linear ARX models, which are commonly used for the building model in predictive control. ARX models are Auto Regressive models with eXternal inputs that can be written as follows:

$$y(t) = B * [u(t-1), u(t-2) \dots] + A * [y(t-1), y(t-2) \dots] + A * \varepsilon(t) \quad (5)$$

where $y(t)$is the output vector, $u(t)$ the input vector and $\varepsilon(t)$ a white noise with zero mean.

Three months of simulation were used to train and test the models: January and February data are used for training and validation of ANN and ARX models, while March is used for test. A wide range of inputs were tested. To evaluate the prediction error of ANN and ARX models, the root mean square error (RMSE) and the mean error (ME) were used as performance criteria over the 6 hours prediction horizon. The main results are summarized below:

- ANN models clearly outperform ARX models in terms of ME and RMSE over the whole prediction horizon. The RMSE is in average 40% lower using non-linear ANN models. ANN forecasts are less biased as the ME is smaller in absolute value.
- Too complicated models do not give accurate results.
- Previous values of heating power P_h(k-1) as well as room temperature T_i(k-1) and T_i(k-2) must be taken into account due to the inertia of the building and the floor heating.
- Taking into account previous values further into the past does not improve the prediction performances of both types of models.

An example of 3 hours prediction results of ANN3 and ARX3 models on a representative week of March is given in Figure 9. ANN model reproduces more accurately the thermal behavior of the building in comparison to the linear ARX model. ANN is in particular much better when the building is subject to strong solar gains (first day of Figure 9).

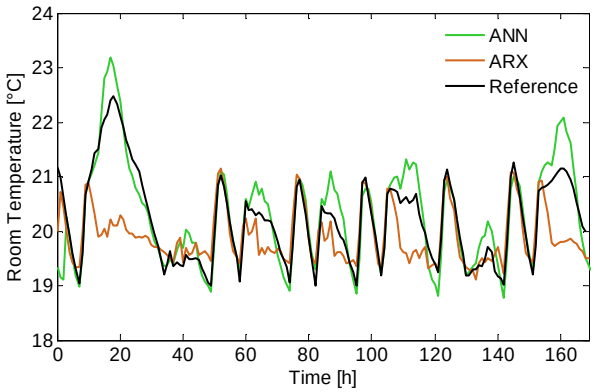


Figure 9: 3 hours prediction of room temperature.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

V. COMPARISON OF CONTROLLERS ON THE SEMI-VIRTUAL TEST BENCH

A. Conventional controller

For the test, the real measured controller output is used as a reference. This on/off signal is applied to the heat pump connected to the bench. It can be noticed that the heat pump installed in the laboratory is the same heat pump of that in the monitored dwelling. This reference controller is a Compensated-Open-Loop (COL) controller that is installed by default with most single-speed GSHP systems. The COL controller is based on the following heating curve that is adjusted with the actual value of room temperature:

$$T_{HC} = (a * T_o + b) - c * (T_i - T_r) \quad (6)$$

where T_o is the outdoor temperature and $(T_i - T_r)$ the difference between the actual and the setpoint room temperature. The coefficients a and b are the heating curve parameters while c is the ambient compensation factor. The COL controller switches on/off the GSHP when the water supply temperature $T_{f,s}$ is beyond T_{HC} ±2°C. This control logic requires the pump on the building side to always be working to keep the fluid circulating. The COL controller is represented in Figure 10.

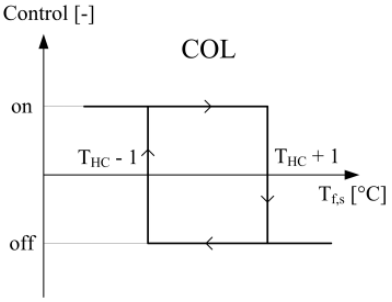


Figure 10 : Control logic of the COL conventional controller.

B. Experiment process

- *Test procedure*

The ANN controller is compared to the COL controller during two sequential tests of one week on the bench. The complete test procedure is illustrated in Figure 11. The procedure starts with an initialization phase from February 15th to March 15th that consists in a simulation of the whole system. During this phase, the measured on/off signal is applied to the simulated heat pump. Initialization period is also required to train the ANN modules of the predictive controller: the training data set is from February 15th to February 28th while the validation data set is from February 29th to March 15th. At the end of the initialization, the real-time testing of the controller starts. The simulated building and boreholes are in the same thermal state at the beginning of each test to ensure an accurate comparison.

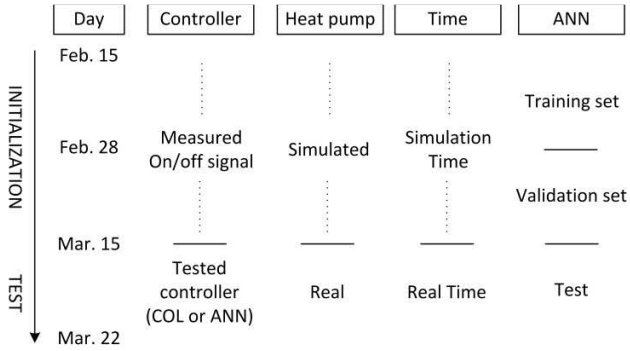


Figure 11 : Procedure of the semi-virtual test of the controllers.

Since the real GSHP has a very small time constant (the steady-state of the heat pump is almost immediately reached), the real-time testing can in fact be accelerated to significantly reduce the duration of the test. The acceleration factor of real-time depends on the minimum duration of a compressor cycle during the test as well as the response time of the bench. In our case, the bench approximately takes 3 minutes to reach the setpoint ± 0.5°C when the compressor starts. With the ANN controller, the minimum duration of a compressor cycle is 15 minutes (time lapse between 2 controller's calls). With the conventional controller, in-situ measurements showed a minimum of 12 minutes per cycle. Based on these durations, the real time has been accelerated by 2 to ensure the bench to accurately control the temperatures.

- *Heat pump control*

The heat pump is controlled via programmable resistances that replace the heat pump outdoor and room temperature sensors. An outdoor temperature drop activates the heat pump compressor, and vice versa. This way the control of the heat pump is non-intrusive.

C. Controllers' performances comparison

Room temperature setpoint of ANN controller is set to 22.5°C with a comfort parameter $\Delta T_{max} = 0.5°C$. This temperature corresponds to the mean room temperature observed with the conventional COL controller.

A comparison of the controllers on the test week is depicted in Figure 12. COL controller leads to small room temperature overshoots in the afternoon. It can be noticed that when the GSHP is switched on in the morning of a sunny day, the dwelling is likely to be overheated in the afternoon. This is of course due to the fact that the conventional control logic does not integrate a prediction of solar gains.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

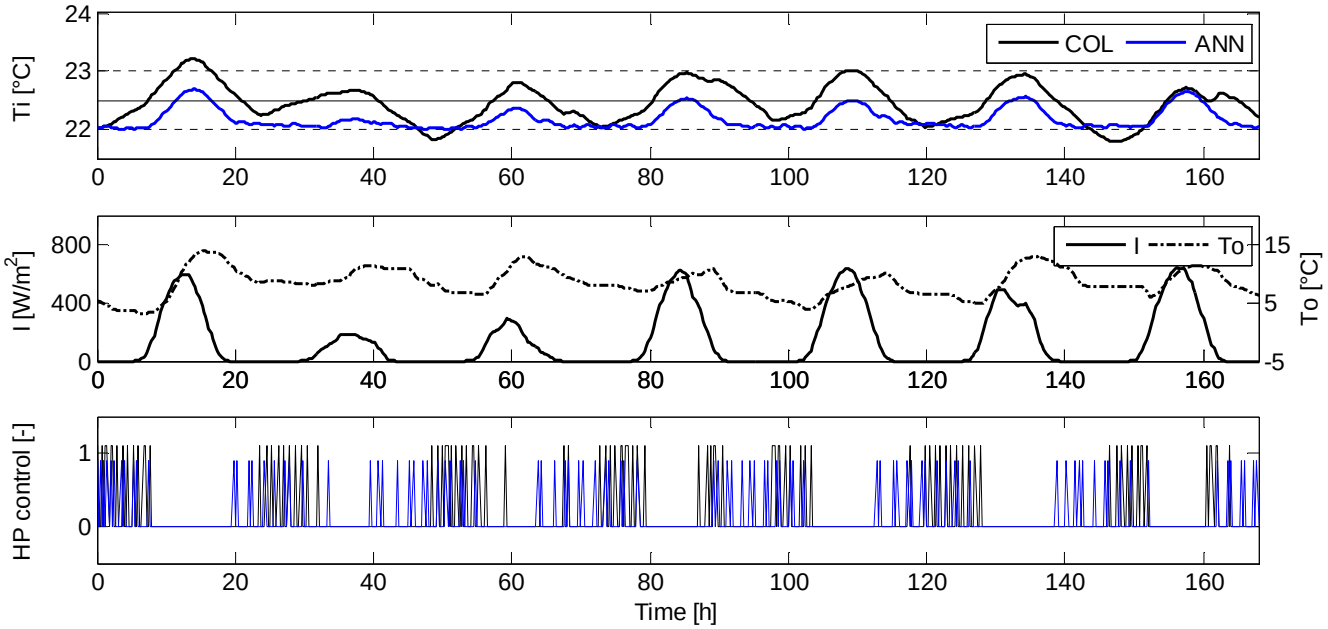


Figure 12 : Comparison of the controllers over the test week. % 15-22 March.

ANN controller keeps room temperature in the comfort range thanks to its prediction capability. Room temperature is lowered just before solar gains are expected so that to avoid overheating and benefit from free heat gains, leading to energy savings. Heating loads are thus shifted to anticipate solar gains.

Results in terms of energy consumed and heat pump performances over the test week are presented in Figure 13. Thermal energy delivered to the floor heating is 152 kWh with COL and 147 kWh with ANN, i.e., a gain of 3%. Total electric energy consumed by the GSHP system is 60 kWh with COL whereas ANN controller only consumes 36 kWh. This gain of 40% in energy consumption is mainly due to the fact that the pump on the floor heating side is constantly running with COL.

Heat pump efficiency is expressed here as a Seasonal Performance Factor (SPF), which is the ratio between the energy delivered by the heat pump and the electrical energy consumed by the compressor or by the compressor and the pumps (global SPF). The compressor SPFs are almost identical with both controllers. The ANN compressor SPF is slightly higher (4.6) than COL (4.5) as mean duration of compressor cycles is lower with ANN. Longer cycles indeed lead to higher temperatures in the floor heating and thus a lower heat pump efficiency. Global SPF with COL is only 2.5 because of the high consumption of the pump on the building side, while global SPF with ANN is 3.9.

VI. CONCLUSION

For the purpose of comparing different controllers sequentially and under identical conditions, a test procedure has been developed on a calibrated semi-virtual test bench. A real GSHP has been connected to the test bench that emulates the building and the boreholes.

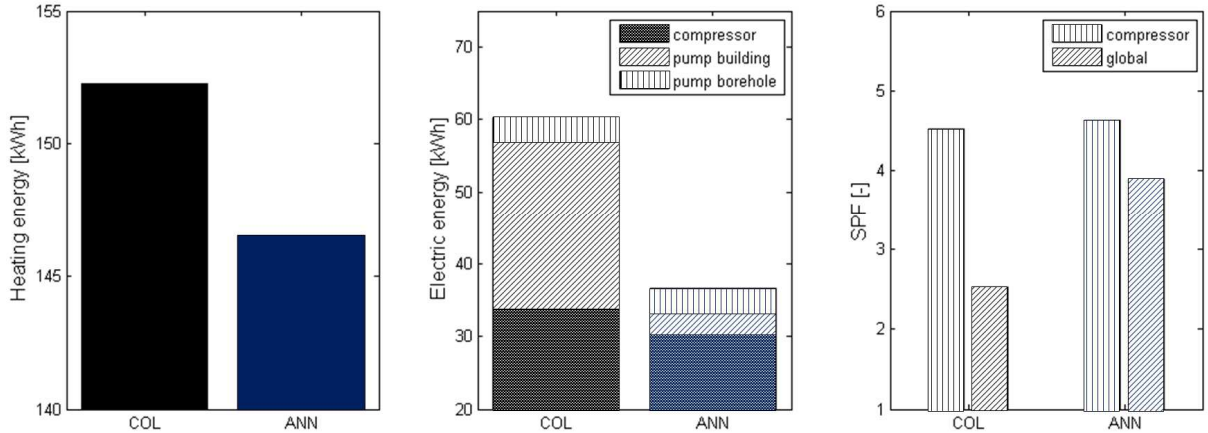


Figure 13 : Test results in terms of energy consumption and heat pump efficiency SPF% over the testing week.

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The controllers' tests can thus be carried out under dynamic conditions: dynamic weather conditions are used as input of a building simulation including floor heating and boreholes. The simulation environment has been designed to reproduce all characteristics (building, weather, boreholes, etc.) of an in-situ GSHP that was monitored during the 2011/2012 heating season in the north of France. This way the tests were carried out under realistic and reproducible conditions, which is practically impossible with sequential in-situ tests. Another advantage of the semi-virtual test-bench is that the real time of the test can be accelerated to significantly reduce the duration of the test (3.5 days instead of 7 days).

The developed ANN predictive controller for single-speed GSHP has been detailed including the training process, the determination of optimal input data, algorithm and structure.

The ANN controller has been compared to the COL conventional controller during two sequential tests of one week on the bench. The ANN controller allows an energy gain of 40%, mainly due to the fact that the pump on the floor heating side has to be constantly running with COL. This also results in a better global SPF with ANN.

VII. ACKNOWLEDGEMENTS

The authors would like to thank the SoDa Service, managed by Transvalor S.A., for providing the solar radiation data used in this study.

REFERENCES

[1] T. Salque, P. Riederer, and D. Marchio, "Development of a Neural Network-based Building Model and Application to Geothermal Heat Pumps Predictive Control", SIMUL 2012, The Fourth International Conference on Advances in System Simulation, November 18-23, Lisbon, Portugal, pp. 24-9, 2012.

[2] B.M. Åkesson, and H.T. Toivonen, "A neural network model predictive controller", Journal of Process Control, vol.16, pp. 937-46, 2006.

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

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

[5] H. Karlsson, and C.-E. Hagentoft, "Application of model based predictive control for water-based floor heating in low energy residential buildings", Building and Environment, vol.46, pp. 556-69, 2011.

[6] J. Široký, F. Oldewurtel, J. Cigler, and S. Prívara, "Experimental analysis of model predictive control for an energy efficient building heating system", Applied Energy, vol.88, pp. 3079-87, 2011.

[7] A.A. Argiriou, I. Bellas-Velidis, M. Kummert, and P. André, "A neural network controller for hydronic heating systems of solar buildings", Neural Networks, vol.17, pp. 427-40, 2004.

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

[9] Riederer P., Partenay V., and Raguideau O., "Dynamic test method for the determination of the global seasonal performance factor of heat pumps used for heating, cooling and domestic hot water preparation.", Eleventh International IBPSA Conference, Glasgow, Scotland, July 27-30, 2009.

[10] "Solar Irradiation Database SODA ", www.soda-is.com,

[11] El Khoury Z., Riederer P., Couillaud N., Simon J., and R. M., "A multizone building model for Matlab/Simulink environment", Ninth International IBPSA Conference, Montreal, Canada, 2005.

[12] T. Salque, D. Marchio, and P. Riederer, "Semi-virtual test bench for comparison of GSHP controllers: tuning of simulated part with measured data ", 11th REHVA World Congress CLIMA 2013, June 16-19 (in press), 2013.

[13] V. Partenay, P. Riederer, T. Salque, and E. Wurtz, "The influence of the borehole short-time response on ground source heat pump system efficiency", Energy and Buildings, vol.43, pp. 1280-7, 2011.

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

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

[16] MATLAB, "Version 7.0.1, (R14SP1). Mathworks Inc., Ma., USA.", Available from: http://www.mathworks.com,

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

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

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

2013, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

www.iariajournals.org

International Journal On Advances in Intelligent Systems

ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, ENERGY, COLLA, IMMM, INTELLI, SMART, DATA ANALYTICS

issn: 1942-2679

International Journal On Advances in Internet Technology

ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING, MOBILITY, WEB

issn: 1942-2652

International Journal On Advances in Life Sciences

eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO, SOTICS, GLOBAL HEALTH

issn: 1942-2660

International Journal On Advances in Networks and Services

ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION, VEHICULAR, INNOV

issn: 1942-2644

International Journal On Advances in Security

ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS

issn: 1942-2636

International Journal On Advances in Software

ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, IMMM, MOBILITY, VEHICULAR, DATA ANALYTICS

issn: 1942-2628

International Journal On Advances in Systems and Measurements

ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL, INFOCOMP

issn: 1942-261x

International Journal On Advances in Telecommunications

AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA, COCORA, PESARO, INNOV

issn: 1942-2601