INFOCOMP 2012

The Second International Conference on Advanced Communications and Computation

ISBN: 978-1-61208-226-4

October 21-26, 2012

Venice, Italy

INFOCOMP 2012 Editors

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster / Leibniz

Universität Hannover / North-German Supercomputing Alliance, Germany

Petre Dini, Concordia University - Montreal, Canada / China Space Agency Center - Beijing, China

Wolfgang Hommel, Leibniz Supercomputing Centre - Munich, Germany

Malgorzata Pankowska, University of Economics, Katowice, Poland

Lutz Schubert, High Performance Computing Centre (HLRS) of the University of Stuttgart, Germany

INFOCOMP 2012

Foreword

The Second International Conference on Advanced Communications and Computation [INFOCOMP 2012], held between October 21-26, 2012 - Venice, Italy, continued a series of events dedicated to advanced communications and computing aspects, covering academic and industrial achievements and visions.

The diversity of data semantics, context gathering and processing, led to complex mechanisms for applications requiring special communication and computation support in terms of volume of data, processing speed, context variety, etc. New computation paradigms and communications technologies are now driven by the needs for fast processing and requirements from data-intensive applications and domain-oriented applications (medicine, geoinformatics, climatology, remote learning, education, large scale digital libraries, social networks, etc.). Mobility, ubiquity, multicast, multi-access networks, data centers, cloud computing are now forming the spectrum of approaches in response to the diversity of user demands and applications. In parallel, measurements control and management (self-management) of such environments evolved to deal with new complex situations.

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

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the INFOCOMP 2012 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that INFOCOMP 2012 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in advanced communications and computation.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the charm of Venice, Italy.

INFOCOMP Chairs:

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster / Leibniz Universität Hannover / North-German Supercomputing Alliance, Germany
Hans-Joachim Bungartz, Technische Universität München (TUM) - Garching, Germany
Petre Dini, Concordia University - Montreal, Canada / China Space Agency Center - Beijing, China
Erik Elmroth, Department of Computing Science and HPC2N, Umeå University / High Performance Computing Center North (HPC2N), Sweden

Sik Lee, Supercomputing Center / Korea Institute of Science and Technology Information (KISTI), Korea
Subhash Saini, NASA, USA
Alexander Knapp, Universität Augsburg, Germany
Malgorzata Pankowska, University of Economics, Katowice, Poland
Kei Davis, Los Alamos National Laboratory, USA
Daniel S. Katz, Computation Institute, University of Chicago & Argonne National Laboratory, USA
Edgar A. Leon, Lawrence Livermore National Laboratory, USA
Ivor Spence, School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Northern Ireland, Research for High Performance and Distributed Computing / Queen's University Belfast, UK
Alfred Geiger, T-Systems Solutions for Research GmbH, Germany
Hans-Günther Müller, SGI Silicon Graphics - Göttingen, Germany
Björn Hagemeier, Juelich Supercomputing Centre, Forschungszentrum Juelich GmbH, Germany
Walter Lioen, SARA Computing and Networking Services - Amsterdam, The Netherlands
Lutz Schubert, High Performance Computing Centre (HLRS) of the University of Stuttgart, Germany
Noelia Correia, University of the Algarve, Portugal
Wolfgang Hommel, Leibniz Supercomputing Centre - Munich, Germany
George Michelogiannakis, Stanford University, USA
Bernhard Bandow, Max Planck Institute for Solar System Research (MPS), Katlenburg-Lindau, Germany
Diglio A. Simoni, RTI International - Research Triangle Park, USA
Dominic Eschweiler, Frankfurt Institute for Advanced Studies (FIAS), Germany
Huong Ha, University of Newcastle, Australia (Singapore campus)
Philipp Kremer, German Aerospace Center (DLR), Institute of Robotics and Mechatronics, Oberpfaffenhofen, Germany
Ulrich Norbisrath, BIOMETRY.com / University of Tartu, Estonia

INFOCOMP 2012

Committee

INFOCOMP General Chair

Claus-Peter Rückemann, Westfälische Wilhelms-Universität Münster / Leibniz Universität Hannover / North-German Supercomputing Alliance, Germany

INFOCOMP Advisory Chairs

Hans-Joachim Bungartz, Technische Universität München (TUM) - Garching, Germany
Petre Dini, Concordia University - Montreal, Canada / China Space Agency Center - Beijing, China
Erik Elmroth, Department of Computing Science and HPC2N, Umeå University / High Performance Computing Center North (HPC2N), Sweden
Sik Lee, Supercomputing Center / Korea Institute of Science and Technology Information (KISTI), Korea
Subhash Saini, NASA, USA

INFOCOMP Academia Chairs

Alexander Knapp, Universität Augsburg, Germany
Malgorzata Pankowska, University of Economics, Katowice, Poland

INFOCOMP Research Institute Liaison Chairs

Kei Davis, Los Alamos National Laboratory, USA
Daniel S. Katz, Computation Institute, University of Chicago & Argonne National Laboratory, USA
Edgar A. Leon, Lawrence Livermore National Laboratory, USA
Ivor Spence, School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Northern Ireland, Research for High
Performance and Distributed Computing / Queen's University Belfast, UK

INFOCOMP Industry Chairs

Alfred Geiger, T-Systems Solutions for Research GmbH, Germany
Hans-Günther Müller, SGI Silicon Graphics - Göttingen, Germany

INFOCOMP Special Area Chairs on Large Scale and Fast Computation

Björn Hagemeier, Juelich Supercomputing Centre, Forschungszentrum Juelich GmbH, Germany
Walter Lioen, SARA Computing and Networking Services - Amsterdam, The Netherlands
Lutz Schubert, High Performance Computing Centre (HLRS) of the University of Stuttgart, Germany

INFOCOMP Special Area Chairs on Networks and Communications

Noelia Correia, University of the Algarve, Portugal
Wolfgang Hommel, Leibniz Supercomputing Centre - Munich, Germany
George Michelogiannakis, Stanford University, USA

INFOCOMP Special Area Chairs on Advanced Applications

Bernhard Bandow, Max Planck Institute for Solar System Research (MPS), Katlenburg-Lindau, Germany
Diglio A. Simoni, RTI International - Research Triangle Park, USA

INFOCOMP Special Area Chairs on Evaluation Context

Dominic Eschweiler, Frankfurt Institute for Advanced Studies (FIAS), Germany
Huong Ha, University of Newcastle, Australia (Singapore campus)
Philipp Kremer, German Aerospace Center (DLR), Institute of Robotics and Mechatronics, Oberpfaffenhofen, Germany

INFOCOMP Special Area Chairs on Biometry

Ulrich Norbisrath, BIOMETRY.com / University of Tartu, Estonia

INFOCOMP 2012 Technical Program Committee

Ajith Abraham, Machine Intelligence Research Labs (MIR Labs), USA
Douglas Archibald, University of Ottawa, Canada
Tulin Atmaca, IT/Telecom&Management SudParis, France
Bernhard Bandow, Max Planck Institute for Solar System Research, Katlenburg-Lindau, Germany
Belgacem Ben Youssef, King Saud University Riyadh, KSA / Simon Fraser University Vancouver, British Columbia, Canada
Rim Bouhouch, National Engineering School of Tunis, Tunisia
Elzbieta Bukowska, Poznan University of Economics, Poland
Hans-Joachim Bungartz, Technische Universität München (TUM) - Garching, Germany
Elena Camossi, European Commission, Joint Research Centre, Institute for the Protection and Security of the Citizen (IPSC) - Ispra, Italy
Laura Carrington, University of California, San Diego/ San Diego Supercomputer Center, USA
Noelia Correia, University of Algarve, Portugal
Kei Davis, Los Alamos National Laboratory / Computer, Computational, and Statistical Sciences Division, USA
Sergio De Agostino, Sapienza University-Rome, Italy
Vieri del Bianco, Università dell'Insubria, Italia
Amine Dhraief, University of Kairouan, Tunisia
Beniamino Di Martino, Dipartimento di Ingegneria dell'Informazione, Seconda Università di Napoli, Italy
Erik Elmroth, Department of Computing Science and HPC2N, Umeå University, Sweden
Christian Engwer, Institute for Computational und Applied Mathematics, Westfälische Wilhelms-Universität Münster, Germany
Dominic Eschweiler, Frankfurt Institute for Advanced Studies (FIAS), Germany
Jürgen Falkner, Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO, Stuttgart, Germany
Lars Frank, Department of IT Management, Copenhagen Business School, Denmark

Alfred Geiger, T-Systems Solutions for Research GmbH, Germany
Birgit Frida Stefanie Gersbeck-Schierholz, Leibniz Universität Hannover/Certification Authority University of Hannover (UH-CA), Germany
Franca Giannini, Consiglio Nazionale delle Ricerche - Genova, Italy
Fabio Gomes de Andrade, Federal Institute of Science, Education and Technology of Paraíba, Brazil
Conceição Granja, Siemens Healthcare Sector, Universidade do Porto - Faculdade de Engenharia, Portugal
Richard Gunstone, Bournemouth University, UK
Huong Ha, University of Newcastle, Australia (Singapore campus)
Björn Hagemeier, Juelich Supercomputing Centre, Forschungszentrum Juelich GmbH, Germany
Wolfgang Hommel, Leibniz Supercomputing Centre - Munich, Germany
Jiman Hong, Soongsil University - Seoul, Korea
Friedrich Hülsmann, Gottfried Wilhelm Leibniz Bibliothek - Hannover, Germany
Udo Inden, Cologne University of Applied Sciences, Research Centre for Applications of Intelligent Systems (CAIS), Germany
Daniel S. Katz, Computation Institute, University of Chicago & Argonne National Laboratory, USA
Abdelmajid Khelil, TU-Darmstadt, Germany
Alexander Kipp, Universität Stuttgart, Germany
Christos Kloukinas, City University London, UK
Alexander Knapp, University of Augsburg, Germany
Manfred Krafczyk, Institute for Computational Modeling in Civil Engineering (iRMB), Braunschweig, Germany
Philipp Kremer, German Aerospace Center (DLR) / Institute of Robotics and Mechatronics - Oberpfaffenhofen, Germany
Herbert Kuchen, Westfälische Wilhelms-Universität Münster, Institut für Wirtschaftsinformatik, Praktische Informatik in der Wirtschaft, Münster, Germany
Sik Lee, Supercomputing Center / Korea Institute of Science and Technology Information (KISTI), Korea
Brian G. Lees, School of Physical, Environmental & Mathematical Sciences, University of New South Wales at the Australian Defence Force Academy, Canberra, Australia
Edgar A. Leon, Lawrence Livermore National Laboratory, USA
Walter Lioen, SARA Computing and Networking Services - Amsterdam, The Netherlands
Georgios Lioudakis, National Technical University of Athens (NTUA), Greece
Dirk Malzahn, OrgaTech, Lunen, Germany
Suresh Marru, Indiana University, USA
Igor Melatti, Sapienza Università di Roma, Rome, Italy
George Michelogiannakis, Stanford University, USA
Jelena Mirkovic, Center for Shared Decision Making and Collaborative Care Research, Oslo University Hospital, Norway
Hans-Günther Müller, SGI Silicon Graphics - Göttingen, Germany
Ulrich Norbisrath, BIOMETRY.com / University of Tartu, Estonia
Aida Omerovic, SINTEF ICT and University of Oslo, Norway
Stephan Onggo, Lancaster University, UK
Malgorzata Pankowska, University of Economics - Katowice, Poland
Giuseppe Patané CNR-IMATI, Italy
Rasmus Ulslev Pedersen, Dept. of IT Management, Embedded Software Laboratory, Copenhagen Business School, Denmark
Paulo Martins Pedro, Chaminade University, USA
Ana-Catalina Plesa, German Aerospace Center, Institute of Planetary Research, Planetary Physics, Berlin,

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

On Efficiency of Solutions of Stochastic Optimal Control Problem with Discrete Time 1
Igor I. Gasanov and Iouldouz S. Raguimov

FlashTKV: A High-Throughput Transactional Key-Value Store on Flash Solid State Drives 5
Robin Jun Yang and Qiong Luo

Control Software Visualization 15
Federico Mari, Igor Melatti, Ivano Salvo, and Enrico Tronci

Parallel Interference Cancellation in DS-OCDMA System Using Novel Multilevel Periodic Codes 21
Besma Hammami, Habib Fathallah, and Houria Rezig

Using Embedded FPGA for Cache Locking in Real-Time Systems 26
Antonio Marti Campoy, Francisco Rodriguez-Ballester, and Rafael Ors Carot

Discovery & Refinement of Scientific Information via a Recommender System 31
Robert Patton, Thomas Potok, and Brian Worley

Enabling Dynamical Use of Integrated Systems and Scientific Supercomputing Resources for Archaeological Information Systems 36
Claus-Peter Ruckemann

Nonlinear Transformation's Impact Factor of Cryptography at Confusion and Cluster Process 42
Lan Luo, Zehui Qu, Yalan Ye, Tao Lv, and Qionghai Dai

Decoupling Modeling From Complexity Based on a Freedom-to-Act Architecture 47
Udo Inden, Sergej Naimark, and Claus-Peter Ruckemann

A Neural Network Ultrasonic Sensor Simulator for Evolutionary Robotics 54
Christiaan J. Pretorius, Mathys C. du Plessis, and Charmain B. Cilliers

Biometric Security Systems for Mobile Devices Based on Fingerprint Recognition Algorithm 62
Michal Szczepanik and Ireneusz Jozwiak

3D Measures Computed in Monocular Camera System for Fall Detection 68
Konstantinos Makantasis, Anastasios Doulamis, and Nikolaos Matsatsinis

FlexRay Static Section Scheduling Using Full Model 74
Rim Bouhouch, Houda Jaouani, Wafa Najjar, and Salem Hasnaoui

Analyzing the Trade-offs Between Minimizing Makespan and Minimizing Energy Consumption in a Heterogeneous Resource Allocation Problem 81
Ryan Friese, Tyler Brinks, Curt Oliver, Howard Jay Siegel, and Anthony A. Maciejewski

Design and Implementation of Context-aware Hadoop InputFormat for Large-scale Scientific Dataset 90
Jae-Hyuck Kwak, Jun Weon Yoon, and Soonwook Hwang

Boundaries of Supercomputing 94
Lutz Schubert and Stefan Wesner

A Framework for Migrating Traditional Web Applications Into Multi-Tenant SaaS 100
Eyad Saleh, Nuhad Shaabani, and Christoph Meinel

Bandwidth-Efficient Parallel Visualization for Mobile Devices 106
Andreas Helfrich-Schkarbanenko, Vincent Heuveline, Roman Reiner, and Sebastian Ritterbusch

High-Performance Network Accommodation and Intra-Slice Switching Using a Type of Virtualization Node 113
Yasusi Kanada, Kei Shiraishi, and Akihiro Nakao

A First Look at AMI Traffic Patterns and Traffic Surge for Future Large Scale Smart Grid Deployments 120
Yaling Nie and Yuanchen Ma

A Platform for the Integrated Management of IT Infrastructure Metrics 125
Christian Straube, Wolfgang Hommel, and Dieter Kranzlmuller

Design, Development and Implementation of a Hybrid Network with Smart Sensors and Power Line Communication for Monitoring of Underground Electricity Substation 130
Paulo Sausen, Airam Sausen, Mauricio de Campos, Camila Gehrke, Rene Emmel Jr., and Fabiano Salvadori

Computing Individual Mobility Profiles From Mobile Phone Usage Traces 136
Miguel Angel Rodriguez-Crespo, Ana Armenta-Lopez-de-Vicuna, Alberto Martin-Dominguez, Rocio Martinez-Lopez, and Ruben Lara-Hernandez

User Involvement and Social Networking for Information System Development 142
Malgorzata Pankowska

A New Approach To Solve Aircraft Recovery Problem 148
Congcong Wu and Meilong Le

A Semi-Supervised Approach for Industrial Workflow Recognition 155
Eftychios Protopapadakis, Anastasios Doulamis, Konstantinos Makantasis, and Athanasios Voulodimos

Analysis of Network Heterogeneity by Using Entropy of the Remaining Degree Distribution 161

Lu Chen, Shin'ichi Arakawa, and Masayuki Murata

Understanding Virtualized Infrastructure in Grid Job Monitoring 167
Zdenek Sustr and Jiri Sitera

A Layered Multi-Tree IP Multicast Protocol With Network Coding 171
Yining Li, Jinguo Quan, Xuelei Tan, and Hui Li

Performance Evaluation of a Planar Layout of Data Vortex Optical Interconnection Network 176
Qimin Yang

An Efficient Scheduling in Consideration of the Signaling Overhead for Relay Networks 182
Deokhui Lee and Jaewoo Soo

On Efficiency of Solutions of Stochastic Optimal Control Problem with Discrete Time

Igor I. Gasanov
Department of Computational and Information Systems

Computing Centre of Russian Academy of Sciences
40 Vavilov Street, 119991 Moscow GSP-1, Russia
E-mail: gasanov@ccas.ru

Iouldouz S. Raguimov
Department of Mathematics and Statistics

York University
4700 Keele Street, Toronto, Ontario, Canada, M3J 1P3
E-mail: raguimov@mathstat.yorku.ca

***Abstract*—For stochastic optimal control problem with discrete time, the efficiency of solutions corresponding to the parameters of a stochastic process determined by the method of optimization on time series is analyzed in comparison to the solutions related to the parameters obtained using a common statistical method of estimation. Parametric optimization problems for continuous and discrete stochastic optimization problems are introduced and the corresponding problems of optimization on time series are formulated. When a sample size is increasing, the asymptotic properties of solutions to the considered problems are investigated. Theorems on the convergence of the optimal objective value of discrete problem of parametric optimization on time series to the optimal objective value of the related discrete stochastic optimization problem have been formulated and proved.**

***Keywords*-*Markov decision process; stochastic optimization; parametric optimization; optimization on time series.*

I. Introduction

The challenges of dealing with uncertainty is a common problem in the management of economic and engineering systems. When uncertainty is modeled probabilistically with random variables, it is usually required to be described as a multidimensional stochastic process for which neither a structure nor parameters are known. Particular challenges related to the estimation of the characteristics of random variables of a process as well as to determining of their interrelationships appear to be very important for this type of problems. At the same time, as a rule, operations research analyst is experiencing a data insufficiency in determining the structure and/or calibrating parameters of a stochastic process. Even in the case, when the model of a stochastic process has been formulated, we obtain an optimization problem that is usually too complicated to solve analytically.

As examples, a decision-making problem with uncertainty related to the natural factors as well as problems of functioning and interaction of financial and economic institutions including a financial portfolio management problem can be referred. To the problems of this type also belongs the equipment replacement problem, which arises when a company has to determine how long a machine should be utilized before being traded in for a new one.

In situations when it is neither possible, nor affordable to obtain an optimal solution analytically, so-called method of optimization on time series [3] is often used to determine the best approximate solution to the problem. Relying on information about realizations of uncontrollable uncertain parameters, it is determined a control for a considered object that would be optimal once were used in the past. Here, it is assumed implicitly that since the uncertainty has a regular character, then a control, which would have been optimal during some sufficiently prolonged period of time in the past, will also be optimal in the future.

The abovementioned idea appears to be rational, especially since the necessity of making decisions in such systems arises frequently, and the authors do not know an efficient alternative approach to solving this type of decision-making problems. On the other hand, this technique raises certain questions and doubts. Particularly, since optimal control is determined and estimated on the same set of realizations of a stochastic parameter, while constructing an optimal control on time series, to what extend are we exploiting systematic properties of the stochastic process, and to what – just are making adjustments by utilizing only some insignificant for the future properties of the stochastic process?

The analysis of this problem seems to be interesting and represents an actual challenge. In Section II of the paper, a parametric optimal control problem with discrete time is introduced. In Section III, the corresponding problem of parametric optimization on time series is constructed and theorem on convergence of its optimal objective value to the optimal objective value of the parametric optimal control problem with discrete time is formulated. In Section IV, optimization of parameters of stochastic process on time series is analyzed and the corresponding control problem with modified stochastic process is introduced. Theorem on convergence of its optimal objective value to the optimal objective value of the corresponding optimal control problem with discrete time is formulated.

II. Parametric Optimal Control Problem with Discrete Time

Consider one of the possible formalizations of a stochastic optimal control problem, namely, the mathematical model of a discrete-time Markov decision process [7]. Suppose that at any time t, $t = 1, 2, \cdots, \infty$, the state of a system is given by the characteristic vector $A_t \in \hat{A} \subset \mathbb{R}^n$. Once the control u_t has been chosen at the stage (time period) t and the value $\tilde{\xi_t}$ of the stochastic parameter ξ is realized, the system moves on from the state A_t to the state

$$A_{t+1} = \varphi(A_t, \tilde{\xi}_t, u_t),$$

where the parameter $\xi \in \Xi \subset \mathbb{R}^m$ is a stationary Markov process with the transition probability function $\Phi(\xi_t \mid \xi_{t-1})$. So, every ordered pair $S_t = (A_t, \tilde{\xi_t})$ of arguments of a function φ determines a state of the system at stage $t+1$. It is assumed that the initial probability distribution, i.e., the probability distribution $F^1(S_1)$ on the set of initial states of the system is known, and at each state S the control $u \in U(S) \subset \hat{U} \subset \mathbb{R}^k$. Here $\hat{A}$ and $\hat{U}$ are bounded sets.

Suppose that every stage t of the process is associated with a certain payoff function (expected reward) $h_t = h_t(S_t, u_t)$ and assume a decision-maker is interested in maximization of the average reward earned per period, i.e., is solving the following maximization problem

$$Q = \lim_{n\to\infty} \frac{1}{n} E(\Sigma_{i=1}^{n} h_t(S_t, u_t)) \implies \max_{u} \qquad (1)$$

Generally speaking, the decision maker wants to maximize function (1) with respect to $u_t = u(S_t)$, where u is a mapping $\hat{A} \times \Xi \to \hat{U}$ (it is assumed that at stage t to the moment of choosing u_t the realization $\tilde{\xi_t}$ of ξ is known).

It is clear that for the existence of the expected value in (1), the functions involved in the model should satisfy certain conditions. Analysis of these conditions is out of the scope of this paper. Related existence problems have been solved in [1], [2].

In [5], discrete models of the stochastic optimization problems are studied and the corresponding discrete problems of optimization on time series are introduced. The convergence of optimal solutions of the discrete problems of optimization on time series to an optimal solution of the discrete stochastic optimization problem has been proved. Properties of optimal solutions of discrete problems of optimization on time series are analyzed and estimates for the optimal objective values are obtained.

It worth noting, that the considered formulation covers a wide range of stochastic control problems. Particularly, it includes the case when it is assumed that at different stage of a process, realizations of a stochastic parameter are independent. A decision-making problem for static models with infinite horizon has been solved in [6].

According to Gasanov and Raguimov [5], the method of optimization on time series is usually applied to parametric optimization problem where a certain parametric class of control functions is considered and a problem is formulated as a problem of choosing optimal values for parameters of a function from the considered class.

Let us consider the problem of maximization of (1) on the set of control functions $\hat{U_\alpha}$ with $\alpha \in \mathbb{A}$, where $\hat{U_\alpha}$ is a class of the functions $u(S;\alpha)$, such that there exists a one-to-one correspondence between $\hat{U_\alpha}$ and $\mathbb{A}$. Therefore, the original problem is reduced to a problem of finding a value of α, such that

$$Q = \lim_{n\to\infty} \frac{1}{n} E(\Sigma_{t=1}^{n} h_t(S_t, u(S_t;\alpha))) \implies \max_{\alpha} \qquad (2)$$

Denote the formulated problem as Problem 1 and compare it with its discrete analogue – Problem 1D, of maximization of function (2) on $\hat{U^D_\alpha}$, the parametric class of discrete functions. Here, $\hat{U^D_\alpha}$ is the set of discrete analogues of $u(S;\alpha) \in \hat{U_\alpha}$. It is supposed that the state space, the set of values of the stochastic parameter and the decision set are finite sets, and consequently, the state vectors of the system, the stochastic parameter and controls take the values on a finite grid, i.e. $A_t \in \{A^i\}_{i=1}^{I} = \hat{A^D} \subset \hat{A}$, $\xi \in \{\xi^j\}_{j=1}^{J} = \Xi^D \subset \Xi$ and $u \in \{u^s\}_{s=1}^{S} \subset \hat{U^D_\alpha}$. Functions φ and Φ are modified correspondingly.

Consider the case when there exists a solution to Problem 1 and assume that these two problems are such that when an appropriate (small) mesh for the grid is chosen, solutions to Problem 1D closely approximate solutions to Problem 1. Therefore, the determining of a solution of Problem 1D is assumed to be the same as the finding of an approximate solution to Problem 1. Certainly, this assumption can be investigated in order ot obtain important analytical results. However, the authors believe that the assumption is valid for a wide range of practical problems and consequently, plausible from the point of view of applications.

III. Problem of Parametric Optimization on Time Series

Suppose, the sequence of realizations $\{\tilde{\xi_1}, \tilde{\xi_2}, \cdots, \tilde{\xi_T}\} \subset \Xi^D$ of stochastic parameter ξ and the initial state $\tilde{A_1} \in \hat{A^D}$ of the system are given. Consider the following discrete optimization problem.

Problem 1R. Maximize the function

$$\tilde{Q^T} = \frac{1}{T} \Sigma_{t=1}^{T} h_t(A_t, \tilde{\xi_t}, u(A_t, \tilde{\xi}; \alpha)) \qquad (3)$$

on the set of control functions $u_t = u(A_t, \tilde{\xi_t})$, subject to the constraint $A_1 = \tilde{A_1}$.

The control function u is said to be *everywhere optimal*, if it is optimal for every initial distribution $F^1(S_1)$.

The following theorem is proved.

Theorem 1. If there exists an everywhere optimal control for Problem 1D, then the optimal objective value of Problem 1R converges almost everywhere to the optimal objective value of Problem 1D, provided that the size of the sample $\{\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T\} \subset \Xi^D$ increases unboundedly, i.e., T approaches infinity.

Therefore, under the abovementioned assumptions an optimal solution of Problem 1R represents a well-grounded estimate for an optimal solution of Problem 1D. At the same time, it is clear that a substantial limitation of decision set will affect the optimal value. It is difficult to measure this effect, unless Problem 1 and the original maximization problem (1) both are solved.

Also, as it was mentioned above, an optimal control for Problem 1R is determined and estimated on the same sample of the realizations, which may result in a displacement (particularly, in an overstatement) of the estimates. The possible range of this displacement for considered series of realizations of ξ is not considered in this paper.

IV. Optimization of Parameters of Stochastic Process on Time Series

Often, when either a given data does not allow to construct a reliable model of a stochastic process or Problem 1 is overly complicated to be solved in the original form, the considered stochastic process is replaced with a simple one, which according to the opinion of an operations research analyst reflects essential characteristics of the original process. The parameters $\beta \in \mathbb{R}^n$ of this auxiliary process are calibrated on the given series of realizations $\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T$ and the problem with the accordingly modified stochastic process is considered. Denote the obtained problem as Problem 1M. Let $Q(u)$ be the objective value of Problem 1 corresponding to the control u and consider the parametric class of optimization problems of the type 1M, where as parameters the calibrated coefficients of the modified stochastic process are considered. Suppose u^β is a solution to Problem 1M, which corresponds to fixed values of the coefficients β and $Q(u^\beta)$ is the corresponding objective value. Consider the problem of maximization of $Q(u^\beta)$ on the set of calibrated parameters β and denote it as Problem 1A.

Let us also estimate the parameters of the stochastic process using one of the commonly used statistical methods, namely, using Monte-Carlo method, and consider the problem corresponding to the determined parameters. Denote the obtained problem as Problem 1S. As before, discrete analogues of the problems 1M, 1A and 1S can be formulated. Denote these problems as Problem 1MD, Problem 1AD and Problem 1SD, respectively.

The efficiency of a control function obtained by solving Problem 1MD can be estimated on the sample $\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T$ by calculating the value of objective function (3). Now, formulate Problem 1MR as a problem of finding the values of the parameters of the stochastic model that maximize the value of objective function (3).

Let $u^{1SD}(\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T)$ and $u^{1MR}(\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T)$ be optimal control functions for Problem 1SD and Problem 1MR, correspondingly.

Using Theorem 1, the following theorem has been proved.

Theorem 2. If there exists an everywhere optimal control function u^{1AD} for Problem 1AD, then the optimal objective value $Q(u^{1MR}(\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T))$ of Problem 1MR converges almost everywhere to the optimal objective value $Q(u^{1AD})$, as T approaches infinity.
Moreover, for every sequence of realizations $\{\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T\}$,

$$Q(u^{1SD}(\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T)) \leq Q(u^{1AD}).$$

Since the stochastic model of Problem 1S is, at most, one of the elements of an heuristic procedure, it would be unfounded to assume that the values $Q(u^{1SD}(\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T))$ will converge to the value of $Q(u^{1AD})$, as T approaches infinity. Therefore, the following conclusion can be deduced. Provided that the size of a sample $\tilde{\xi}_1, \tilde{\xi}_2, \cdots, \tilde{\xi}_T$ increases, the solving Problem 1MR is, generally speaking, a more efficient method to solve Problem 1 than the solving Problem 1M with the parameters of stochastic model estimated on the same sample using any of the commonly used statistical methods. In other words, the solving Problem 1MR as a method of solving Problem 1 is asymptotically preferred to the solving Problem 1M with the parameters estimated by any other statistical method.

We understand that the asymptotic preference of one method to another does not provide formal grounds to consider the first method as more efficient in solving practical problems where data samples are always limited and mostly not large enough. For applied problems with a stochastic process of a non-established structure, the theoretical evaluation of the method based on solving of Problem 1MR appears to be difficult. Therefore, to estimate the presented method from the practical point of view it is necessary to carry out series of computational experiments.

In [4], mathematical models of a controlled system containing a model of a stochastic process in the form of Markov process are implemented where the Markov process is modeling a financial market. Using the Markov process, data imitating series of observations is generated. Then, from the point of view of an operations research analyst who does not know the structure of the stochastic process but knows only the series of observations, various problems of optimization on time series have been investigated. Computational experiments are implemented for different size of data imitating the series of realizations and different behavior of the operations research analyst. The results, that is, the obtained optimal controls and their estimates on the given series of realizations can be compared with their “real” efficiency, i.e., with the efficiency on the original Markov process.

Certainly, such experiments cannot be considered as a formal proof of efficiency of the presented method. Nevertheless, from the point of view of their further utilization, the results of the experiments seem to be essential.

V. Conclusions and Future Work

For the stochastic optimal control problem with discrete time, the efficiency of solutions has been analyzed. Solutions related to the values of the parameters of a stochastic process determined by the method of optimization on time series has been compared with the solutions related to the parameters obtained using a common statistical method of estimation. Parametric optimization problems for the corresponding continuous and discrete stochastic optimization problems have been introduced and the related problems of optimization on time series have been formulated. When a sample size increases, the asymptotic properties of solutions to the considered problems have been analyzed. Theorems on the convergence of the optimal objective value of a discrete problem of parametric optimization on time series to the optimal objective value of the discrete stochastic optimization problem have been formulated and proved. The authors are intending to extend the obtained results to stochastic decision-making problems for hidden Markov processes.

References

[1] D. P. Bertsekas and S. E. Shreve, *Stochastic Optimal Control. The Discrete Time Case*, New York: Academic Press, 1978.

[2] E. B. Dynkin and A. A. Yushkevich, *Controlled Markov Processes*, New York: Springer-Verlag, 1979.

[3] G. A. Agasandyan, I. I. Gasanov, I. S. Menshikov, A. N. Chaban, and Yu. M. Chebanyuk, Calculation methods for problems of control of reservoir modes, in *Cybernetics and Computational Techniques,* vol. 3, Moscow: Nauka, 1987, pp. 57-101 (Russian).

[4] I. I. Gasanov and I. S. Raguimov, “On algorithm of financial portfolio control by the method of optimization on time series”, In the proceedings of the *2004 International Conference on Computational Intelligence for Modeling Control and Automation-CIMCA”2004* /ISBN 1740881885, pp. 511-519.

[5] I. I. Gasanov and I. S. Raguimov, “On solution of stochastic control problem by the method of optimization on time series”, In the proceedings of the *2003 Hawaii International Conference on Statistics and Related Fields* /ISSN 1539-7211, pp. 1-7.

[6] I. S. Raguimov, “On decision making in operations with stochastic factors”, In the proceedings of the *2001 International Conference on Computational Intelligence for Modeling Control and Automation-CIMCA”2001* /ISBN 0858898470, pp. 580-587.

[7] W. L. Winston, *Operations Research: Applications and Algorithms*, Belmont, California: Duxbury Press, 1994.

ISBN: 978-1-61208-226-4

FlashTKV: A High-Throughput Transactional Key-Value Store on Flash Solid State Drives

Robin Jun Yang
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Hong Kong, China
yjrobin@cse.ust.hk

Qiong Luo
Department of Computer Science and Engineering
Hong Kong University of Science and Technology
Hong Kong, China
luo@cse.ust.hk

***Abstract*—We propose FlashTKV, a high-performance transactional key-value store optimized for flash-based solid state drives. Transactional key-value stores process large numbers of concurrent reads and writes of key-value pairs, and maintain transactional consistency. As such systems are I/O dominant, flash SSDs are a promising storage alternative to improve the system performance. Catering the asymmetry in the read and write performance of flash SSDs, FlashTKV uses a purely sequential storage format where all data and transactional information are log records. Furthermore, this sequential storage format supports multi-version concurrency control (MVCC) efficiently. We evaluate FlashTKV on enterprise SSDs as well as on magnetic disks. While on magnetic disks FlashTKV performs similarly to systems with MVCC on page-based storage or locking on sequential storage under TPC-C workloads, it improves the transaction throughput by 70% over the competitors on flashSSDs.**

Keywords**-KV-store; Flash SSD; Log-structured; MVCC .***

I. Introduction

Flash Solid State Drives (SSDs) are emerging as a competitive storage alternative for laptops, desktops, as well as servers, due to their outstanding I/O performance, shock resistance, and energy efficiency. Table I shows the performance comparison between a representative enterprise flash SSD and a high-end magnetic disk. While both disks achieve an almost identical throughput on sequential writes, the sequential read throughput of the flash SSD is 1.5 times of that on the hard disk. A striking difference between the two disks across access patterns, is that, the read and write performance is symmetric on the magnetic disk but is not on the flash SSD. In particular, on the SSD the sequential read throughput is 1.5 times of the sequential write, and the random read throughput is over 10 times of the random write. Finally, the performance gap between random and sequential patterns is reduced from a factor of 200 on the hard disk to around 2 for reads and 15 for writes on the flash SSD. While these numbers confirm the superb performance of flash SSDs, they also suggest that performance optimization strategies for the flash may be different from those for the hard disk due to the read-write asymmetry.

Table I: Performance Comparison between An Intel X25-E Flash SSD and A SAS 15kRPM Magnetic Disk

Device	Flash SSD	Magnetic disk
Seq. Read Throughput	248MB/s	164MB/s
Seq. Write Throughput	167MB/s	166MB/s
Ran. 4KB Read IOPS (Calculated Throughput)	33,569 (127.2MB/s)	192 (0.75MB/s)
Ran. 4KB Write IOPS (Calculated Throughput)	2,940 (11.5MB/s)	192 (0.75MB/s)
Read Latency	75μs	5200μs
Write Latency	85μs	5200μs

Recently there have been studies on optimizing the I/O performance of a database management system component, such as query processing [1], buffer management [2], [3], indexing [4], [5], [6], [7] and storage management [8] for flash SSDs. There has also been work on using flash SSDs for key-value stores (KV-stores), such as FlashStore [9] and SkimpyStash [10]. In comparison, we focus on transactional key-value stores, which is an important type of workload in practice yet is challenging for flash SSDs due to the large number of random writes.

A transactional KV-store, such as the Oracle BerkeleyDB [11], supports read and write operations on key-value pairs, and guarantees transactional consistency of these read and write operations. As a result, there are large numbers of random I/O for key-value pair reads and writes as well as a large amount of transaction log writes. Considering the characteristics of flashSSDs, we propose FlashTKV, a transactional KV-store for flash SSDs. FlashTKV has the following three distinguishing features:

- It has a purely sequential storage format where all the data are stored as log records (log as data).
- All transactional information are also written as log records into the sequential storage.
- The sequential storage supports the multiversion concurrency control protocol (MVCC) for transactional consistency.

Acknowledgement: This work was supported in part by grant HUAW28-15L05211/12PN from Huawei Technologies.

The main technical challenges in FlashTKV are how to support (1) reads and (2) MVCC efficiently on the sequential storage. Specifically, log-structured approaches [12] optimize writes by converting random data writes into sequential log writes, but slow down reads as up-to-date data pages must be constructed by applying change logs to the original data pages. Also, MVCC has two main drawbacks: (1) the overhead of writing multiple versions of each data item; and (2) wasted processing due to transaction rollbacks. The first drawback is less costly on flash SSDs than on hard disks as writes to flash memory will be to new pages anyway and random reads are fast on flash. The second drawback remains on flash SSDs; nevertheless it is outweighed by the fast reads on flash SSDs, as we will see in the experiments. Furthermore, existing MVCC algorithms and implementations all assume a page-based data storage format and a separate, write-ahead logging (WAL) transaction log. It is unclear how a sequential storage format without a page-based data storage or a separate transaction log can support MVCC correctly and efficiently.

To support reads efficiently, our sequential storage with a uniform set of logs replaces separated sets of data pages and change logs. Consequently, there is no merging operation between data pages and change logs. Instead, we only need to retrieve a suitable log record for a given key on a read request. To speed up exact-match as well as range searches, we further maintain a B$^+$-tree to index the KV-pairs and use an in-memory node buffer pool to keep recently accessed B$^+$-tree nodes. To support MVCC on our sequential storage format, we keep necessary transactional information in log records and retrieve a suitable log record for each transactional read based on timestamp information.

We have implemented FlashTKV and evaluated it in comparison with the Oracle Berkeley DB (BDB), a leading industrial-strength transactional key-value store on an enterprise-grade flash SSD. Our results show that (1) the estimated read I/O time in FlashKTV was almost identical to that in BDB and the estimated write time in FlashKTV was only 30% of that in BDB; (2) the measured performance of FlashKTV under different degrees of read-write contention was up to 40% faster than that of BDB; (3) under TPC-C workloads, FlashKTV improves the throughput by up to 70% over BDB. This paper is organised as follows: Section II discusses the background and related work of our paper, Section III describes the detailed design and implementation of FlashTKV, Section IV compares the I/O operations in the traditional page storage and the sequential storage used in FlashTKV, Section V shows the experimental setup and results and Section VI concludes the paper.

II. Background and Related Work

In this section, we first discuss the read-write asymmetry of flash SSDs. Then we review related work on optimization techniques that addressed this issue, especially log-structured approaches. Finally we compare FlashKTV with other key-value stores, especially the Oracle Berkeley DB Java Edition (BDBJE), which also adopts a sequential storage format.

Flash SSDs use the NAND flash memory as the storage media which does not support in-place update, but instead requires an erase operation before a write. The erase operation can only be performed at the granularity of an erase block (typically 64 flash pages). The FTL (Flash Translation Layer) inside an SSD alleviates this problem by directing writes to clean pages; however, it also causes garbage collection to run more frequently. As a result, random writes continue to be the worst-performing access pattern on flash SSDs.

To address the random write problem on flash SSDs, a few new file systems [13], [14] have been proposed. They are similar to the log-structured file system [12], which maintains a mapping between logical and physical addresses of pages and transforms write requests to sequential append operations to the storage device. This log-structured approach avoids random writes to the storage device, but slows down read operations due to the use of the mapping table to locate the current page. Furthermore, garbage collection in these file systems needs to run frequently and degrades the performance severely, especially on flash SSDs of a large capacity.

There has been a flurry of work on optimizing DBMS components such as query processing [1], buffer management [2], [3], indexing [4], [5], [6], [7] and storage management [8] for flash SSDs. As transactional workloads such as OLTP (Online Transactional Processing) generate a large number of random writes on traditional database systems, they are the most challenging to optimize on flash SSDs. There has also been work on using flash SSDs for lightweight database systems such as the Key-Value Stores (KV-stores), e.g., FlashStore [9] and SkimpyStash [10]. These systems focus on minimizing the metadata size per key-value pair (KV-pair) in the RAM so that they can provide fast access and insertion to large datasets without introducing a significant maintenance cost for the metadata (index) of the KV-pairs. Nevertheless, these systems do not support user-defined database transactions and thus are unsuitable for OLTP applications.

Traditional two-phase locking has been the protocol of choice for concurrency control. With the read-write performance asymmetry of flash SSDs, there has been initial work exploring alternative concurrency control protocols on flash disks. In particular, Lee et al. [15] experimented with storing the MVCC rollback segments in a commercial database server on flash SSDs. Nevertheless, there has not been work on studying a full MVCC transactional system with sequential storage on flash SSDs. Our work is most related to the Berkeley DB Java Edition [16] (BDBJE), a well-known sequential storage engine. As it is Java-based, BDBJE relies on JAVA NIO and has no explicit control on the underlying storage device. Furthermore, only locking, not MVCC, is supported for concurrency control in BDBJE.

A drawback of log-structured approaches, which are often adopted for flash-optimized techniques, is an essential and expensive operation, known as *merge*. The merge operation is necessary because the original data and the logged changes are separate entities, and these two need to be integrated from time to time to bring the data up-to-date. In comparison, our FlashTKV adopts a sequential storage format where all data and their changes are recorded uniformly as logs in time order. As a result, there is no merge operation needed. To speed up the random reads, we maintain an in-memory buffer pool for log nodes and organize these nodes into an in-memory B^+-tree for exact match as well as range searches on the keys.

III. Design and Implementation

In this section, we present the design and implementation of FlashTKV.

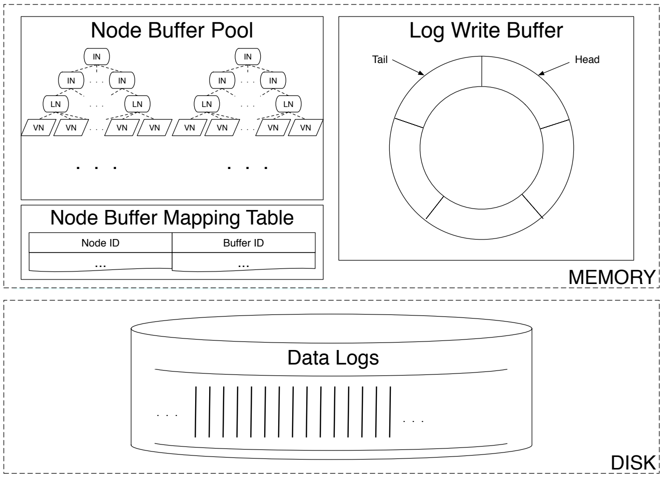


Figure 1: FlashTKV Storage

A. System Overview

Figure 1 illustrates the storage design in FlashTKV. On disk, we store all KV-pairs in *data logs* in the order of time when an insertion/deletion/update happens. For efficiency, we use an in-memory ring buffer as the *log write buffer* to batch up the tail of the data logs and write them to disk when the buffer is full or when transactions commit.

Since data logs are written in time order to the disk whereas KV-pair operations are based on keys, we use an in-memory buffer to cache frequently accessed KV-pairs. Furthermore, to support lookups and range searches efficiently, we maintain a B^+-tree index for each set of KV-pairs. Specifically, we store the keys in LNs (Leaf Nodes) and the values in VNs (Value Nodes), and create INs (Internal Nodes) to form a tree. We separate keys and values in memory because the sizes of values may vary greatly. Since the nodes are variable-sized, multiple nodes may reside on a single buffer page, and large nodes may span over multiple pages. A retrieval on a B^+-tree in this *node buffer pool* will start from the root, find the node ID of the child by the search key, use the node buffer mapping table to find the buffer page that contains the child node, and go down the tree iteratively until it finds the value node in the buffer or on disk or reports the non-existence of such a key in the database.

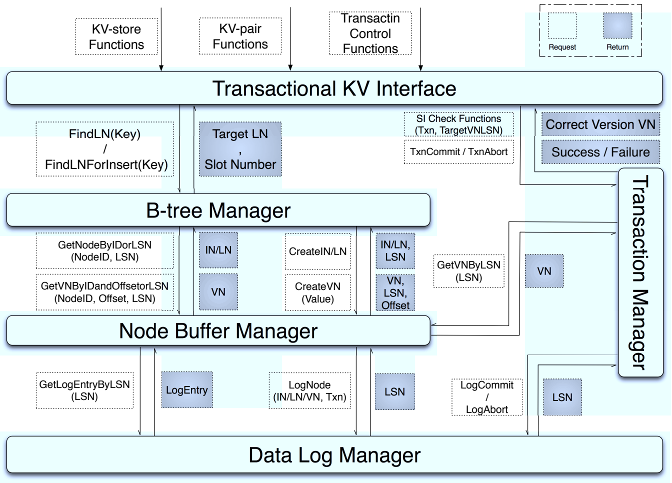


Figure 2: FlashTKV System Architecture

FlashTKV consists of five main components as shown in Figure 2. All the changes of the KV-pairs are stored in *data logs* and appended to the disk through the **Data Log Manager**. To support search on the KV-pairs, the **B^+-tree Manager** builds B^+-trees for all the KV-pairs using their keys. Frequently visited KV-pairs are kept in the RAM in the form of *nodes* in the *node buffer pool* maintained by the **Node Buffer Manager**. The **Transaction Manager** manages transactions of KV-pair operations. It utilizes MVCC to provide SI (Snapshot Isolation) for all transactions. By introducing a few more types of data logs in the **Data Log Manager** for storing all the information of transactions, *data logs* can be used to not only store KV-pairs but also provide transaction support. All the available functions in FlashTKV are provided by **Transactional KV Interface**. We discuss the five components in the following sections.

B. Transactional KV Interface

The *transactional KV interface* provides the interface of KV-store functions (create, open or close the KV-stores), KV-pair functions (transactional retrieval, insertion, update and deletion of KV-pairs), transaction control functions (start, abort and commit a transaction). It calls the B^+-tree manager and the transaction manager to implement all the functions. The *Database* in FlashTKV is a directory in the file system. Each set of KV-pairs of the same schema are stored in a *TupleStore*. A database may contain multiple TupleStores.

C. B^+-tree Manager

The KV-pairs in one TupleStore are stored in one B^+-tree. The B^+-tree in FlashTKV has three types of nodes, **IN**, **LN** and **VN**. As shown in Figure 3, all keys are stored in the LNs, all values are stored in the VNs. One LN contains multiple keys whereas one VN contains only one value. The B^+-tree manager relies on the node buffer manager to maintain the memory space used by all nodes.

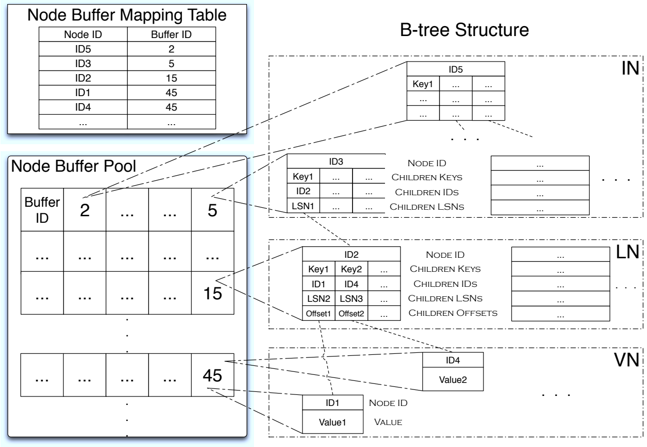


Figure 3: The B^+-tree Structure and Node Buffer Pool Layout for Sequential Storage in FlashTKV

To support the sequential storage and MVCC, the B^+-tree in FlashTKV has a few unique features compared to the standard B^+-tree. The standard B^+-tree uses one identifier, the node ID, which is the same as the page ID in the page-based storage, as the *persistent pointer* to locate a node in memory and on disk. However, such identifier is not enough to locate the node on disk in the sequential storage because writing an updated node to the disk is to append a new log to the disk, which means the physical position of the node on disk is changed. Therefore, the B^+-tree for the sequential storage uses the LSN of the node on disk as the persistent pointer. Moreover, the node size is flexible because all the data of the nodes are stored in data logs. Considering the maintenance cost and the efficiency of the memory access, we set the size of the IN/LN/VN to the size of a node buffer in the node buffer pool. We discuss the details of the memory allocation in Section III-D. Lock coupling [17] is used to provide high concurrency in INs while LNs and VNs can be accessed by multiple transactions. Furthermore, we perform opportunistic split: we split all full nodes on the search path for the insertion. Thus, latches can be obtained strictly from top down so that deadlocks can be avoided.

The biggest drawback of such design is the update efficiency. More specifically, when a VN is inserted or updated, its parent LN also needs to be updated because one of the LN's children LSNs changes and such updates will propagate up all the way to the root. We call this the *update propagation problem*. To overcome this problem, we write the log for the new VN and update the corresponding child LSN in the LN but only mark the status of the slot for the new VN in the LN *dirty* without writing logs for the updated LN immediately so that the update propagation is prevented. The logs for the updated IN/LN are written only when the IN/LN is evicted from memory. This treatment does not lose any change in data because the logs for the VNs already contain the whole KV-pair.

D. Node Buffer Manager

The node buffer manager is responsible for maintaining the memory space used by the nodes in the B^+-tree and returning the memory address of the requested IN/LN or VN. As shown in Figure 3, the *node buffer pool* is a large chunk of memory, with each unit called *a node buffer*. Compared to a buffer manager for the page-based storage, it has some unique features.

The node buffer manager allocates exactly one node buffer for each IN/LN but multiple VNs can be stored in a single node buffer. This different treatment is because (1) the numbers of INs/LNs are much fewer than VNs in the memory because of the tree structure; (2) the size of each VN varies. The maximum size of the VN is limited to the size of a node buffer, and we put multiple VNs into a single node buffer to save memory space.

When the node buffer pool is full, we use an LRU algorithm to choose a node buffer for eviction. Such LRU may choose a node buffer for IN to swap out while some of the node's children are still in the buffer pool and marked dirty. Since we must guarantee its latest version is written on disk when a node is evicted from the node buffer pool, we must write all its dirty children to disk to get their latest LSNs before writing the parent node. This process happens recursively until all the dirty nodes in the subtree rooted at the victim IN are written to disk. As a result, we can free all the node buffers for all the INs/LNs in this subtree. Note that the node buffers for the VNs in this subtree are not evicted because they may contain frequently visited VNs from other LNs. The node buffers for VNs are evicted only when the replacement algorithm chooses them as victims, which indicates all of VNs in this buffer are not recently used.

E. Data Log Manager

The data log manager is responsible for (1) transforming B^+-tree nodes into data logs and writing them onto the disk, and (2) reading data logs from the disk and transforming them to B$^+$-tree nodes. Figure 4 shows all types of data logs in FlashTKV. The data log manager maintains a global log write buffer. The data logs to be written to disk are appended in the buffer and the buffer is flushed to disk when (1) the size of the existing data logs that have not been flushed exceeds the size of a flash erase block or (2) a transaction commits. The size of the buffer is an integral multiple of the flash erase block size. We organize the buffer as a ring buffer which further saves read I/O cost in the retrieval of the last committed version of a KV-pair in SI transactions. In addition, we implement the *group commit* algorithm to further increase the write I/O efficiency.

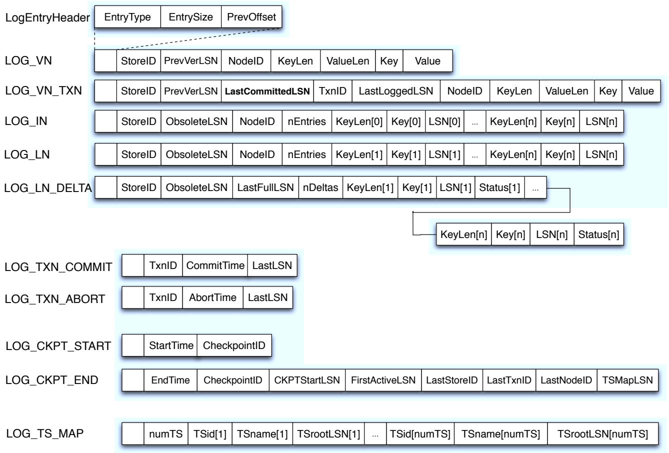

LogEntryHeader | EntryType | EntrySize | PrevOffset
LOG_VN | StoreID | PrevVerLSN | NodeID | KeyLen | ValueLen | Key | Value
LOG_VN_TXN | StoreID | PrevVerLSN | LastCommittedLSN | TxnID | LastLoggedLSN | NodeID | KeyLen | ValueLen | Key | Value
LOG_IN | StoreID | ObsoleteLSN | NodeID | nEntries | KeyLen[0] | Key[0] | LSN[0] | ... | KeyLen[n] | Key[n] | LSN[n]
LOG_LN | StoreID | ObsoleteLSN | NodeID | nEntries | KeyLen[1] | Key[1] | LSN[1] | ... | KeyLen[n] | Key[n] | LSN[n]
LOG_LN_DELTA | StoreID | ObsoleteLSN | LastFullLSN | nDeltas | KeyLen[1] | Key[1] | LSN[1] | Status[1] | ...
KeyLen[n] | Key[n] | LSN[n] | Status[n]
LOG_TXN_COMMIT | TxnID | CommitTime | LastLSN
LOG_TXN_ABORT | TxnID | AbortTime | LastLSN
LOG_CKPT_START | StartTime | CheckpointID
LOG_CKPT_END | EndTime | CheckpointID | CKPTStartLSN | FirstActiveLSN | LastStoreID | LastTxnID | LastNodeID | TSMapLSN
LOG_TS_MAP | numTS | TSid[1] | TSname[1] | TSrootLSN[1] | ... | TSid[numTS] | TSname[numTS] | TSrootLSN[numTS]

Figure 4: The Format of All Types of Data Logs

1) Data Logs for B$^+$-tree Nodes: We have five types of data logs to store three kinds of B$^+$-tree nodes. Both LOG_VN and LOG_VN_TXN are for the VNs. The difference between them is that LOG_VN_TXN is for the VNs inserted or updated by user-defined transactions. The *PrevVerLSN* in LOG_VN and LOG_VN_TXN is the LSN of the data log for the previous version of the VN. LOG_LN, and LOG_IN are for LNs and INs, respectively. The *ObsoleteLSN* is the LSN of the previous version of the corresponding LN or IN. The difference between LOG_IN and LOG_LN is that the IN has *nEntries + 1* children (IN or LN) but the LN has *nEntries* children VNs.

We further optimize the data logs for LNs because we found it is inefficient to log the entire dirty LN every time it is evicted from the buffer pool because only a few slots of the LSN array or the children ID array are dirty. We introduce a LOG_LN_DELTA log which contains only the updated part of the LN since its last LOG_LN log on disk. We use a simple I/O cost estimation to decide which type of log for LNs to use. Table II shows the total I/O time of writing either type of the logs and reading the LN back based on the number of the dirty entries to be written in a delta log. If $W_{delta} + R_{delta} < W_{LN} + R_{LN}$, LOG_LN_DELTA is chosen; otherwise, LOG_LN is chosen.

Table II: Total I/O Time Estimation for Deciding Which Log to Use

	Write I/O Time	Read I/O Time
DELTA	$W_{delta} = \left(\frac{N_{dirty}S_{LN}}{N_{LN}S_{fp}}\right)T_{SW}$	$R_{delta} = 2T_{RR}$
FULL	$W_{LN} = \left(\frac{S_{LN}}{S_{fp}}\right)T_{SW}$	$R_{LN} = T_{RR}$

S_{delta}: the size of the delta log
S_{LN} : the size of the log of the full version LN
S_{fp} : the flash page size of the flash SSD in use
N_{dirty} : the number of the dirty entries since last full version of the LN
N_{LN} : the total number of entries of LN
T_{SW} : I/O time of write a flash page sequentially
T_{RR} : I/O time of read a flash page randomly

When the LN is later reconstructed from a LOG_LN_DELTA, the dirty entries in it are still marked *dirty* in the LN. This marking is necessary because later if we decide to log the LOG_LN_DELTA again, we still need to log the previous dirty slots. An LN can be reconstructed either directly from (1) an LOG_LN or (2) an LOG_LN_DELTA and the LOG_LN. We need at most two random read I/Os to reconstruct an LN. Considering disk space cost, we set the maximum number of consecutive LOG_LN_DELTA logs for each LN as a configurable parameter.

2) Snapshot Isolation (SI) Support: To support SI transactions, we log the KV-pairs updated or created by SI transactions as LOG_VN_TXN. It has a similar format to LOG_VN except it contains some transactional information of the VN. More specifically, *LastLoggedLSN* is the LSN of the previous data log that belongs to the same transaction as the current log. *LastCommittedLSN* in LOG_VN_TXN is the key field to provide MVCC support in the sequential storage: it is the LSN of the last committed version of the VN before the transaction which generates this log starts. We discuss how this field helps implement SI in transaction processing in Section III-F.

F. Transaction Manager

One of the most important features of FlashTKV is its efficient support of Snapshot Isolation (SI).The SI for a KV-store means a transaction T never sees the modifications of KV-pairs done by other transactions that start later than T. Since the LSN used by the entire system is monotonically increasing, we use it as the timestamp to decide the order of transactions. More specifically, when a transaction starts, we use its first LSN as the start timestamp of the transaction.

1) KV-pair Operations in SI Transactions: Because all the KV-pairs in FlashTKV are contained in LOG_VN/LOG_VN_TXN logs, the transactional KV-pair operations only affect the access of LNs and VNs. For

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

the KV-pair retrieval, the correct version of the KV-pair is located by following the *PrevVerLSN* in the logs. For the KV-pair insertion/update/deletion, the VN with new value is logged using LOG_VN_TXN. It contains *LastCommittedLSN*, the LSN of the last committed version of this VN to be seen by this SI transaction, *TxnID*, and *LastLoggedLSN*. These fields are used later to (1) check whether the transaction can commit or not and (2) undo the aborted transaction.

2) Transaction Commit: We adopt the *First-Committer-Wins* rule[18] to decide whether a transaction can be committed or not. In FlashTKV, the rule requires that an SI transaction T can commit only if all the KV-pairs it wrote are not written by any other committed transactions that started later than T. More specifically, when an SI transaction wants to commit, for each LOG_VN_TXN it generates, we check the corresponding VN to see if the *LastCommittedLSN* is the same as that in the data log. If all of them are the same, the SI transaction can commit, otherwise, FlashTKV automatically aborts it. If a transaction commits, we write a LOG_TXN_COMMIT log to the log write buffer and flush it.

3) Transaction Abort: To abort an SI transaction, we must undo all the changes the transaction made. Before the undo, we add a LOG_TXN_ABORT log and flush it to make sure the transaction will be aborted even if a crash happens during the undo. More specifically, for each LOG_VN_TXN it generates, if the current LSN of the corresponding VN is the same as the LSN of the data log, we set its LSN to the *PrevVerLSN* in the data log.

G. Checkpoint and Recovery

The recovery of FlashTKV is quite different from those storage systems that contain data pages: It involves rebuilding the B$^+$-tree with all KV-pairs updated or inserted by committed transactions. Similar to the traditional DBMSs, we do checkpointing to help reduce the recovery time.

The checkpointing in FlashTKV flushes the following data logs to disk: (1) the LOG_CKPT_START log, (2) the LOG_LN log for a dirty LN, and the LOG_IN logs for the INs that are ancestors of a dirty LN, (3) the LOG_TS_MAP log, and (4) the LOG_CKPT_END log.

The recovery in FlashTKV starts by a backward scan of the data logs. The scan stops immediately after find the most recent checkpoint. Then, starting from the end of the checkpoint, we scan the data logs forward to replay all the data logs for INs/LNs to reconstruct them. Finally, we start from the *FirstActiveLSN* in the LOG_CKPT_END log to undo (redo) all the VN logs from uncommitted (committed) transactions.

H. Garbage Collection

In FlashTKV, the data logs for INs/LNs/VNs may become obsolete when the corresponding nodes are updated. To recycle the disk space used by those obsolete data logs, we perform garbage collection (GC) on those log files in which most of the data logs are obsolete (default is 70% in FlashTKV). BDBJE proposed a solution to recycle a data log file in the sequential storage scheme: the system copies the non-obsolete data logs in the file to a new place before erasing the entire file. However, this requires the exclusive locks on those data logs which violates the design principle of FlashTKV that reads are never blocked. In addition, we cannot block all the SI transactions during the GC because FlashTKV is designed for OLTP workloads that usually have response time requirement (such as TPC-C). Therefore, there are two main challenges for doing GC in FlashTKV: (1) how to determine whether a data log is recyclable when there are some active SI transactions and (2) how to recycle a file without exclusive locks.

We propose a novel approach to do GC in FlashTKV. For the first challenge, we observe that if the up-to-date version of the data is already visible to the oldest active transaction, all the previous versions of the data are safe to be recycled. Therefore, we use an array, called *GC-array*, to keep track of all committed updates (old and new LSNs). The old versions of the data are marked obsolete only when the up-to-date versions of the data are visible to the current oldest active transaction. For the second challenge, we treat the copying of unrecyclable data logs as the update of the corresponding INs/LNs/VNs with the some content, and group those updates into a normal SI transaction, called *GC-transaction*. As long as *GC-transaction* commits, the log file can be erased. *GC-transaction* always restarts automatically when it aborts because of other SI transactions. Note that the abortion may not only happen to the *GC-transaction*, but also the user transactions due to the commit of the *GC-transaction* under First-Committer-Win rule. However, our experiments show that the number of transaction abortion caused by *GC-transaction* (<0.04%) is neglectable compared to the normal abortion rate for TPC-C (≈0.5%). Details can be found in Section V-E.

IV. I/O Cost Comparison

We compare the time cost of all the I/O operations in traditional page storage scheme and sequential storage scheme in Table III. We only discuss the I/O operations during the normal execution. In other words, the I/O during the recovery, checkpoint, and garbage collection is not included because these operations are not frequently executed. The comparison is based on the workloads that do not involve any scan and all queries can be processed through indices, e.g. TPC-C.

In a traditional storage scheme, data pages contain all the data and transaction logs are stored separately. Under a transactional read-write workload, the database system using the traditional storage scheme may produce physical I/Os in three ways during the normal execution: (1) Page read due to

Table III: Comparison of I/O Operations in Page Storage and Sequential Storage

Storage Scheme	I/O Operation	I/O Time
Page Storage	Dirty Page Flush	T_{RW}
	Txn Log Flush	T_{SW}
	Page Read	T_{RR}
Sequential Storage	Data Log Flush	T_{SW}
	Data Log Read	T_{RR}

T_{SW} : I/O time of writing a flash page sequentially
T_{RR} : I/O time of reading a flash page randomly
T_{RW} : I/O time of writing a flash page randomly

page buffer miss (random read), (2) dirty page flush (random write) and (3) transaction logs flush (sequential write). In our sequential storage scheme, however, there are no data pages, instead, data is encapsulated in the *data logs*. As a result, there are only two ways to produce physical I/Os: (1) Data logs read due to node buffer miss (random read) and (2) data logs flush (sequential write). Note that the random write I/Os generated by flushing dirty pages in the page storage scheme no longer exist in the sequential storage. This is because all the updates are transformed into data logs which are appended to the log write buffer sequentially.

V. EXPERIMENT

In this section, we first compare the performance of the sequential storage scheme and the traditional page-based storage scheme on synthetic workloads with different read-write ratios. Then we quantify the performance impact of the locking-based concurrency control on the flash SSDs under workloads with different degrees of read-write lock contentions. Finally, we compare the overall performance of our FlashTKV with two well-known KV-stores, Berkeley DB (which uses the page-based storage) and Berkeley DB Java Edition (which uses the sequential storage).

A. Experimental Settings

1) Hardware: All of our experiments run on a Dell R410 server with a 2.7GHz Intel Xeon E5520 CPU and 8GB RAM. In the server, we have a 150GB 15000RPM magnetic disk connected with the SAS interface, and a 64GB Intel X25-E flash SSD [19] connected through the SATA interface. The hardware specification and detailed I/O performance of the storage devices are listed in Table I.

2) Software: The operating system is CentOS 5.2 Final (kernel version 2.6.18-92.el5). We use Ext3 of Linux as the default file system and the file system cache is disabled to stress the I/O performance. GLib 2.30 is used in the FlashTKV library. For comparison, we use BerkeleyDB (BDB) 5.0.32 and BerkeleyDB Java Edition (BDBJE) 4.0.71 as the representatives of the page storage and sequential storage accordingly. We modify an existing TPC-C [20] implementation [21] to work for FlashTKV, BDB and BDBJE.

Table IV: Workloads for I/O Time Comparison of The Page-based Storage and The Sequential Storage

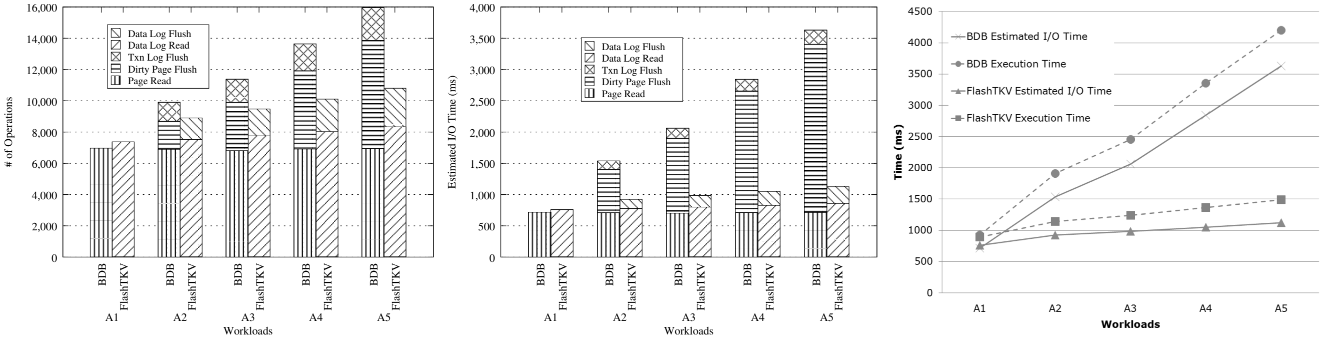
Workload	No. of Retrievals	No. of Updates
A1	10,000	0
A2	7,500	2,500
A3	5,000	5,000
A4	2,500	7,500
A5	0	10,000

Table V: Workloads for Comparing MVCC and Locking On Flash SSDs

Workload	Key Range For Retrieval
B1	1 - 8000
B2	1 - 4000
B3	1 - 1000

3) Workloads: The workloads used for comparing the page storage and sequential storage are listed in Table IV. The synthetic workloads are generated in a database with 100 million key-value pairs (around 10GB in size). The benchmark is a single-threaded program that generates a sequence of KV-pair retrieval/update (read/write) operations with random keys in BDB or FlashTKV with every 10 operations forming a transaction. We modify the source code of BDB and FlashTKV so that it can count the total number of each kind of operations listed in Table III and we observe almost no performance degradation caused by the modification compared with the original system. Both BDB and FlashTKV have a 2GB memory buffer (page buffer/node buffer) and a 30 minutes warm-up time before counting the operations. The counting lasts for 10,000 KV-pair operations for all the workloads.

The workloads used for comparing MVCC and locking on flash SSDs are listed in Table V. The workload is generated in a database with 10,000 key-value pairs (around 1MB in size). The workload consists of the writer threads and the reader threads. Each writer thread is responsible for repeatedly updating a subset of KV-pairs. The key range of the KV-pairs each writer thread updates is equal to the number of KV-pairs divided by the number of the writer threads so that we can guarantee there are no write-write conflicts. In our case, every writer thread updates 100 KV-pairs. Each reader thread continuously picks one KV-pair with a random key to retrieve and each retrieval operation forms a read-only transaction. In our experiments, we change the key range of the retrieval operation to get different degrees of read-write conflicts. Note that we use a 100MB buffer which is much larger than the data size (1MB) so that there are no other write I/Os than the transaction logs flush. The big buffer also guarantees that there is always enough space in the buffer to hold the old versions for MVCC. Furthermore, we bring all the KV-pairs into the buffer before measuring the total time of the workloads so that we can eliminate the impact of the read I/O caused by



(a) Total Number of Different I/O Operations in Berkeley DB and FlashTKV

(b) Estimated I/O Time of Berkeley DB and FlashTKV on The Flash SSD

(c) Comparison of Estimated I/O Time and Execution Time on The Flash SSD

Figure 5: Page-based Storage v.s Sequential Storage

the buffer miss in the locking-based system and focus on the lock waiting time and the I/O time of the transaction log flushes.

Table VI: TPC-C Workload Settings

Workload	Scale	Database Size	Buffer Size
C1	100W	12GB	512MB
C2	100W	12GB	2GB
C3	100W	12GB	4GB
D1	100W	12GB	4GB
D2	200W	24GB	4GB
D3	300W	37GB	4GB

The workloads used to measure the overall performance are described in Table VI. TPC-C workloads have a large number of concurrent random read/write operations. There are three types (C1, C2, C3) of TPC-C workloads with a fixed database size and different buffer sizes. There are another three types (D1, D2, D3) of TPC-C workloads with a fixed buffer size and different database sizes. Note that by default, FlashTKV uses MVCC for transaction processing, therefore we show the performance of the locking-based FlashTKV only to quantify the impact of the programming language of the storage systems when comparing with BDBJE.

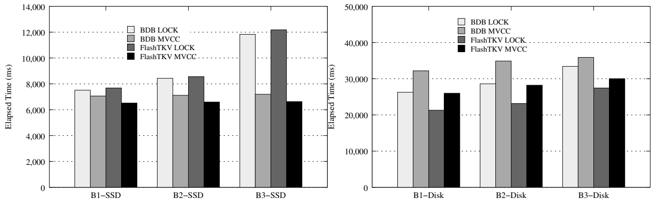
B. *Comparison of The Page-based Storage and The Sequential Storage*

To quantify the benefit of using the sequential storage instead of the page-based storage on flash SSDs, we count the total number of each operation listed in Table III under synthetic workloads with different read-write ratios. Figure 5a shows the total number of each kind of operations listed in Table III under the synthetic workloads described in Table IV. Under the synthetic workloads, both BDB and FlashTKV have a similar buffer miss rate since they use the same buffer replacement policy, LRU. Because the read-only workload does not generate any LN delta logs, the numbers of the page read I/O and the node read I/O number are almost the same. This indicates even FlashTKV uses an entirely different storage scheme from BDB, the node-based buffer strategy can achieve a similar performance to the traditional page-based buffer strategy. As the workload becomes more write-intensive, the dirty page flush in the BDB increases but the transaction log flush remains the same because we only flush the transaction logs when the transaction commits and the number of transactions for each workload is the same. In FlashTKV, the number of data log reads for buffer miss also increases when the workload becomes write-intensive. This increase is because LN delta logs may incur one to two data log reads for each LN retrieval. However, this increase is moderate because LNs are likely to be hold in the memory. Different from BDB where the number of transaction log flushes remains almost the same, the number of data log flushes in FlashTKV increases slightly because LNs may also be flushed to the data log.

Based on the I/O performance of the flash SSD we use, we can derive the three hardware-related parameters in Table III by taking the read/write latency into account, in the worst case, $T_{RR} = 103\mu s$, $T_{RW} = 388\mu s$, $T_{SW} = 108\mu s$. We then calculate the total I/O time of the workloads shown in Figure 5a according to Table III. As shown in Figure 5b, by avoiding the random writes, the most expensive operations in the flash SSD, FlashTKV can achieve a higher performance than BDB under read-write workloads on the flash SSD. More specifically, the more write-intensive the workload is, the more performance speedup FlashTKV gains over BDB, e.g., about 3x speedup for the write-only workload. We compare the estimated I/O time and the execution time of BDB and FlashTKV on all the workloads in Figure 5c. The difference between the estimated I/O time and the execution time for each storage engine can be accounted to the CPU and memory access time. The difference is small, which indicates that in both FlashTKV and BDB, the total execution time is dominated by I/O time. In both engines, our estimated I/O time follows the trend of the total execution time.

C. MVCC Versus Locking

To quantify the negative impact of the read-write lock contention on the flash SSD, we implement a small benchmark to compare the performance drop when increasing the degree of the read-write conflict.



(a) Comparison of MVCC and Locking on SSD with Degree of The Read-write Conflict Varied

(b) Comparison of MVCC and Locking on Disk with Degree of The Read-write Conflict Varied

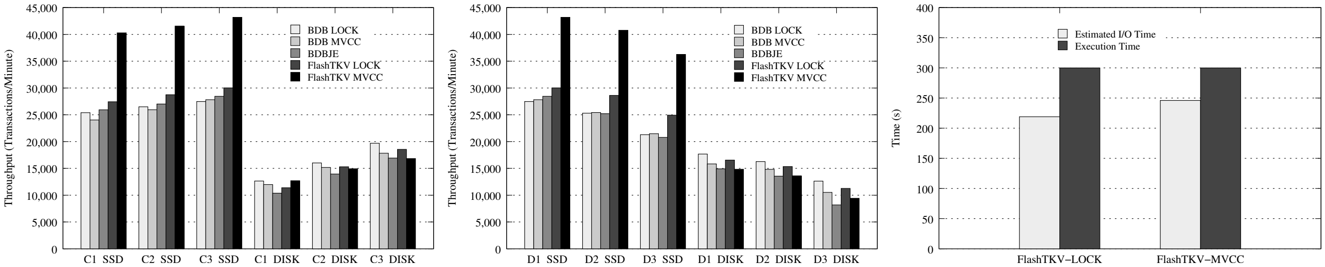
Figure 7: MVCC v.s Locking

Figures 7a and 7b show the total elapse time of BDB and FlashTKV running the workloads in Table V on the SSD and the magnetic disk. Note that in the order of workload B1, B2, and B3, the degree of the read-write conflict increases. Under all of these three workloads, with the same storage engine, locking always outperforms MVCC on the magnetic disk. This is because the random read performance is much worse than the sequential write performance on the disk. As a result, the I/O time spent on the random reads for multiple versions of data in MVCC is more than the time of waiting for the log flushes (sequential writes). In contrast, on the flash SSD, the MVCC version always wins. This is because on the SSD, the random reads for multiple versions of data cost much less than waiting for the sequential writes. This result suggests that on the flash SSD, MVCC is better than the locking-based concurrency control under workloads with read-write conflicts.

D. Overall Performance

We compare the overall performance of our FlashTKV with a sequential storage engine (BDBJE) and a page-based storage engine (BDB with MVCC) by measuring the throughput under TPC-C workloads with different database sizes and buffer sizes. As shown in Figure 6a, on the flash SSD, MVCC-based FlashTKV always outperforms other storage engines. However on the disk, MVCC-based FlashTKV has a similar performance to others, and is even worse when the buffer gets larger. This is because the extra read I/Os used to retrieve old versions of KV-pairs cannot be saved by increasing the buffer size.

Figure 6b compares the performance among the storage engines with different numbers of warehouses. BDBJE and locking-based FlashTKV are very similar in both the storage scheme and concurrency control approach, but there is about 20% performance difference in D3 workload. This performance difference is mainly due to the platform (Java versus C) and implementation. Furthermore, as shown both in Figure 6a and 6b, the flash SSD substantially helps increase the overall throughput of the storage engines. Due to the poor performance of the random read on the magnetic disk, the performance of the sequential storage engines, including both BDBJE and FlashTKV, is even worse than the page-based storage engine BDB on the magnetic disk. On the SSD, however, FlashTKV outperforms the other two storage engines, achieving a speedup of 1.68x over BDB, and 1.54x over BDBJE. We count the numbers of the I/O operations in FlashTKV under Workload D1 for five minutes and compare the estimated I/O time with the execution time in Figure 6c. As one can see, the estimated I/O time is 73% and 82% of the execution time in locking-based FlashTKV and MVCC-based one, respectively. This indicates that the TPC-C workload in FlashTKV is still I/O dominant. Therefore, the performance improvement is mainly because the I/O time is reduced in the sequential storage.



(a) Throughput Comparison among Different Storage Engines with Varied Buffer Size

(b) Throughput Comparison among Different Storage Engines with Varied Database Size

(c) Comparison of Estimated I/O Time and Execution Time of FlashTKV on SSD under Workload D1

Figure 6: Overall Performance Comparison

E. Garbage Collection

Since GC in FlashTKV may introduce more transaction abortion, we quantify the impact of GC on the overall performance by counting the total number of transaction abortion because of the *GC-transaction*. We run the TPC-C workload C3 with and without GC for 2 hours.

Table VII: Comparison of The Transaction Abortion under Workload C3 with And without GC

	Without GC	With GC
Total # of Transactions	5,181,842	5,166,385
New-Order Abortion	24,974	25,753
GC-transaction Abortion	/	504
Other transactions Abortion	/	608
Overall Abortion Rate	**0.48%**	**0.52%**

As shown in Table VII, without GC, there is a 0.48% abortion rate for New-Order transactions in TPC-C and no abortion of other transactions. With GC, the number of New-Order transaction abortion slightly increased. In addition, the *GC-transaction* and some other transactions (such as Payment or Delivery) also have abortion. However, the total abortion rate of all the transactions only increased 0.04% which is neglectable compared to that without GC.

VI. Conclusion and Future Work

In conclusion, we have presented the design and implementation of FlashTKV, a transactional KV-store optimized for flash-based solid state drives. The two main features of FlashTKV are (i) a sequential storage format that stores logs as data and also incorporates transactional information; (ii) Snapshot Isolation transaction support through MVCC on the sequential storage. We have evaluated FlashTKV in comparison with both BerkeleyDB C version (BDB), which has a page-based storage layout, and Java version (BDBJE), which has a sequential storage layout with locking based concurrency control. Our results show that, under TPC-C workloads, while FlashTKV is slightly worse than BDB with locking on magnetic disks, it outperforms its competitors by 70% in throughput on flash SSDs. Based on these results, we believe that our sequential storage format with MVCC is a promising approach for transactional key-value stores on flash disks.

References

[1] D. Tsirogiannis, S. Harizopoulos, M. A. Shah, J. L. Wiener, and G. Graefe, “Query Processing Techniques for Solid State Drives,” in *SIGMOD Conference*, 2009, pp. 59–72.

[2] Y. Ou, T. Härder, and P. Jin, “CFDC: A Flash-aware Replacement Policy for Database Buffer Management,” in *DaMoN*, 2009, pp. 15–20.

[3] Y. Lv, B. Cui, B. He, and X. Chen, “Operation-aware Buffer Management in Flash-based Systems,” in *SIGMOD Conference*, 2011, pp. 13–24.

[4] D. Agrawal, D. Ganesan, R. Sitaraman, Y. Diao, and S. Singh, “Lazy-Adaptive Tree: An Optimized Index Structure for Flash Devices,” *PVLDB*, vol. 2, no. 1, pp. 361–372, 2009.

[5] Y. Li, B. He, Q. Luo, and K. Yi, “Tree Indexing on Flash Disks,” in *ICDE*, 2009.

[6] C.-H. Wu, L.-P. Chang, and T.-W. Kuo, “An Efficient R-tree Implementation Over Flash-memory Storage Systems,” in *GIS*, 2003, pp. 17–24.

[7] C.-H. Wu, T.-W. Kuo, and L. P. Chang, “An Efficient B-tree Layer Implementation for Flash-memory Storage Systems,” in *RTCSA*, 2003, pp. 409–430.

[8] S.-W. Lee and B. Moon, “Design of Flash-based DBMS: An In-page Logging Approach,” in *SIGMOD Conference*, 2007, pp. 55–66.

[9] B. Debnath, S. Sengupta, and J. Li, “FlashStore: High Throughput Persistent Key-Value Store,” *PVLDB*, vol. 3, no. 2, pp. 1414–1425, 2010.

[10] B. Debnath, S. Sengupta, and J. Li, “SkimpyStash: RAM Space Skimpy Key-value Store on Flash-based Storage,” in *SIGMOD Conference*, 2011, pp. 25–36.

[11] Oracle, “Berkeley DB Products,” 2010.

[12] M. Rosenblum and J. K. Ousterhout, “The Design and Implementation of a Log-Structured File System,” *ACM Trans. Comput. Syst.*, vol. 10, no. 1, pp. 26–52, 1992.

[13] C. Manning, “YAFFS: The NAND-specific Flash File System,” 2002.

[14] D. Woodhouse, “JFFS: The Journalling Flash File System,” in *Ottawa Linux Symposium*, 2001.

[15] S.-W. Lee, B. Moon, C. Park, J.-M. Kim, and S.-W. Kim, “A Case for Flash Memory SSD in Enterprise Database Applications,” in *SIGMOD Conference*, 2008, pp. 1075–1086.

[16] Oracle, “White Paper: Berkeley DB Java Edition Architecture,” 2006.

[17] R. Bayer and M. Schkolnick, “Concurrency of Operations on B-Trees,” *Acta Inf.*, vol. 9, pp. 1–21, 1977.

[18] H. Berenson, P. A. Bernstein, J. Gray, J. Melton, E. J. O’Neil, and P. E. O’Neil, “A Critique of ANSI SQL Isolation Levels,” in *SIGMOD Conference*, 1995, pp. 1–10.

[19] Intel Corp., *Intel X25-E SATA Solid State Drive Datasheet*, 2008.

[20] TPC, *TPC Benchmark C Standard Specification Revision 5.9*. Transaction Processing Performance Council, 2007.

[21] SYNAR, Simon Fraser University, “Tpc-c benchmark on bdb,” <http://synar.cs.sfu.ca/systems/code/tpcc-bdb.tar>, 08.19.2012.

Control Software Visualization

Federico Mari, Igor Melatti, Ivano Salvo and Enrico Tronci
Department of Computer Science
Sapienza University of Rome
Via Salaria 113, 00198 Rome, Italy
Email: {*mari,melatti,salvo,tronci*}*@di.uniroma1.it*

***Abstract*—Many software as well digital hardware automatic synthesis methods define the set of implementations meeting the given system specifications with a boolean relation K (*controller*). Such relation, given a system state s and an action u, returns 1 iff taking action u in state s leads in the system goal or at least one step closer to it. In order to determine at hand if K is a "good" controller, e.g., if it covers a wide enough portion of the system state space, or to provide an high level view of the actions enabled by K, it is useful to picture K in a 2D or 3D diagram. In this paper, starting from a canonical representation for K, we propose an algorithm to automatically generate such a picture, relying on available graphing tools.**

Keywords*-*Control Software Visualization*; *Embedded Systems*; *Model Checking

I. Introduction

Many *Embedded Systems* are indeed *Software Based Control Systems* (SBCSs). An SBCS consists of two main subsystems: the *controller* and the *plant*. Typically, the plant is a physical system consisting, for example, of mechanical or electrical devices whereas the controller consists of *control software* running on a microcontroller. In an endless loop, the controller reads *sensor* outputs from the plant and sends commands to plant *actuators* in order to guarantee that the *closed loop system* (that is, the system consisting of both plant and controller) meets given *safety* and *liveness* specifications (*System Level Formal Specifications*).

Software generation from models and formal specifications forms the core of *Model Based Design* of embedded software [1]. This approach is particularly interesting for SBCSs since in such a case system level (formal) specifications are much easier to define than the control software behavior itself.

The typical control loop skeleton for an SBCS is the following. Measure x of the system state from plant *sensors* goes through an *analog-to-digital* (AD) conversion, yielding a *quantized* value $\hat{x}$. A function *ctrlRegion* checks if $\hat{x}$ belongs to the region in which the control software works correctly. If this is not the case, a *Fault Detection, Isolation and Recovery* (FDIR) procedure is triggered; otherwise a function *ctrlLaw* computes a command $\hat{u}$ to be sent to plant *actuators* after a *digital-to-analog* (DA) conversion. Basically, the control software design problem for SBCSs consists in designing software implementing functions *ctrlLaw* and *ctrlRegion*.

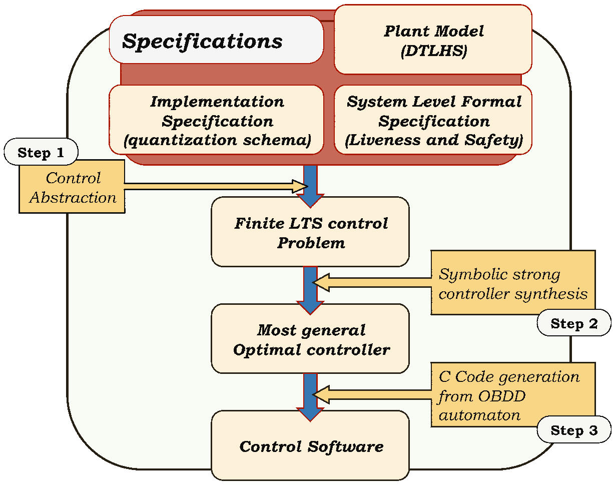

Specifications
Plant Model (DTLHS)
Implementation Specification (quantization schema)
System Level Formal Specification (Liveness and Safety)
Step 1
Control Abstraction
Finite LTS control Problem
Symbolic strong controller synthesis
Step 2
Most general Optimal controller
C Code generation from OBDD automaton
Step 3
Control Software

Figure 1. Control Software Synthesis Flow.

Automatic methods and tools aiming at synthesizing both functions *ctrlLaw* and *ctrlRegion* above have been developed in the last years, e.g., in [2][3][4][5][6][7]. In this paper, we will refer to the method described in [7], but the approach we describe may be applied to the other ones as well. Figure 1 shows the model based control software synthesis flow in [7]. A specification consists of a plant model, given as a Discrete Time Linear Hybrid System (DTLHS), System Level Formal Specifications that describe functional requirements of the closed loop system, and Implementation Specifications that describe non functional requirements of the control software, such as the number of bits used in the quantization process, the required worst case execution time, etc. Given such an input, in step 1 a suitable finite discrete abstraction (*control abstraction* [7]) $\hat{\mathcal{H}}$ of the DTLHS plant model $\mathcal{H}$ is computed; $\hat{\mathcal{H}}$ depends on the quantization schema and it is the plant as it can be seen from the control software after AD conversion. Then (step 2), given an abstraction $\hat{G}$ of the goal states G, it is computed a controller K that starting from any initial abstract state, drives $\hat{\mathcal{H}}$ to $\hat{G}$ regardless of possible nondeterminism. Control abstraction properties ensure that K is indeed a (quantized representation of a) controller for the original plant $\mathcal{H}$. Finally (step 3), the finite automaton K is translated into control software (C code).

In the following, we represent the control software with a boolean relation K (*controller*) taking as input (the n-bits encoding of) a *state* x of the plant and (the r-bits encoding of) a proposed *action* to be performed u, and returns *true* (i.e., 1) iff the system specifications are met when performing action u in state x. In this approach, K is synthesized so that a given (*initial*) plant states region I (which is given as part of the system level formal specifications) is guaranteed to be covered by K. That is, for all states $x \in I$, there must exist at least an action u s.t. $K(x, u)$ holds. Typically, I is set to be small in order to increase the likelihood that a K fulfilling the above given property exists. However, the set of states covered by K, i.e., $\text{dom}(K) = \{x \mid \exists u : K(x, u)\}$ may result to be much bigger than I. Therefore, once a K is built, it is useful to have a tool to graphically depict $\text{dom}(K)$, in order to be able to visualize how big the region $\text{dom}(K)$ is, as well as to have a glimpse of which actions are turned on by K on given plant states regions.

A. Our Main Contributions

In this paper we present an algorithm that, from an OBDD (*Ordered Binary Decision Diagram* [14]) representation of a controller K for a DTLHS modeling an SBCS, effectively generates a 2D picture (namely, an input file for Gnuplot [8]) depicting K. Such picture consists on a cartesian plane where each point corresponds to a state of the starting DTLHS, and shows as painted with the same color all regions of states for which the same *actions set* is defined on K. The color for a state (x, y) depends on which actions set is enabled by K in the DTLHS state (x, y), i.e., it is uniquely determined by $c(x, y) = \{u \mid K((x, y), u)\}$. As a special case, if $c(x, y) = \varnothing$ for some (x, y), i.e., (x, y) is not controlled by K, then the color is white. A separated picture showing the relation between a color and the corresponding actions set is also automatically generated. In this way, the state region for which any color is shown depicts the coverage of K, whilest the regions colors give a glimpse of which actions are turned on by K.

In our setting, since we seek K for which a software implementation is possible, a finite number of bits is used to encode both the states and the actions of the starting DTLHS. Suppose now that $|u| = r$, i.e., if r bits are needed in order to encode an action of the given DTLHS. Then, there may be at most 2^{2^r} different actions sets, i.e., $|\{c(x, y) \mid (x, y)$ is a state$\}| = 2^{2^r}$. That is, with $r = 5$ we need 4×10^9 colors, which is more than a typical RGB with 8 bits per color may achieve. Thus, our method may work only up to $r = 4$. Note however that this is not a limitation, since typical DTLHSs do not need more than 3 bits per action. Moreover, for most systems $|\{c(x, y) \mid (x, y)$ is a state$\}| << 2^{2^r}$, thus we may generate the picture even if $r \geq 5$.

We present experimental results showing effectiveness of the proposed algorithm. As an example, in about 1 hour we are able to generate the pairs of pictures described above for a multi-input buck DC/DC converter [9] with $r = 4$ action bit variables.

B. Paper outline

This paper is organized as follows. Section III provides the background needed to understand the results of this paper. Section IV describes our method to generate a picture visualizing a controller. Section V provides experimental results. Finally, Section VI summarizes and concludes the paper.

II. Related Work

Many papers (e.g., see [7][11][12][13]) tackling the problem of synthesizing control software (which looks to quantized states) or control laws (which looks at real states) of hybrid systems show pictures of the type we generate in this paper (with $r = 1$, i.e., only one bit for the actions). However, to the best of our knowledge there are no papers directly focusing on the method to generate such pictures, thus no automatic approach to controllers visualization is described.

Therefore, to the best of our knowledge this is the first time that an algorithm generating a picture of the coverage of a controller for a DTLHS is presented.

III. Basic Definitions

To make this paper self-contained, in this section we briefly summarize previous work on automatic generation of control software for *Discrete Time Linear Hybrid System* (DTLHS) from System Level Formal Specifications focusing on basic definitions and mathematical tools that will be useful in the sequel.

Figure 1 shows the control software synthesis flow that we consider here [7]. We model the controlled system (i.e., the plant) as a DTLHS (Section III-D), that is a discrete time hybrid system whose dynamics is modeled as a *linear predicate* (Section III-A) over a set of continuous as well as discrete variables. The semantics of a DTLHS is given in terms of a *Labeled Transition Systems* (LTS, Section III-C).

Given a plant $\mathcal{H}$ modeled as a DTLHS, a set of *goal states* G (*liveness specifications*) and an *initial region* I, both represented as linear predicates, we are interested in finding a *restriction* K *of the behaviour* of $\mathcal{H}$ such that in the *closed loop system* all paths starting in a state in I lead to G after a finite number of steps. Finding K is the DTLHS *control problem* (Section III-D) that is in turn defined as a suitable LTS control problem (Section III-C).

Finally, we are interested in controllers that take their decisions by looking at *quantized states*, i.e., the values that the control software reads after an AD conversion. This is the *quantized control problem*.

A. Predicates

We denote with $X = [x_1, \ldots, x_n]$ a finite sequence of variables. Each variable x ranges on a known (bounded or unbounded) interval $\mathcal{D}_x$ either of the reals or of the integers (discrete variables). We denote with $\mathcal{D}_X$ the set $\prod_{x \in X} \mathcal{D}_x$. Boolean variables are discrete variables ranging on the set $\mathbb{B}$

ISBN: 978-1-61208-226-4

= {0, 1}. Unless otherwise stated, we suppose real variables to range on $\mathbb{R}$ and integer variables to range on $\mathbb{Z}$.

A *linear expression* over a list of variables X is a linear combination of variables in X with rational coefficients. A *linear constraint* over X (or simply a *constraint*) is an expression of the form $L(X) \leq b$, where $L(X)$ is a linear expression over X and b is a rational constant. Finally, a *conjunctive predicate* is conjunction of constraints.

B. OBDD Representation for Boolean Functions

We will denote boolean functions $f : \mathbb{B}^n \to \mathbb{B}$ with boolean expressions on boolean variables involving $+$ (logical OR), $\cdot$ (logical AND, usually omitted thus $xy = x \cdot y$), $\bar{}$ (logical complementation) and $\oplus$ (logical XOR). We will also denote vectors of boolean variables in boldface, e.g., $\boldsymbol{x} = \langle x_1, \ldots, x_n \rangle$. Moreover, we also denote with $f|_{x_i=g}(\boldsymbol{x})$ the boolean function $f(x_1, \ldots, x_{i-1}, g(\boldsymbol{x}), x_{i+1}, \ldots, x_n)$ and with $\exists x_i \;\; f(\boldsymbol{x})$ the boolean function $f|_{x_i=0}(\boldsymbol{x}) + f|_{x_i=1}(\boldsymbol{x})$. A *truth assignment* μ is a partial map from a set of boolean variables $\mathcal{V}$ to $\mathbb{B}$. A *minterm* of μ is a total extension of μ, i.e., a total truth assignment ν s.t. $\mu(x) \neq \perp \to \nu(x) = \mu(x)$ for all $x \in \mathcal{V}$. The *value* of a minterm (or of a total truth assignment) ν is $\sum_{i=1}^{n} 2^{i-1}\nu(x_i)$, being $\mathcal{V} = \{x_1, \ldots, x_n\}$.

An *OBDD with complemented edges* (COBDD [14][15][16]) is a rooted directed acyclic graph (DAG) with the following properties. Each node v is labeled either with a boolean variable $\mathrm{var}(v)$ (an internal node) or with $1 \in \mathbb{B}$ (the unique terminal node **1**). Each internal node v has exactly two children, labeled with $\mathrm{high}(v)$ (representing the case in which $\mathrm{var}(v)$ is true) and $\mathrm{low}(v)$ ($\mathrm{var}(v)$ is false). Moreover, $\mathrm{low}(v)$ may be complemented, depending on a label $\mathrm{flip}(v)$ being true. Finally, on each path from the root to a terminal node, the variables labeling each internal node must follow the same ordering. The semantics of a COBDD internal node v w.r.t. a flipping bit b, with $\mathrm{var}(v) = x$, is the boolean function

$$[\![v, b]\!] := x[\![\mathrm{high}(v), b]\!] + \bar{x}[\![\mathrm{low}(v), b \oplus \mathrm{flip}(v)]\!]$$

C. Most General Optimal Controllers

A *Labeled Transition System* (LTS) is a tuple $\mathcal{S} = (S, A, T)$ where S is a finite set of states, A is a finite set of *actions*, and T is the (possibly non-deterministic) *transition relation* of $\mathcal{S}$. A *controller* for an LTS $\mathcal{S}$ is a function $K : S \times A \to \mathbb{B}$ enabling actions in a given state. We denote with $\mathrm{Dom}(K)$ the set of states for which a control action is enabled. An LTS *control problem* is a triple $\mathcal{P} = (\mathcal{S}, I, G)$, where $\mathcal{S}$ is an LTS and $I, G \subseteq S$. A controller K for $\mathcal{S}$ is a *strong solution* to $\mathcal{P}$ iff it drives each *initial* state $s \in I$ in a *goal* state $t \in G$, notwithstanding nondeterminism of $\mathcal{S}$. A strong solution K^* to $\mathcal{P}$ is *optimal* iff it minimizes path lengths. An optimal strong solution K^* to $\mathcal{P}$ is the *most general optimal controller* (we call such solution an *mgo*) iff in each state it enables all actions enabled by other optimal controllers. For more formal definitions of such concepts, see [7]. For efficient algorithms to compute mgos starting from suitable (nondeterministic) LTSs, i.e., see [17].

D. Discrete Time Linear Hybrid Systems

In this section we introduce the class of discrete time Hybrid Systems that we use as plant models, namely *Discrete Time Linear Hybrid Systems* (DTLHSs for short). For a more complete introduction, see [10].

Definition 1. A *Discrete Time Linear Hybrid System* is a tuple $\mathcal{H} = (X, U, Y, N)$ where: X is a finite sequence of *present state* variables (we denote with X' the sequence of *next state* variables obtained by decorating with $'$ all variables in X); U is a finite sequence of *input* variables; Y is a finite sequence of *auxiliary* variables; $N(X, U, Y, X')$ is a conjunctive predicate over $X \cup U \cup Y \cup X'$ defining the *transition relation* (*next state*) of the system. Note that X, U, Y may contain discrete as well as continuous variables.

DTLHSs may be used to represent many interesting real-world plants, such as e.g., the buck DC/DC converter with multi inputs used in Section V [9].

Given a DTLHS $\mathcal{H}$ = (X, U, Y, N), we define LTS($\mathcal{H}$) = ($\mathcal{D}_X$, $\mathcal{D}_U$, $\tilde{N}$) where: $\tilde{N} : \mathcal{D}_X \times \mathcal{D}_U \times \mathcal{D}_X \to \mathbb{B}$ is a function s.t. $\tilde{N}(x, u, x') \equiv \exists\; y \in \mathcal{D}_Y \;\; N(x, u, y, x')$. A *state* x for $\mathcal{H}$ is a state x for LTS($\mathcal{H}$). A DTLHS control problem $\mathcal{P} = (\mathcal{H}, I, G)$ is defined as the LTS control problem (LTS($\mathcal{H}$), I, G). To accommodate quantization errors, always present in software based controllers, it is useful to relax the notion of control solution by tolerating an (arbitrarily small) error ε on the continuous variables. Accordingly, we look for controllers that drive the plant to the goal G with an error at most ε (we call such a controller a *ε-solution* to $\mathcal{P}$). Such an error is defined by the given *quantization* for the DTLHS.

In classical control theory the concept of *quantization* has been introduced (e.g., see [18]) in order to manage real valued variables. Quantization is the process of approximating a continuous interval by a set of integer values. Formally, a *quantization function* γ for a real interval $I = [a, b]$ is a non-decreasing function $\gamma : I \mapsto \mathbb{Z}$ s.t. $\gamma(I)$ is a bounded integer interval. Finally, a *quantization* $\mathcal{Q} = (A, \Gamma)$ for a DTLHS encloses quantization functions Γ for all state variables as well as the bounded (safe) *admissible region* A on which the desired controller is supposed to work. Namely, A bounds both state variables (subregion A_X) on which the controller has to keep the system and action variables (subregion A_U) on which the controller works.

A control problem admits a *quantized* solution if control decisions can be made by just looking at quantized values. This enables a software implementation for a controller.

Definition 2. Given a quantization $\mathcal{Q}$, a $\mathcal{Q}$ *Quantized Feedback Control* (QFC) solution to a DTLHS control problem $\mathcal{P}$ is a $\|\Gamma\|$ solution $K(x, u)$ to $\mathcal{P}$ such that $K(x, u) =$

$\hat{K}(\Gamma(x), \Gamma(u))$, where $\hat{K} : \Gamma(A_X) \times \Gamma(A_U) \to \mathbb{B}$ and $\|\Gamma\|$ is the size of the largest interval of values that are mapped to the same quantized value.

For efficient (non-complete) algorithms to compute QFC solutions to a DTLHS control problem, e.g., see [7].

IV. Automatic Visualization of Control Software

In this section, we describe (Algorithms 1 and 2) our method to automatically generate a 2D picture describing a $\mathcal{Q}$ QFC solution K to a DTLHS control problem $\mathcal{P} = (\mathcal{H}, I, G)$ with a given quantization $\mathcal{Q} = (A, \Gamma)$.

The picture we generate lies on a 2D cartesian plane, where each axis is labeled with a state variable of $\mathcal{H}$ and has a range bounded by A. Then, a point (x, y) in the picture is colored depending on which actions set is enabled by K in the DTLHS state (x, y), i.e., on

$$c(x, y) = \{u \mid K((x, y), u) = 1\}$$

If $\mathcal{H}$ has $\ell{+}2$ state variables, then the actions set we consider is $c(x, y) = \{u \mid \exists d_1, \ldots, d_\ell K((x, y, d_1, \ldots, d_\ell), u) = 1\}$. Note that such a picture is practically useful if $\mathcal{H}$ has at least two real variables, which is indeed the case in most real-world SBCSs. Finally, a second picture showing the correspondence between actions sets and colors is also generated.

A. Input and Output

The above is performed by our main function *Visualize* (described in Algorithm 1), which takes as input:

- a DTLHS plant model $\mathcal{H} = (X, U, Y, N)$;
- a quantization $\mathcal{Q} = (A, \Gamma)$ for $\mathcal{H}$;
- a subset $\Xi \subseteq X$ of plant state variables s.t. $|\Xi| = 2$; variables in Ξ are those to be shown in the axes of the final 2D picture;
- a $\mathcal{Q}$ QFC solution K to a control problem involving $\mathcal{H}$. By Definition 2, K is based on a controller $\hat{K}$ that only looks at integer (quantized) values. Thus, by considering the boolean encoding of such values (as it is usual in Model Checking Applications), $\hat{K}$, and by abuse of notation K, can be represented as a COBDD ρ, a node v of ρ and a flipping bit b s.t. $[\![v, b]\!] = K$.

The output of *Visualize* is a Gnuplot [8] source files pair (P, C) describing the picture P to be generated and the color legend C. Note however that *Visualize* may be easily adjusted to work with any other graphing tool, provided that it generates pictures from textual descriptions. In Algorithm 1, we represent P as a list of rectangles in the plant state space (restricted to variables in Ξ). To each rectangle, we associate the RGB code of the corresponding color to be displayed. Analogously, C is a list of colored rectangles with height equal to the height of the picture: on the x axis the actions set corresponding to each colored rectangle is shown.

B. Algorithm Details

Function *Visualize* works as follows. First of all, in line 2, state bit variables encoding plant state variables not in Ξ (i.e., those *not* to be displayed in the final picture) are existentialized out from K, thus obtaining COBDD node v' and flipping bit b' such that $[\![v', b']\!] = \exists v_1, \ldots, v_\ell [\![v, b]\!] = \exists v_1, \ldots, v_\ell K = \tilde{K}$. As a result, the final picture will show all values for plant state variables in Ξ s.t. there exists at least a value for all plant state variables in $X \setminus \Xi$ that is controlled by K.

The workflow of the remaining lines is as follows. In order to obtain a better compression, controllers are typically represented with COBDDs where action bit variables come first in the variables ordering; this is also the case for [7]. In order to generate the desired picture, we reverse such order by placing state bit variables before action bit variables (line 4), thus obtaining a new COBDD ρ'. Since there always exists a COBDD representing a given boolean formula, in the new COBDD ρ' there will be a node v'' s.t. $[\![v'', b']\!] = \tilde{K}$. This allows us to perform a depth-first visit (DFS) of the COBDD representing $\tilde{K}$, by calling (line 5) function *CreateGnuplotBody* described in Algorithm 2. Namely, function *CreateGnuplotBody* returns a list M of (μ, v, b) triples s.t. μ is a total truth assignment to state bit variables with value $\hat{x}$, and for all plant states x in the quantized state $\hat{x}$ (i.e., such that $x \in \Gamma^{-1}(\hat{x})$) K enables the set of actions u s.t. the boolean encoding of u satisfies $[\![v, b]\!]$.

In order to achieve this goal, function *CreateGnuplotBody* of Algorithm 2 starts a depth-first visit (DFS) of ρ' from node v'' with flipping bit b'. On each path from v'' to **1**, the DFS stops as soon as an action bit variable is found at node z (i.e., $\mathrm{var}(z)$ is part of plant action variables U encoding) with flipping bit c. While exploring such a path, the corresponding truth assignment μ is maintained, i.e., if the then edge of a node w has been traversed, then $\mu(\mathrm{var}(w)) = 1$ (lines 5–6); if the else edge has been traversed, then $\mu(\mathrm{var}(w)) = 0$ (lines 7–9). Moreover, if a complemented edge is traversed, the flipping bit b is flipped (line 8). Once, in line 1, a node z is found s.t. $\mathrm{var}(z)$ is an action bit variable, or directly **1** is encountered (meaning that all actions are enabled by K for the quantized states corresponding to values of minterms of μ), the to-be-returned list M is updated (lines 2–3) by adding all minterms of the current μ together with the pair (z, b).

Once function *CreateGnuplotBody* has finished, the returned list M may be directly translated in a Gnuplot file P as follows. For each triple (μ, v, b) in M, the value $\hat{x}$ of μ is translated in a rectangle having as bounds those of $\Gamma^{-1}(\hat{x})$, i.e., of the cartesian product of the intervals that are mapped to $\hat{x}$ (line 10). The RGB color of such a rectangle may be determined starting from the address (a C language pointer) of (v, b). However, this has the following drawbacks: i) the Gnuplot file for the picture may be too big; ii) different runs of function *Visualize* (e.g., with different quantizations, and thus different boolean encoding, for plant state variables) may result in different colors for equal actions sets, which may make difficult an effective comparison between different experiments. In order to counteract i), M is compacted, by collapsing contiguous quantized states with the same action sets (function *CompactRectangularRegions* in line 7 of Algorithm 1). To avoid ii), we first generate all possible 2^{2^r} colors (line 8, using an approach similar to [19]) and we use a lexicographical ordering on action sets to pick one of such colors. Finally, the Gnuplot file C maintaining the correspondence between colors and action sets is generated in lines 11–12, where SatAll returns all satisfying minterms of the given COBDD (boolean function).

Algorithm 1 Visualizing a controller.

```
Require: DTLHS $\mathcal{H}$, quantization $\mathcal{Q}$, state variables set $\Xi$
    s.t. $|\Xi| = 2$, COBDD $\rho$, node $v$, boolean $b$
Ensure: Visualize($\mathcal{H}, \Xi, \rho, v, b$):
 1: let $v_1, \ldots, v_\ell$ be the state bit variables encoding plant
    variables in $\Xi$
 2: let $v', b'$ be s.t. $[\![v',\, b']\!] = \exists v_1, \ldots, v_\ell [\![v, b]\!]$
 3: let $w_1, \ldots, w_r, w_{r+1}, \ldots, w_{n+r}$ be the current bit variables ordering in $\rho$, being $r$ (resp. $n$) the number of
    action (state) bits variables
 4: modify the ordering in $w_{r+1}, \ldots, w_{n+r}, w_1, \ldots, w_r$;
    call $\rho'$ the resulting COBDD and $v''$ the node of $\rho'$ s.t.
    $[\![v'', b']\!]_{\rho'} = [\![v', b']\!]_\rho$
 5: $M \leftarrow$CreateGnuplotBody($\rho', v'', b', w_1, \bot, \varnothing$)
 6: for all $i \in [|\alpha|]$ do
 7:     $M \leftarrow$CompactRectangularRegions($M, i$)
 8: $\chi \leftarrow$DifferentColorsRGB($2^{2^r}$)
 9: for all triples $(\mu, v, b) \in M$ do
10:     using $\mathcal{Q}$, append to $P$ the rectangle corresponding to
        $\mu$ with color $\chi_{lexOrder(v,b)}$
11: for all $(v, b)$ s.t. $\exists(\mu, v, b) \in M$ do
12:     append to $C$ a rectangle of color $\chi_{lexOrder(v,b)}$ with
        label $\mathrm{SatAll}(\rho', v, b)$
13: return $\langle P, C \rangle$
```

V. Experimental Results

We implemented our picture generation algorithm in C programming language, using the CUDD package for OBDD based computations and BLIF files to represent input OBDDs. We name the resulting tool KPS (*Kontroller Picture Synthesizer*). KPS is part of a more general tool named QKS (*Quantized feedback Kontrol Synthesizer* [7]). In this section we present our experiments that aim at evaluating effectiveness of KPS.

1) Experimental Settings: We present experimental results obtained by using KPS on given COBDDs $\rho_1, \ldots, \rho_4$ and DTLHSs $\mathcal{H}_1, \ldots, \mathcal{H}_4$ s.t. for all $i \in [4]$ ρ_i represents the mgo $K_i(\boldsymbol{x}, \boldsymbol{u})$ for a *buck DC/DC converter with i inputs* (see [9] for a description of this system) modeled by $\mathcal{H}_i$, where quantization $\mathcal{Q}$ is s.t. $n = |\boldsymbol{x}| = 20$ and $r_i = |\boldsymbol{u}| = i$. K_i is an intermediate output of the QKS tool described in [7]. For each ρ_i, we run KPS so as to compute *Visualize*($\mathcal{H}_i$, $\mathcal{Q}$, X, ρ_i, v_i, b_i) (see Algorithm 1). All our experiments have been carried out on a 3.0 GHz Intel hyperthreaded Quad Core Linux PC with 8 GB of RAM.

Algorithm 2 Visualizing a controller: Gnuplot body.

```
Require: COBDD $\rho$, node $v$, boolean $b$, first action bit
    variable $a$, truth assignment $\mu$, (assignment, COBDD
    node, flipping bit) triples set $M$
Ensure: CreateGnuplotBody($\rho, v, b, a, \mu, M$):
if $(v = \mathbf{1} \wedge \neg b) \vee (v \neq \mathbf{1} \wedge \mathrm{var}(v) > a)$ then
    for all minterms $\nu$ of $\mu$ do
        $M \leftarrow M \cup (\nu, v, b)$
else if $v \neq \mathbf{1}$ then
    $\mu(\mathrm{var}(v)) \leftarrow 1$
    $M \leftarrow$CreateGnuplotBody($\rho, \mathrm{high}(v), b, a, \mu, M$)
    $\mu(\mathrm{var}(v)) \leftarrow 0$
    if $\mathrm{flip}(v)$ then $b \leftarrow \neg b$
    $M \leftarrow$CreateGnuplotBody($\rho, \mathrm{low}(v), b, a, \mu, M$)
return $M$
```

Table I
KPS PERFORMANCE (CPU TIMES ARE IN SECONDS).

r	CPU(P)	CPU(G)	$\lvert P\rvert$	$\lvert J\rvert$	$\lvert E\rvert$
1	9.15e+00	3.25e+02	6.17e+03	2.46e+01	5.19e+03
2	1.00e+01	1.47e+03	1.29e+04	2.91e+01	1.09e+04
3	1.06e+01	2.43e+03	1.67e+04	2.91e+01	1.39e+04
4	1.10e+01	3.58e+03	2.02e+04	3.16e+01	1.68e+04

2) KPS Performance: In this section we will show the performance (in terms of computation time and output size) of the algorithms discussed in Section IV. Table I show our experimental results. The i-th row in Table I corresponds to experiments running KPS so as to compute *Synthesize*($\mathcal{H}_i$, $\mathcal{Q}$, X, ρ_i, v_i, b_i). Columns in Table I have the following meaning. Column r shows the number of action variables $|\boldsymbol{u}|$ (note that $|\boldsymbol{x}| = 20$ on all our experiments). Column *CPU(P)* shows the computation time of KPS, i.e., of function *Visualize* of Algorithm 1 (in seconds). Columns $|P|$, $|J|$ and $|E|$ show the size in KB of, respectively, the source Gnuplot file for the 2D picture (i.e., the output P of function *Visualize* of Algorithm 1), the JPEG file generated by Gnuplot from P (i.e., with compression), and the EPS file generated by Gnuplot from P (i.e., without compression). Finally, Column *CPU(G)* shows the computation time of Gnuplot (in seconds) to generate the JPEG and the EPS files (computation time and size for file C are negligible).

From Table I we can see that, in slightly more than 10 seconds we are able to generate the Gnuplot file for the multi-input buck with $r = 4$ action variables. Then, Gnuplot needs about one hour to synthesize the actual picture (either in JPEG or in EPS).

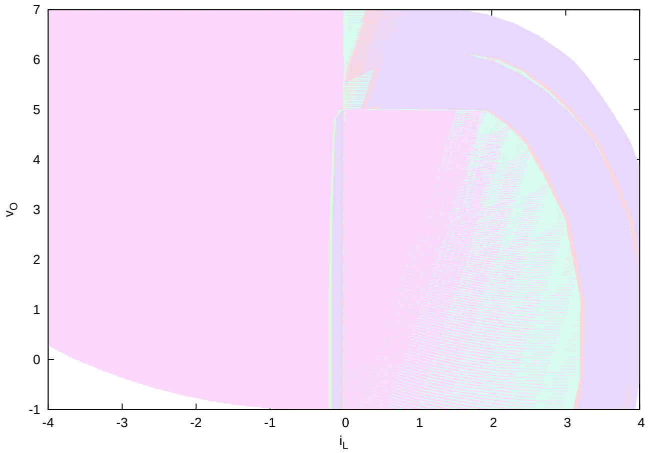


Figure 2. KPS+Gnuplot generated picture (P) for K_2.

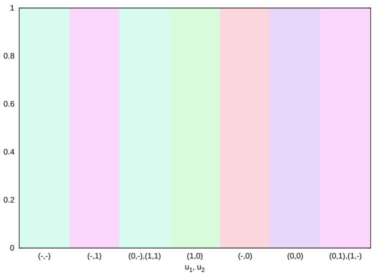


Figure 3. KPS+Gnuplot generated picture (C) for K_2.

3) KPS Evaluation: In Figures 2 and 3 we show the pictures generated by the KPS–Gnuplot chain for K_2. First of all, from Figure 3 we note that only 7 actions sets out of $2^{2^2} = 16$ are indeed enabled in K. Moreover, from Figure 2 we may immediately see that K indeed covers nearly all the admissible region of the buck converter. Finally, combining the two figures, we may see that the actions set $\{(-, 1)\}$ (i.e., $u_2 = 1$ and u_1 may be either 1 or 0) is the most used one.

VI. Conclusions

In this paper, we addressed the problem of visualizing a controller K for a DTLHS modeling an embedded system (plant). To this aim, we presented an algorithm and a tool KPS implementing it, which, from an OBDD representation of K, effectively generates a 2D picture depicting K. Such picture consists on a cartesian plane where each point corresponds to a state of the starting DTLHS, and colors with the same color all regions of states for which the same actions set is defined on K. A separated picture showing the relation between a color and the corresponding actions set is also automatically generated. In this way, the state region for which any color is shown depicts the coverage of K, whilest the regions colors give a glimpse of which actions are turned on by K on given plant states regions. We have shown feasibility of our proposed approach by presenting experimental results on using it to visualize the controller for a multi-input buck DC-DC converter.

The proposed approach currently generates a 2D picture, which forces to focus on just two plant state variables. Thus, a natural possible future research direction is to investigate how to generate a 3D picture. Finally, a 3D bar picture may also be used if there are more than 2 state variables in the input DTLHS plant, in order to show for each quantized value of the variables to be shown (i.e., those in Ξ) the percentage of coverage w.r.t. variables not to be shown (i.e., not in Ξ).

Acknowledgments: We are grateful to our anonymous referees for their helpful comments. Our work has been partially supported by: MIUR project DM24283 (TRAMP) and by the EC FP7 project GA218815 (ULISSE).

References

[1] T. A. Henzinger and J. Sifakis, “The embedded systems design challenge,” in *FM’06*, LNCS 4085.

[2] T. Henzinger, P.-H. Ho, and H. Wong-Toi, “Hytech: A model checker for hybrid systems,” *STTT*, 1(1), pp. 110–122, 1997.

[3] G. Frehse, “Phaver: algorithmic verification of hybrid systems past hytech,” *STTT*, 10(3), pp. 263–279, 2008.

[4] H. Wong-Toi, “The synthesis of controllers for linear hybrid automata,” in *CDC’97*, pp. 4607–4612.

[5] C. Tomlin, J. Lygeros, and S. Sastry, “Computing controllers for nonlinear hybrid systems,” in *HSCC’99*, LNCS 1569.

[6] M. Mazo, A. Davitian, and P. Tabuada, “Pessoa: A tool for embedded controller synthesis,” in *CAV’10*, LNCS 6174.

[7] F. Mari, I. Melatti, I. Salvo, and E. Tronci, “Synthesis of quantized feedback control software for discrete time linear hybrid systems,” in *CAV’10*, LNCS 6174.

[8] “Gnuplot: http://www.gnuplot.info/,” accessed: Jul 31, 2012.

[9] F. Mari, I. Melatti, I. Salvo, and E. Tronci, “On model based synthesis of embedded control software,” in *EMSOFT’12*.

[10] F. Mari, I. Melatti, I. Salvo, E. Tronci. Quantized feedback control software synthesis from system level formal specifications. *CoRR*, abs/1107.5638v1, 2011.

[11] A. Girard, “Synthesis using approximately bisimilar abstractions: time-optimal control problems,” in *CDC’10*.

[12] M. J. Mazo and P. Tabuada, “Symbolic approximate time-optimal control,” *Systems & Control Letters*, 60(4), pp. 256–263, 2011.

[13] A. Girard, G. Pola, and P. Tabuada, “Approximately bisimilar symbolic models for incrementally stable switched systems,” *IEEE Trans. on Aut. Contr.*, 55(1), pp. 116–126, 2010.

[14] K. S. Brace, R. L. Rudell, and R. E. Bryant, “Efficient implementation of a bdd package,” in *DAC’90*.

[15] S. Minato, N. Ishiura, and S. Yajima, “Shared binary decision diagram with attributed edges for efficient boolean function manipulation,” in *DAC’90*, pp. 52–57.

[16] F. Mari, I. Melatti, I. Salvo, and E. Tronci, “From boolean relations to control software,” in *ICSEA’11*.

[17] A. Cimatti, M. Roveri, and P. Traverso, “Strong planning in non-deterministic domains via model checking,” in *AIPS’98*.

[18] M. Fu and L. Xie, “The sector bound approach to quantized feedback control,” *IEEE Trans. on Automatic Control*, 50(11), pp. 1698–1711, 2005.

[19] “How to generate random colors programmatically: http://martin.ankerl.com/2009/12/09/how-to-create-random-colors-programmatically/,” accessed: Jul 31, 2012.

[20] F. Mari, I. Melatti, I. Salvo, and E. Tronci, “Synthesis of quantized feedback control software for discrete time linear hybrid systems,” in *CAV’10*, LNCS 6174.

Parallel Interference Cancellation in DS-OCDMA System Using Novel Multilevel Periodic Codes

Besma Hammami
National Engineering School of Tunis, Tunisia
hammamibesma6@gmail.com

Habib Fathallah
King Saud University
Riyadh, Saudi Arabia
habib.fathallah@gmail.com

Houria Rezig
National Engineering School of Tunis, Tunisia
houria.rezig@enit.rnu.tn

Abstract— **In this paper, we introduce the optimization of Bit Error Rate (BER) in parallel cancellation of multiple access interference (PIC) using a novel periodic optical encoder applied to fiber-to-the-X (FTTX) passive optical network (PONs) with a direct sequence optical code division multiple access (DS-OCDMA) system. The principle of this structure of receiver consists to reduce the output error in the data received. The performance of our system is analyzed in a synchronous network using multilevel periodic codes (ML-PC) and the results are compared with those for different receivers.**

Keywords- *direct-sequence optical code-division multipleaccess (DS-OCDMA); fiber-to-the-X (FTTX); passive optical network (PONs); multilevel periodic codes (ML-PC); parallel interference cancellation (PIC).*

I. INTRODUCTION

Direct-sequence code-division multiple access (DS-CDMA) [1] is currently the subject of much research as it is a promising multiple access capability for third and fourth generations mobile communication systems.

In Direct Sequence transmission, the user data signal is multiplied by a code sequence. Mostly, binary sequences are used. To obtain better performance than those obtained by the detection single-user, multiuser detection has been investigated for links OCDMA [2][3].

Indeed, this type of detection, already used for the radio CDMA has proven its efficacy in reducing the impact of interference on performance [4].

The advantage of the multiuser detection over single-user detection is the knowledge of codes of undesired users that evaluates more precisely the interference present in the received signal. Consequently, the data are better detected.

In this paper, we present a parallel cancellation method (called PIC) developed for radiofrequency systems, applied to the direct sequence optical CDMA system, the spreading codes considered here are achieved with a new periodic coding scheme [5], that has been previously proposed for FTTX monitoring, and to the best of our knowledge never explored for data coding/decoding. The receiver studied here is constituted by a limiter optical device placed in front of a PIC structure.

Our study is done when the direction of data transmission is the uplink direction, from Optical Network Unit (ONU), to Optical Line Termination (OLT). Using the DS-OCDMA technique for the upstream, would provide necessary bit rate, dispensing of synchronization for this track. The bit error rate (BER) performances were reported in the case of an optical synchronous incoherent DS-OCDMA system using multilevel periodic codes (ML-PC) when applied to FTTX-PON architecture.

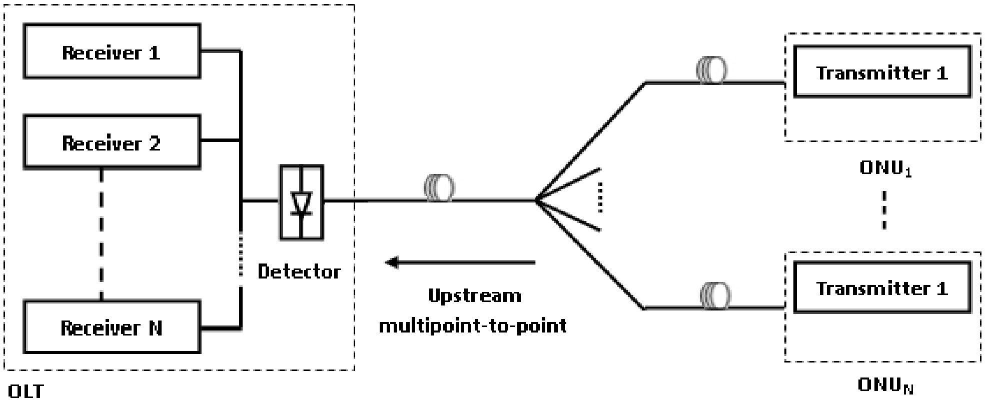


Figure 1. Direct Sequence OCDMA system

In this paper we compared the efficacy of the receptor PIC with the conventional correlation receiver (CCR), and then with their amelioration which is the parallel interference cancellation with an optical limiter (called HL+PIC), we deduce the superiority of HL+PIC structure not only in performance but also in regards to feasibility.

This paper is organized as follows: In the second section, we present the description of the DS-OCDMA system. In the third section we introduce the principle of the parallel interference cancellation structure and their improvement. In the fourth section, we evaluate the performance of the proposed system through the bit error rate (BER).

II. SYSTEM MODEL

In a DS-OCDMA system, users transmit binary data equiprobable and independently in an optical fiber. Differentiation of users is done by multiplying the data by a code (Figure 1). This code should be specific to each user, so that we can extract the data by comparing the received signal with the desired user code.

The codes studied in this paper are the multilevel periodic codes (ML-PC) [5], which are determined by the length of the silent intervals separating the multilevel pulses, i.e, its period. The codes length of the ith customers (*l_{ci}*) is related by the silent period between the subpulses and is given as:

$$l_{ci}=p_iwT_sc \tag{1}$$

where c is the speed of light, $p_i=l_i/cT_s$ is an integer number that determines the length of the ith encoders ring l_i, Ts is the transmitted pulse duration and w is the weight of the code (c_i).

In DS-OCDMA system the data of active users are spread by multiplication with the code sequence, and at the output of the encoder the kth user signal is obtained as:

$$S_k(t)=a_kb_k(t)c_k(t) \tag{2}$$

a_k The power level at the output of encoder and b_k is the data transmitted by the kth user. In the case of multilevel periodic codes (ML-PC), the total power for any code with weight [4] is:

$$P_t = \sum_{j=1}^{w} \rho_j \tag{3}$$

ρ_j is the jth subpulse power level generated by the encoder. The first subpulse power level ρ_1 is equal to $\rho_1=s^2$. . For j=2,…., the level of ρ_j can be derived as:

$$\rho_j=(1-s)^2s^{j-1}+(1-s)\ \rho_{j-1} \tag{4}$$

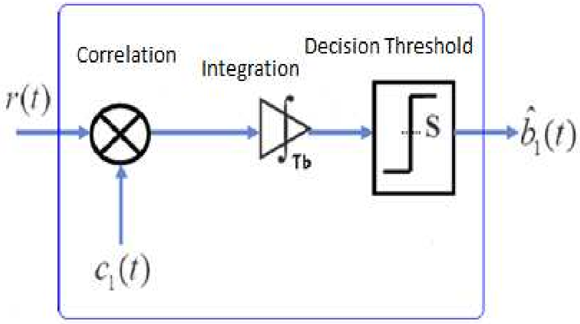


Figure 2. Conventional Correlation Receiver for user 1

s is the power coupling ratio which determines the amount of power coupled to the ring encoder proposed in [5]. It was shown in [5] that the interval of s between 0.5 and 0.6 gives good distribution for the power between the subpulses with cumulative power that depends on the weight w.

Finally, at the input of the receiver, the signal S(t) is the superposition of signals transmitted by the N users:

$$S(t) = \sum_{k=1}^{N} S_k(t-\tau_k) \tag{5}$$

A) Principle of conventional correlation receiver

The conventional correlation receiver (CCR) is the simplest receiver in a DS-OCDMA system, the principle of this receiver is the estimation of the power contained in the chips unit code, to compare thereafter to the decision threshold. It provides three functions:

- Multiplying the received signal by the code of the desired user. This step, equivalent to the realization of a mask between the received signal and the code sequence, can retain only the power present in the chip unit code,
- Integration of the signal obtained on the bit time: This step evaluates the total power present on the signal previously obtained during the interval of a bit time. This step provides the value of the decision variable.
- Decision making by comparison to a threshold: comparing the decision variable with the decision threshold used to obtain the estimated data.

Assuming that the user # 1 is the desired user, the decoding part of the DS-OCDMA system is performed by correlation (Figure 2).

B) Principle of parallel interference cancellation receiver

In a structure with parallel cancellation, all undesired users are detected at the same time using the conventional receiving systems.

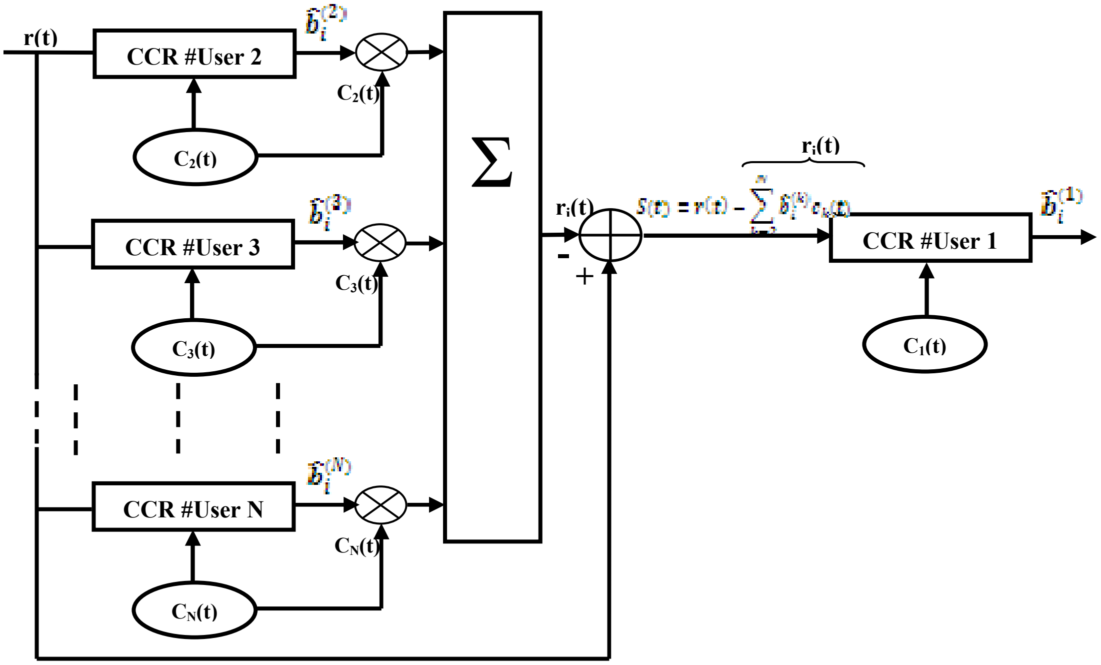


Figure 3. Schematic of the PIC receiver

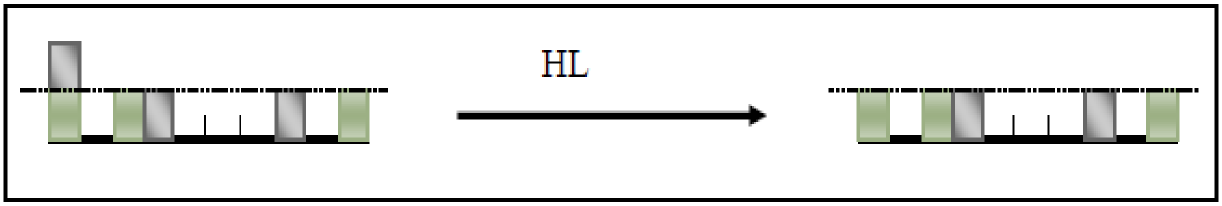


Figure 4. Effect of Hard Limiter on an example of received signal

The parallel interference cancellation receiver has the principle of the reproduction interference from undesired users, to remove it from the total received signal (Figure 3). The PIC requires several steps:

- The detection of data sent by each undesired user is done by the conventional correlation receiver (CCR) with a detection threshold “S_t”, at the output of each receiver, we obtain the estimation $\hat{b}_i^{(k)}$ of the data sent by the undesired user # k,
- The second step is to reconstruct the signals transmitted by undesired users by multiplying the estimated data $\hat{b}_i^{(k)}$ by the corresponding code $c_k(t)$,
- We obtain in the third step, the interference term $r_i(t)$ which is actually the sum of the reconstructed signals, then it is subtracted from the received signal r (t): $S(t) = r(t) - r_i(t)$ and as, $r_i(t) = \sum_{k=2}^{N} \hat{b}_i^{(k)} c_k(t)$, then:

$$S(t) = r(t) - \sum_{k=2}^{N} \hat{b}_i^{(k)} c_k(t) \qquad (6)$$

- The last step is the detection of the desired user data # 1 from the signal "cleaned" from the interference S(t). This detection is done through a CCR with a decision threshold S_f.

C) Amelioration

1) Principle of hard limiter (HL)

The ideal function of the component called "Hard Limiter" (HL) is defined by:

$$g(x)\begin{cases}1 & x \geq 1\\ 0 & 0 \leq x < 1\end{cases} \qquad (7)$$

In practice, this component removes a part of the received power to get at the end a signal which each chip contains a power equal 0 or 1. For example, in Figure 4, we observe that the HL removed a part of the power contained

ISBN: 978-1-61208-226-4

in the first chip, and left unchanged the rest of the signal. Indeed, the power contained in the first chip of the received signal has a value of 2, while the one in the same chip after the action of HL is 1.

Thus, the HL has eliminated a part of the interference contained in the first chip. On the other side, the chips containing a power equal to 1 before the HL remain unchanged, and those for which the power was zero. As a result, levels 0 and 1 will be unchanged, and levels greater than 1 will be reduced to one. This limitation of the power in each chip reduces the interference, and removes some interference patterns leading to an error.

2) HL+PIC

To improve the performance of the PIC, the detection of undesired users can be achieved by a HL + CCR receiver. Thanks to the limiters placed before the receivers of the undesired users, the data are therefore better estimated so the contribution of these users in the received signal is better evaluated.

III. PERFORMANCE EVALUATION

We will present in this section the algorithm used in our simulation and we will analyze the results.

A) Numerical simulation

At the transmitter of the DS-OCDMA channel, we begin by the generation of periodic codes and then the random generation of bits sent by each user and random selection of N active users among users of the family, afterwards the step of the spreading is done by multiplying the data of the desired user by the corresponding code, subsequently the spreading of data of the undesired users and adding their contribution to the signal of the desired user. Finally, we sum the encoded data and transmit over a channel assumed to be ideal.

At the receiver, we will follow the different stages of the parallel interference cancellation structure described in Section II, and to analyze the performance of this structure multi-user, we will compare it with another receiver such as, the conventional correlation receiver (CCR), and the CCR improved by adding an optical limiter (known as Hard Limiter), and then the improved of PIC (HL+PIC).

B) Analysis of results

The simulation has been carried out in MATLAB to evaluate the BER performance for the parallel interference cancellation (PIC) and compared it with other receivers (CCR, HL+CCR, HL+PIC).

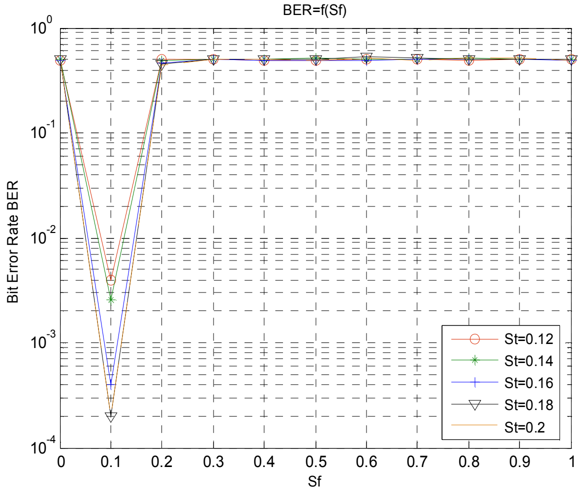


Figure 5. BER versus decision threshold of the undesired users Sf using ML-PC, N=6 Users

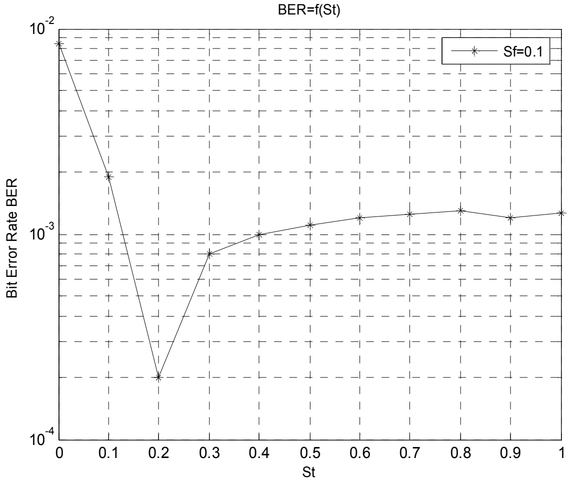


Figure 6. BER versus decision threshold of the desired users St using ML-PC, N=6 Users

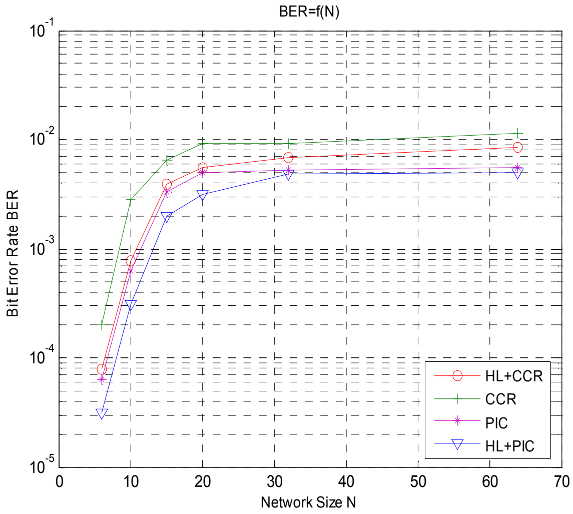


Figure 7. BER versus the network size N

So we must first determine the optimal thresholds (S_t: optimal threshold of the desired user, S_f: optimal threshold of the undesired users) of the PIC receiver.

In Figure 5, we plotted the evolution of the BER of the PIC receivers with ML-PC codes with period p_i, weight w=5, s=0.4 and N=6 users. This performance was evaluated as a function of the S_f and varying S_t between 0.12 and 0.2. From this presentation, we can observe that the best performance is obtained for a decision threshold $S_f = 0.1$ whatever the value of S_t.

Now, we will fix the value of S_f at 0.1 and we will present in Figure 6, the variation of BER as a function of S_t with the same ML-PC code and N=6 users. So we can look that the best performance is achieved when $S_t = 0.2$.

We can conclude that the two optimal thresholds are:

- The optimal threshold of the desired user: S_t=0.2,
- The optimal threshold of the undesired users: S_f=0.1,

We worked with the two optimal threshold estimated in the previous figures, and we plotted the variation of the BER as a function of the network size N (Figure 7), with the same ML-PC code. First, we can see that the performance of the four receivers degrade when the number of users increases, but does not exceed $2*10^{-2}$ and that thanks to the use of periodic codes.

Furthermore, we observe that for a given code, the PIC allows a number of active users more important than the CCR or HL+ CCR. Indeed, for a ML-PC code (with period p_i, w = 5 and s = 0.4) and BER = $5.5*10^{-2}$, the PIC allows 64 simultaneous users to communicate, while the CCR and HL+CCR allow only 20 users at most, to be active on the network.

By comparing the four receivers, one can conclude that the best performances are obtained when we work with a HL+PIC receiver and here the BER can achieve $3.125*10^{-5}$ for N = 6 users.

IV. Conclusion

In this paper, we investigated the multi-users detection with the parallel interference cancellation (PIC) structure by comparing it with their amelioration (HL+PIC) and other receivers (CCR and HL+CCR), using a novel coding scheme so called multilevel periodic coding (Ml-PC) for DS-OCDMA system. We studied the characteristics of these codes and investigated their performance in BER. We derived the values for an optimum threshold that minimizes the bit error rate when we use the PIC receiver. In our system, we can achieve almost a BER = $3.125*10^{-5}$ for N = 6 users.

References

[1] C. Goursaud, Naufal M. Saad, Y. Zouine, A. Vergonjanne, C. Aupetit-Berthelemot, J. Zaninetti, J.P. Cances et J.M. Dumas, “Application of temporal optical CDMA in access broadband networks” , 23rd National Days of Guided Optics (JNOG 2004), October 2004.

[2] Y. Zouine, S. M. Naufal, C. Goursaud, A. Julien-Vergonjanne, C. Aupetitberthlemot, J.P. Cances and J.M. Dumas «The influence of the optical successive interference cancellation in the optical CDMA network», XV IEE Int. Symp. On Service and Local Access 2004 (ISSLS 2004), Edinburgh, UK, 21-24 March 2004.

[3] C. Goursaud, S. M. Naufal, Y. Zouine, A. Julien-Vergonjanna, C. Aupetitberthlemot, J.P. Cances and J.M. Dumas «Performances of parallel cancellation (PIC) receivers in high-speed access optical-networks», Wireless and Optical Communications 2004 (WOC 2004), Banff, Canada, July 8-10, 2004, pp. 738-743.

[4] S. Moshavi, “Multi-user detection for DS-CDMA communications”, IEEE communication Magazine, pp 124-136, October 1996.

[5] Maged Abdullah Esmail and Habib Fathallah, “Novel Coding for PON Fault Identification” IEEE communications letters, vol. 15, no. 6, June 2011

ISBN: 978-1-61208-226-4

Using Embedded FPGA for Cache Locking in Real-Time Systems

Antonio Martí Campoy, Francisco Rodríguez-Ballester, Rafael Ors Carot
Departamento de Informática de Sistemas y Computadores
Universitat Politècnica de València
Spain
{amarti, prodrig, rors}@disca.upv.es

***Abstract*—In recent years, locking caches 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 locking caches in real-time systems. This work propose the use of this embedded programmable logic device to implement a logic function that provides the locking 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.**

***Keywords*-Real-Time Systems; Locking Caches; FPGA.**

I. Introduction

Cache memories are an important advance in computer architecture, giving 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 behavior of a cache memory reduces the average access time to main memory, but presents a non deterministic fetching time [5]. This way, estimating execution time of tasks is complicated. Furthermore in preemptive, multi-tasking systems, estimating the response time of every 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, locking caches 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 has been presented to apply locking caches in real-time, multi-task, preemptive systems, both for instructions [1][4][7] and data [8]. In this work, we focus on instruction caches only, because 75% of accesses to main memory are to fetch instructions [5].

A locking 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 locking cache memory.

One of the ways to use locking caches in preemptive real-time systems is called the dynamic use. In this way of using a locking cache, cache contents change only when a task starts or resumes its execution. Then, 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.

This paper is organized as follows. Section two describes previous implementation proposals for the dynamic use of a locking cache in real-time systems, and the pursued goals of this proposal to improve previous works. Section three presents a detailed implementation of the Locking State Generator (LSG), a logic function that signals to the cache controller whether to load a main memory block in cache. Section four presents some analysis about the complexity of the proposal, and Section five outlines some ideas about how to simplify the circuit complexity. Finally, this paper ends with the ongoing work and conclusions.

II. State of the Art

Two ways of implementing dynamic use of locking cache can be found in the bibliography. First of them, [1], 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 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 load and lock 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 locking cache, in [4] is introduced the Locking State Memory (LSM). This is a hardware solution where the loading of memory blocks in cache is controlled by a one-bit signal coming from a 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 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 locking cache controller. If the bit is set to 1, indicates 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 0, 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 cache contents remain unchanged.

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 size and cache-line size (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. This way, if the system has a small locking cache and a very big 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 (Field-Programmable Gate Array), 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, however this time hardware complexity is proportional to the size of system, both software-size and hardware-size. Not only the circuit required to dynamically use the locking cache may be reduced but also those parts of the FPGA not used for the control of the locking 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 locking cache.

III. The proposal: Locking State Generator

Recent devices for the embedded market [3][6] couple a processor and an FPGA (Field-Programmable Gate Array), 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 [3]) or even in a single die (like the Xilinx's Zynq-7000 series [6]) and may help to improve the use of locking caches 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.

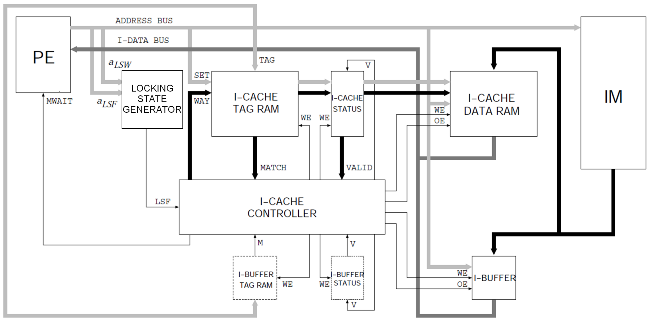


Figure 1. The LSG architecture.

This work proposes the substitution of the Locking State Memory 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 generator instead of the LSM. First, the LSG may adjust its complexity and circuit-related size to both the hardware and software characteristics. While the LSM size depends only on the main memory size and cache-line size, 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.

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 for 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,

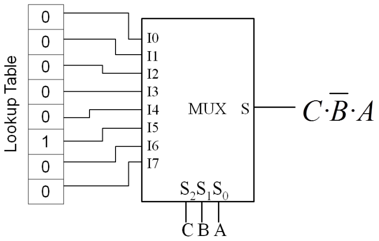


Figure 2. Implementing mini-term 5 of arity 3 (C, B, A are the function inputs).

but it is not a good option for the LSG: the total number of bits stored in the set of combined LUTs is 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 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 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 1 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 Lookup table will store a set of zero values and a unique one value. This one will be stored in the 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 [6]. 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, two questions arise: first, how many mini-terms belong to the function, and second, how to obtain the logic sum of all them.

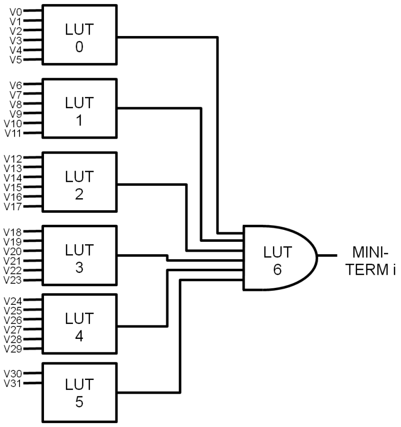


Figure 3. Implementing a 32-input mini-term using 6-input LUTs.

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 a locking cache, each task can use the whole cache for its own blocks.

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 $L = (32KB/4B)/4instructions = 2K$ lines.

This means that, for every task in the system, the maximum number of main-memory blocks that can be selected is around 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 all mini-terms are zero), or ii) the result will be $M-1$ and the final carry will be set to one (being M 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, and the final carry indicates if the referenced main memory block has to be loaded and

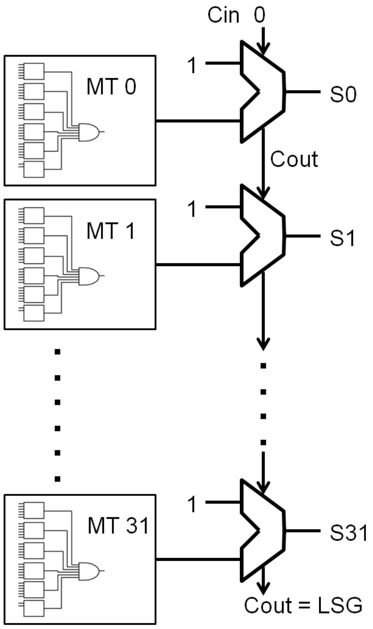


Figure 4. Implementing the LSG function.

locked in cache. Figure 4 shows a block diagram of this sum.

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 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 with a locking 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 forced to generate a fixed one value, obtaining a system with the same behavior as 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 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 32 bits, so main-memory blocks addresses are 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 [6]. 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.

V. Reducing complexity

The estimated value of 120 000 LUTs required to build the LSG function is an upper bound, and there are some ways this number may be reduced. A real-time system with five tasks will need just half this value of LUTs. Same if the cache size is divided by two.

In some cases, not all tasks will use the whole cache, that is, the number of selected blocks for a given task may be lower than the cache capacity, reducing the number of mini-terms in the LSG. In this aspect, the LSG improves the LSM because it better adapts to hardware and software characteristics of the system. Finally, as with any logic function implementation, there are well-known simplification algorithms that may be applied, reducing both the number of terms and their size (arity), which in turn reduce the number of LUTs required for the implementation.

This simplification may be improved by the selection algorithm. To use a locking cache, no matter the way it is used and how locking information is stored or generated, an off-line algorithm has to select those main memory blocks that will be loaded and locked in cache [2]. Usually, the target of these algorithms is to provide predictable execution times and improve the overall performance of the system and its schedulability, for example reducing global utilisation or enlarging the slack of tasks to allow scheduling non-critical tasks. But, new algorithms may be designed that take into account not only this main target, but also trying to select blocks with adjacent addresses, enhancing simplification and reducing 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— selecting the five first blocks will give the same performance than selecting the last five, or selecting alternate blocks. Previous research show that genetic algorithms applied to this problem produce different solutions, that is, different sets of selected main memory blocks but with the same results regarding performance and predictability.

This is a first approach to a new architecture, and many experiments are needed to precisely evaluate the complexity and cost of the LSG implementation, and to state the scenarios where its use is more suitable than using LSM. Number of LUTs detailed in this work are for the worst case, that is, real-time systems with many tasks, large cache memory and many main memory blocks selected to lock in cache. Not in all cases the upper bound of LUTs will be reached.

VI. Ongoing work

Next step 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, evaluation of implementation complexity will be accomplished.

Also, some design strategies have to be explored in detail in order to reduce the number of LUTs required to implement the LSG. In particular initial experiments from a design strategy merging LUTs from pairs of mini-terms show promising results as the number of bits to be added by the carry chains may be cut in half without a serious impact on the circuit operating frequency.

VII. Conclusion

This work presented a new way of implementing the dynamic use of locking cache for preemptive real-time systems. The proposal benefits from recent devices coupling a processor with a 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. 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 with 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 exclusively on the size of main memory.

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, ongoing research steps about the selection algorithm of main memory blocks and the LSG hardware implementation are outlined.

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. Marti Campoy, A. Perles Ivars, and J. V. Busquets 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.

[2] Antonio Marti Campoy, Isabelle Puaut, Angel Perles Ivars, and Jose Vicente Busquets Mataix. Cache contents selection for statically-locked instruction caches: An algorithm comparison. In *Proceedings of the 17th Euromicro Conference on Real-Time Systems*, pages 49–56, Washington, DC, USA, 2005. IEEE Computer Society.

[3] Intel Corp. Intel atom processor e6x5c series. http://www.intel.com/p/en_US/embedded/hwsw/hardware/atom-e6x5c/overview, 2012. [Online; accessed 16-June-2012].

[4] J.V. Busquets-Mataix E. Tamura and A. Mart 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)*, pages 75–84, 2007.

[5] John L. Hennessy and David A. Patterson. *Computer Architecture: A Quantitative Approach, 4th Edition*. Morgan Kaufmann, 4 edition, 2006.

[6] Xilinx Inc. Zynq-7000 extensible processing platform. http://www.xilinx.com/products/silicon-devices/epp/zynq-7000/index.htm, 2012. [Online; accessed 16-June-2012].

[7] Jan C. Kleinsorge Sascha Plazar and Peter Marwedel. Wcet-aware static locking of instruction caches. In *Proceedings of the 2012 International Symposium on Code Generation and Optimization*, pages 44–52, 2012.

[8] Xavier Vera, Björn Lisper, and Jingling Xue. Data cache locking for tight timing calculations. *ACM Trans. Embed. Comput. Syst.*, 7(1):4:1–4:38, December 2007.

Discovery & Refinement of Scientific Information via a Recommender System

Robert M. Patton, Thomas E. Potok, and Brian A. Worley
Computational Data Analytics
Oak Ridge National Laboratory
Oak Ridge, TN
{pattonrm, potokte, worleyba} @ ornl.gov

***Abstract*—The ability to maintain awareness within a field of research has been a hallmark of scientific expertise for centuries. As diverse scientific information becomes available through various Internet sources, not merely conference proceedings and journals, maintaining scientific awareness becomes a significant challenge. One challenge is how to discover sources that may be of interest. A second challenge lies in finding significant information over many sources. Scanning through hundreds of posts, feeds, and articles a day is very time consuming and error prone. Our approach uses a set of author published papers as seed documents to recommend documents of interest across various Internet sources. This enables 1) discovery of new sources that may be of interest, and 2) refine the information within a source to only the most relevant.**

Keywords-recommender system; text analysis; rss feeds

I. Introduction

Big data demands the need for intelligent, recommender agents that can enhance a person’s situational or domain awareness of their environment. The ability to have a keen awareness and availability of relevant information provides a critical competitive edge. Unfortunately, there is simply too much data streaming too quickly for a person to manually process, analyze, and take action within a reasonable amount of time. This challenge is true in research and academia, as well as industry and government, and has remained a challenge for quite some time [15]. In an attempt to alleviate this challenge, many people subscribe to relevant Internet information. There may be forms of subscriptions with the most common being Really Simple Syndication (RSS), blogs, even Facebook and Twitter. The concept is simple, when new information is posted to the site; a subscriber sees a list of this new information. The subscriber then has the option of following a link to read more. For researchers, the areas to monitor are fairly specific, for example, new research, publication opportunities (e.g., conferences and journals), funding opportunities, the activities of key people in your field, and inspiration for new ideas.

Traditionally, reading key journals and presenting at key conferences and workshops could accomplish this. Now, much of the data has moved to the Internet. The subscriber model is a very useful and successful model for monitoring this data, but it does have some significant drawbacks. In practice, the feeds of new information become quite lengthy, and contain more information than can be practically read. Furthermore, there can be a significant number of items that have little interest to the subscriber. A particular researcher may be strongly interested in another researcher's technical postings, but not interested that researcher's vacation or political postings. Another challenge is how to select the sources to subscribe to. A common model is to subscribe to what your community subscribes to, or to subscribe to the most followed sources. In doing so, it is very difficult to discover a new and interesting source that is not known by another researcher, or known in general. Thus, the ability to find new and relevant information proves critical.

We propose a content-based recommender system was designed and developed called Distribute The Highest Selected Textual Recommendation (DTHSTR) that addresses both of these problems, and is initially developed as a support tool for researchers. However, the flexibility of input allows the system to be adaptable to industry and government use cases as well. Recommender systems enable the filtering of information based on the relevance to or interests of the user. They are often used for e-commerce or entertainment, but rarely, if at all, for the research community in a widespread manner. Our approach attempts to fill this gap for the research community. The following sections will discuss related works, the approach, and an example use case using the approach described.

II. Related Works

There are many successful approaches to recommending research articles to a user. These approaches are typically applied solely against a corpus of research articles, not on broader information such as call for papers, call for proposals, or science news. Collaborative filtering analyzes information from article reviews of other users as the basis for recommending new articles. There are various machine learning methods to recommend articles based on the citations of other users. Text analysis methods have been used to compare the full or partial content of a set of interesting articles to a set of potential interesting articles using methods such as Term Frequency Inverse Document Frequency (TF-IDF), Latent Semantic Analysis (LSA), and Topic Modeling (TM) [1][6][8][12][13][17][30][31].

In this work our , corpus is a very large and dynamic collection of RSS feed documents, not a research article

ISBN: 978-1-61208-226-4

corpus. Since there are no reviews and limited citations (e.g., blog comments or Facebook Like), we have focused on comparing the full content of a user's articles against the full content of the RSS feeds. Given the large volume of documents, we are focusing on parallel enhancements to TF-IDF that we believe will perform faster and with less memory requirements than LSA or TM.

In [14], a recommendation system is described based on the similarity to user profiles (i.e., collaborative filtering). The system consists of several databases that contain documents produced over time from a research laboratory, and makes use of the TF-IDF [27][28] term weighting scheme. This work focuses on providing long-term and short-term components of representing a user profile. The long-term representation is created from natural language sentences as a vector space model using the TF-IDF term weighting. The short-term component is based on documents recently downloaded by the user. Profiles are then compared for similarity to each other. The primary drawback to this approach is the database of documents as input, as opposed to a richer, more current on-line data set.

In [33], a personalized recommendation system for scientific and technical papers is described. The system automatically summarizes the technical papers and then performs similarity comparisons to a user's query. While the authors do not clearly describe the details of their approach, the system appears to be dependent on keyword queries provided by the user. Furthermore, another weakness of the approach is the use of automated summarization prior to the similarity comparison, which is very likely to significantly impact the similarity comparison.

In [19], a multi-agent system for recommending scientific documents is presented. Various agents perform different tasks in collaboration with each other via a central profiling agent in order to provide a recommendation to the user. This system appears to be completely ubiquitous in that it simply monitors the users actions as a means of input for recommendations. Recommendations are based on either similarity to other users or similarity to documents that the user has either saved or bookmarked as well as other criteria.

In [6], a recommender system is developed called Scienstein. Like previous works, the system is intended specifically for searching academic papers, and uses both content-based and collaborative-based techniques. In addition, Scienstein performs citation, author, and source analysis as part of its recommendation. Despite the advanced analysis capabilities, there are a few drawbacks. The system deploys are user interface as a desktop application, although a newer version appears to be web-based. In addition, the system is oriented toward academic papers only and does not appear to be expandable to other sources of data.

In [34], an ontology based recommendation system for scholars is described. Unlike the works previously mentioned, this system makes use of search engines such as Google or Yahoo to perform domain specific searches. Results are then processed to extract information using an ontology database, and further refined with a information recommender. Unfortunately, the major drawback to this system is the use of a domain specific ontology. In their work, artificial intelligence ontology was used. In order to apply their system to another domain requires changing the ontology, which may be significantly challenging, or non-existent. In addition, the use of search engines enables the exploration of potentially unknown Internet sources. However, it does not guarantee that information will be monitored consistently from a particular source, and only as regularly as the search engine crawls the source of interest.

In [16], a content-based recommender system using a genetic algorithm is proposed. In this approach, the genetic algorithm is bootstrapped with 20-30 documents that represent a single category of interest. The documents are represented with a vector space model [25] using TF-IDF [27][28] term weighting scheme and cosine similarity metric. The genetic algorithm then evolves to build a classifier for recommending documents to the user. According to the authors, one of the main drawbacks is the bootstrapping process. Another drawback is that TF-IDF approach caused important terms in the user supplied documents to be lower weighted as a result of the documents being very similar to each other. As will be described later, the work described here avoids this through the use of a different term weighting scheme.

In [10], an ontology based recommendation system is developed for browsing web pages and makes use of long-term and short-term preferences. The ontology is built by analyzing book web pages from Amazon, while long-term and short-term preferences are developed by monitoring the web pages viewed by the user. As with the previous ontology-based system, the major drawback is the use of an ontology, which must be learned or developed depending on the domain. Furthermore, the preferences are based on web browsing alone. While this approach may work well for searching books on Amazon, it is not easily or accurately adapted to other domains.

Furthermore, other work [1][5][20][24][35] has focused on developing techniques for filtering RSS feeds. These works focus on RSS feeds in general with a tendency toward news feeds only. Our work uses a manual categorization of the feeds in order to identify content as being associated with a particular aspect of a researcher's workflow (e.g., funding, publications, patents). In addition, they predominantly rely on using some form of supervised learning that requires a training set. Of these works, the work of [35] is the most similar to this work. In [35], a vector space model and TF-IDF term weighting are used while a Rocchio feedback algorithm is used to adapt to the user. Our work uses a different term weighting scheme, and currently, does not support a feedback algorithm.

III. APPROACH

A. *Ingest*

The DTHSTR system requires two primary sources of input. The first source of input is a list of RSS feeds to be monitored. Currently, the system monitors more than 9,500 feeds from 130 sites. Table I shows a sample of the different sites that are monitored.

ISBN: 978-1-61208-226-4

TABLE 1. SAMPLE RSS FEEDS

Site	Category
Sciencedaily.com	News
Freshpatents.com	Inventions
Freepatentsonline.com	Inventions
Acm.org	Publications
Ieee.org	Publications
Nejm.org	Publications
Grants.gov	Call for Proposals
Wikicfp.com	Call for Papers

Documents from each feed are monitored and collected on a regular basis. For this particular use case, the second source of input is a set of recent publications for each researcher as a means of representing their research interests. Other documents could be used for different use cases. For this particular work, each publication supplied by the researcher is used individually to provide a reference point for Internet content. This allows the researcher to know which individual publication is most similar to the Internet content.

B. *Analysis*

In order to process and analyze the input feeds and publications, each document is converted into a collection of terms and associated weights using the vector space model method. The vector space model (VSM) is a recognized approach to document content representation [25] in which the text in a document is characterized as a collection (vector) of unique terms/phrases and their corresponding normalized significance.

Developing a VSM is a multi-step process. The first step in the VSM process is to create a list of unique terms and phrases. This involves parsing the text and analyzing each term/phrase individually for uniqueness using a term weighting scheme. The weight associated with each unique term/phrase is the degree of significance that the term or phrase has, relative to the other terms/phrases. For example, if the term “plan” is common across all or most documents, it will have a low significance, or weight value. Conversely, if “strategic” is a fairly unique term across the set of documents, it will have a higher weight value. The VSM for any document is the combination of the unique term/phrase and its associated weight as defined by a term weighting scheme.

In our approach, the term frequency-inverse corpus frequency (TF-ICF) developed in [22] is used as the term weighting scheme. Over the last three decades, numerous term weighting schemes have been proposed and compared [9][11][26][27]. The primary advantage of using TF-ICF is the ability to process documents in *O(N)* time rather than $O(N^2)$ like many term weighting schemes, while also maintaining a high level of accuracy. For convenience, the TF-ICF equation is provided here:

$$w_{ij} = \log(f_{ij}) \times \log(N / n_j) \qquad (1)$$

In this equation, f_{ij} represents the frequency of occurrence of a term j in document i. The variable N represents the total number of documents in the static corpus of documents, and n_j represents the number of documents in which term j occurs in that static corpus. For a given frequency f_{ij}, the weight, w_{ij}, increases as the value of n decreases, and vice versa. Terms with a very high weight will have a high frequency f_{ij}, and a low value of n.

For the prototype system described here, a corpus of 258,231 documents from a TREC data collection [29] was used for the ICF table. In the ICF table, we store N, which is the total number of documents in the corpus. Also, for each unique term j, after removing the stop words and applying Porter’s Stemming Algorithm [21], we store n_j, which is the number of documents in the corpus where term j occurred one or more times. As a result, the task of generating a weighted document vector for a document in a dynamic data stream is as simple as one table lookup. The computational complexity of processing N documents is therefore, *O(N)*.

By using the TF-ICF term weighting scheme, the system avoids the problems with TF-IDF as described in [16]. The ICF component of a user’s profile documents (i.e., publications supplied to the system) is compared to a static corpus of news documents rather to each other. This provides a critical benefit: domain specific terms are weighted higher not lower as the TF-IDF scheme would do. This causes the system to be sensitive to different domains, thus enabling its flexibility and use for various domains.

Once a vector representation is created for each document, similarity comparisons can be made. In our approach, a cosine similarity is used to compare two vectors A and B, as shown in (2).

$$\text{Similarity} = (A \cdot B) / (\|A\| \, \|B\|) \qquad (2)$$

Similarity values ranges between 0 and 1, inclusive. A value of 1 means that vectors A and B are identical; while a value of 0 means that they are not alike at all. Recommendations by the DTHSTR system are the documents from each feed that have the highest similarity to the researcher’s publications. The number of documents from the feeds can be adjusted either with a threshold setting for the similarity values, or specifying a fixed number of the most similar documents.

C. *Output*

The DTHSTR system provides the recommendation results via RSS feeds according to categories such as: Call for Proposals, Call for Papers, Inventions, and News. It will also provide a feed of all the results combined into one feed. By providing the results as an RSS feed, this enables the use of any RSS reader program or web component (e.g., as a web part in Microsoft SharePoint) to be used. In addition, this enables the output RSS feed to be used as ingest to another system for further analysis. Finally, this also supports mobile devices.

IV. USE CASE

As an example use case, the authors used one of their own publications entitled "Discovering Potential Precursors of Mammography Abnormalities based on Textual Features, Frequencies, and Sequences" [18]. This publication

ISBN: 978-1-61208-226-4

discusses research related to temporal analysis of mammograms using Haar wavelets. In [18], the Haar wavelet was used for pattern recognition of precursors to breast cancer or other anomalies. This single publication was used an input to the DTHSTR system running on a single desktop system with two 4-core processors. Example results are shown in Tables II through VI. Table II shows a closely related call for funding proposals from the Air Force Medical Support Agency with a total program funding level of nearly $50 million USD. Table III shows a call for papers for a workshop on data mining healthcare management where the topics include pattern recognition in medical images and data, clinical data analysis, and medical diagnosis. Table IV shows a patent that describes "a method for characterizing signal-vector data using automatic feature selection techniques on wavelet-transformed data to enhance the use of pattern recognition techniques for classification purposes". Table V shows a recommendation for a breast cancer related article from the New England Journal of Medicine. Table VI shows a recommendation for an Association for Computing Machinery news article discussing how machine learning can be used to improve patient diagnosis. As can be seen, with little to no extra effort in their workflow, the authors are now aware of news, patents, funding and publication opportunities that are directly related to their work. The feeds are monitored automatically and the results immediately pushed to the researchers. The researchers no longer have to manually go to each site and perform keyword searches or check each site's feed individually.

TABLE II. CALL FOR PROPOSALS RECOMMENDATION EXAMPLE

Title	Air Force Medical Support Agency (AFMSA/SG8) Modernization Directorate Research / Development and Innovations
Common Terms	Patients, detection, research, diagnosis, performance, medical

TABLE III. CALL FOR PAPERS RECOMMENDATION EXAMPLE 1

Title	DMHM 2012: Third Workshop on Data Mining for Healthcare Management
Common terms	Patients, data, patterns, workshop, detection, diagnosis, analysis

TABLE IV. PATENT RECOMMENDATION EXAMPLE

Title	Method and system for analyzing signal-vector data for pattern recognition from first order sensors
Common terms	Wavelets, coefficient, pre-cursors, haar, patterns, detection, temporal, sampling

TABLE V. JOURNAL ARTICLE RECOMMENDATION EXAMPLE

Title	Breast-cancer Adjuvant Therapy with Zoledronic Acid
Common terms	Patients, breast cancer, lymph, abnormality, diagnosis, analysis

TABLE VI. NEWS ARTICLE RECOMMENDATION EXAMPLE

Title	Better Medicine Through Machine Learning
Common terms	Patients, radiologist, abnormality, diagnostic, patterns, radiology

V. FUTURE WORK

Even with the success of the initial prototype system, there are still areas for improvement. One area is to provide a means for researchers to give feedback to the system on the recommendations. One approach to implementing this is with a semi-supervised approach [3] that relies on the graph Laplacian from spectral graph theory [4]. In this case, a graph is constructed that joins together documents that are similar to each other. The graph can be constructed using a nearest neighbor or similar approach based on proximity in the original feature space. This form of learning would require significantly fewer examples to learn, thus reducing the level of effort by the users to train the system.

Another area for future work involves the automation of finding RSS feeds or other sources of information. The authors manually collected and identified over 130 Internet sources (i.e., sites). Once a site was identified, there was some automation to extract the RSS links in order to eventually collect over 9,500 RSS feeds. Unfortunately, this process was tedious and time consuming. A better approach would be to leverage commercial search engines such as RSS Search Hub [23] to automatically search and collect RSS feeds that are producing information that may be of interest.

VI. SUMMARY

The Internet contains an enormous amount of data that streams faster than can be humanly processed and analyzed. In order for researchers to leverage this data, a recommender system was designed and developed called Distribute The Highest Selected Textual Recommendation (DTHSTR). This system helps fill a critical gap that exists in current technology that can enhance a researcher's awareness in their respective field. The system uses a researcher's recent publications to identify relevant information across more than 9,500 RSS feeds from 130 sources. Documents in the system are represented with a vector space model using the term frequency / inverse corpus frequency (TF-ICF) term weighting scheme. A cosine similarity is used to compare a researcher's publications with those document retrieved from the RSS feeds. Most similar documents are presented to the researcher via an RSS feed. The system is currently deployed and used at Oak Ridge National Laboratory and has been shown to be flexible to various domains as well as enable researchers to quickly maintain awareness of relevant information to their work.

ACKNOWLEDGMENT

Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285; managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract DE-AC05-00OR2225. This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 for the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the

published form of this manuscript, or allow others to do so, for United States Government purposes.

REFERENCES

[1] Burkepile, A. and Fizzano, P., "Classifying RSS Feeds with an Artificial Immune System," Second International Conference on Information, Process, and Knowledge Management, pp. 43-47 (2010)

[2] Buckley, C., Singhal, A., and Mitra, M., New retrieval approaches using SMART. In Proc. of the 4th Text Retrieval conference (TREC-4), Gaithersburg (1996)

[3] Chapelle, O., Scholkopf, B., and Zien, A. eds., Semi-Supervised Learning, MIT Press: Cambridge, MA (2006)

[4] Chung, F.R.K., Spectral Graph Theory. American Mathematical Society, Providence, RI (1997)

[5] Garcia, I., and Ng, Y-K., "Eliminating Redundant and Less-Informative RSS News Articles Based on Word Similarity and a Fuzzy Equivalence Relation," 18th IEEE International Conference on Tools with Artificial Intelligence, pp.465-473 (2006)

[6] Gipp, B., Beel, J., and Hentschel, C., "Scienstein: A Research Paper Recommender System," In Proceedings of the International Conferenc on Emerging Trends in Computing, pp. 309-315, (2009)

[7] Hung Chim, and Xiaotie Deng, "Efficient Phrase-Based Document Similarity for Clustering," IEEE Transactions on Knowledge and Data Engineering, vol.20, no.9, pp. 1217-1229, (2008)

[8] Iwasaki, W., Yamamoto, Y., & Takagi, T. (2010). TogoDoc Server/Client System: Smart Recommendation and Efficient Management of Life Science Literature.

[9] Jones, K.S. and Willett, P., Readings in Information Retrieval, Chap. 3. Morgan Kaufmann Publishers, San Francisco, CA, pp. 305-312 (1997)

[10] Kang, J., and Choi, J., "An Ontology-Based Recommendation System Using Long-Term and Short-Term Preferences," 2011 International Conference on Information Science and Applications (ICISA), pp. 1-8, (2011)

[11] Lan, M., Sung, S-Y., Low, H-B, and Tan, C-L., "A comparative study on term weighting schemes for text categorization," In Proc. of the 2005 IEEE International Joint Conference on Neural Networks, vol.1, no., pp. 546- 551, (2005)

[12] Leong, S. (2009). A survey of recommender systems for scientific papers.

[13] McNee, S. M., Kapoor, N., & Konstan, J. A. (2006). Don’t look stupid: avoiding pitfalls when recommending research papers.

[14] Nakagawa, A., and Ito, T., "An implementation of a knowledge recommendation system based on similarity among users' profiles," Proceedings of the 41st SICE Annual Conference, pp. 326- 327, (2002)

[15] Pagonis, J., and Sinclair, M., "Evolving personal agent environments to reduce internet information overload: initial considerations," Proceedings of the IEE Colloquium on Lost in the Web: Navigation on the Internet, (1999)

[16] Pagonis, J., and Clark, A.F., "Engene: A genetic algorithm classifier for content-based recommender systems that does not require continuous user feedback," 2010 UK Workshop on Computational Intelligence (UKCI), pp. 1-6, (2010)

[17] Parra, D. (2009). RecULike: Recommending Scientific Articles on CiteULike using variations of Collaborative Filtering Algorithms.

[18] Patton, R.M., and Potok, T. E., “Discovering Potential Precursors of Mammography Abnormalities based on Textual Features, Frequencies, and Sequences”, 10th International Conference on Artificial Intelligence and Soft Computing, (2010)

[19] Popa, H.-E.; Negru, V.; Pop, D.; Muscalagiu, I., "DL-AgentRecom - A Multi-Agent Based Recommendation System for Scientific Documents," 10th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, pp. 320-324, (2008)

[20] Pinheiro, W.A., de S. Rodrigues, T., da Silva, M.A.R., da Silva, M.A.N., Silva, M.C.O,; Xexeo, G., and de Souza, J.M., "Autonomic RSS: Discarding Irrelevant News," Fifth International Conference on Autonomic and Autonomous Systems, pp.148-153 (2009)

[21] Porter, M.F., An algorithm for suffix stripping. Program, 14(3), pp. 130-137 (1980)

[22] Reed, J.W., Jiao, Y., Potok, T.E., Klump, B.A., Elmore, M.T., and Hurson, A.R., "TF-ICF: A new term weighting scheme for clustering dynamic data streams," In Proc. of the 5th International Conference on Machine Learning and Applications, pp. 258-263 (2006)

[23] RSS Search Hub, current January 2012, http://www.rsssearchhub.com/

[24] Saha, S., Sajjanhar, A., Gao, S., Dew, R., and Zhao, Y., "Delivering Categorized News Items Using RSS Feeds and Web Services," IEEE 10th International Conference on Computer and Information Technology, pp.698-702 (2010)

[25] Salton, G., Wong, A., and Yang, C.S., "A Vector Space Model for Automatic Indexing," Communications of the ACM, 18(11), pp. 613–620, (1975)

[26] Salton, G., and McGill, M.J., Introduction to Modern Information Retrieval, McGraw Hill Book Co., New York, (1983)

[27] Salton, G. and Buckley, C., Term-weighting approaches in automatic text retrieval. Journal of Information Processing and management, 24(5), pp. 513-523, (1988)

[28] Spark-Jones, K., “A statistical interpretation of term specificity and its application in retrieval,” Journal of Documentation, 28(5), pp. 111-121, (1972)

[29] Text Retrieval Conference data, current January 2012, http://trec.nist.gov/data.html

[30] Torres, R., McNee, S. M., Abel, M., Konstan, J. A., & Riedl, J. (2004). Enhancing digital libraries with TechLens.

[31] Wang, C., & Blei, D. M. (2011). Collaborative Topic Modeling for Recommending Scientific Articles.

[32] Xu, H., and Li, C., "A Novel Term Weighting Scheme for Automated Text Categorization," Seventh International Conference on Intelligent Systems Design and Applications, (2007)

[33] Yang, Q., Zhang, S., and Feng, B., "Research on Personalized Recommendation System of Scientific and Technological Periodical Based on Automatic Summarization," First IEEE International Symposium on Information Technologies and Applications in Education (2007)

[34] Yang, S-Y, and Hsu, C-L., "A New Ontology-Supported and Hybrid Recommending Information System for Scholars," 13th International Conference on Network-Based Information Systems, pp. 379-384, (2010)

[35] Zeng, L., Zhang, Y., and Qiu, R. G., "Adaptive User Profiling in Enhancing RSS-based Information Services," IEEE International Conference on Service Operations and Logistics, and Informatics, pp.1-5 (2007)

Enabling Dynamical Use of Integrated Systems and Scientific Supercomputing Resources for Archaeological Information Systems

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 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. 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.**

Keywords–Integrated Systems; Information Systems; Scientific Supercomputing; Computing Systems; Archaeology; Geosciences; High Performance Computing.

I. Introduction

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 dynamical computing. The studies and implementations of Integrated Information and Computing Systems (IICS) have shown a number of queuing aspects and challenges [1], [2]. 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 GEXI collaborations [3] 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 [4], integrating GIS, and parallel computing are promising candidates for future support.

This paper is organised as follows. Section two introduces with the complexity of required information and structure. Section three shows the essential prerequisites for integrated information and computing. Section four describes the basics of Archaeological IICS. Section five discusses the implementation of the components: information sources, structure and classification, communication and computing. Section six presents the system implementation in practice, with various views from the components. Section seven evaluates for the lessons learned and summarises conclusions and future work.

II. 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 [5], [6], [7], [8], [9], [10], [11]. For all components presented, the main information, data, and algorithms are provided by the LX Foundation Scientific Resources [12].

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) [13]. 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.

III. Integrated information and computing

The integration issues of information, communication, and computing are well understood [1], [14], [15] from the “collaboration house” (Figure 1) framework.

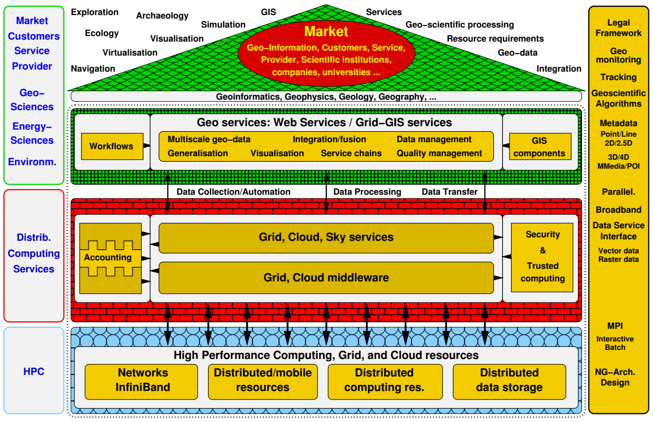


Figure 1. Collaboration house framework, integrating information and scientific computing. Resources (blue), services (red), disciplines (green).

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 [16]. The screenshot (Figure 2) 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.

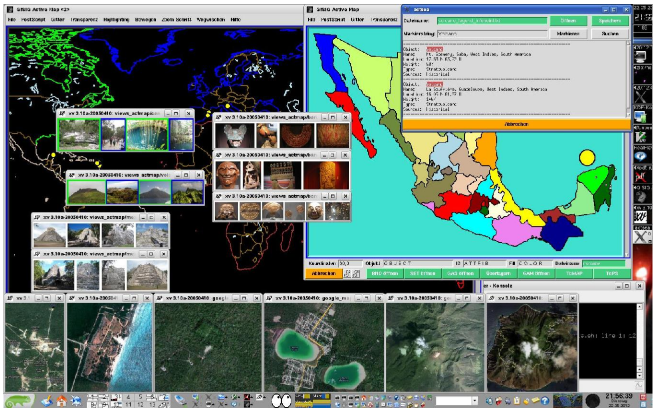


Figure 2. Dynamical use of information systems and scientific computing.

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 discovered without the combination of archaeological information, aerial photographs, satellite images, and digital terrain models [17]. Even in unorganised circumstances, like with this discovery, the multidisciplinary cooperation can lead to success. The more we need an integrated information system approach for "disciplines on demand". On the other hand we have a synergetic effect 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 [18] and it's attractiveness [19].

IV. 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 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 multidisciplinary 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. 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 [20], [21], [22], [23], [24]. 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, 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.

V. Implementation of components

A. Targets and means

The main target categories and means of information to be addressed are interdisciplinary, multidisciplinary, 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 [12]. The related information, all data, and algorithm objects presented are copyright the LX Foundation Scientific Resources [12]. It provides multidisciplinary information and data, 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 [12], [1], [25] and categorised with means like UDC. Listing 1 shows an excerpt of a LX object entry used with IICS. Listing 2 shows a classification set of UDC samples used with IICS.

```
Cenote Sagrada [Geology, Spelaeology, Archaeology]:
               Cenote, Yucatán, México.
               Holy cenote in the area of Chichén Itzá.
               ...
               %%UDC:[55+56+911.2]:[902+903+904]:
                [25+930.85]"63"(7+23+24)=84/=88
```

Listing 1. Structure of object entry (LX Resources, excerpt).

```
UDC:[902+903+904]:[25+930.85]"63"(7)(093)=84/=88
UDC:[902+903+904]:[930.85]"63"(23)(7):(4)=84/=88
UDC:[55+56+911.2]:[902+903+904]:[25+930.85]"63"
 (7+23+24)=84/=88
UDC:[25+930.85]:[902]"63"(7)(093)=84/=88
UDC:[911.2+55+56]:[57+930.85]:[902+903+904]"63"
 (7+23+24)=84/=88
UDC:[911.2+55]:[57+930.85]:[902]"63"(7+23+24)=84/=88
```

Listing 2. Classification set (UDC samples, excerpt).

The classification deployed for documentation [26] 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 3 shows an example of a dynamical dataset from an Active Source [16] 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 3. Dynamical data set of Active Source component.

Batch and interactive features are integrated with Active Source event management [16], e.g., allowing structure and UDC based filtering. 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.

VI. 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,

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

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 a multi-dimensional "potato shapes". 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 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 Soufriére 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 2) as computed and filtered with the IICS, using photo media samples. It must be emphasised that the applications can provide any type of objects, high resolution media, and detailed information. The first view (Figure 3) 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 3. Object SAMPLE – regional Pyramid of Maya, Yucatán, México.

Figure 4 illustrates the computed objects for the above REFERTO-TOPIC and REFERTO-SPACE chain classification, e.g., here via UDC "(7):(4)" relation.

Figure 4. Cross-purpose REFERTO – Diving god, Tulúm, Colom.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

Besides that, viewing directions can be referred, e.g., “view to”, “view from”, “detail” as shown with a VIEW example (Figure 5) for the above selection with UDC “(23)”, “(24)”.

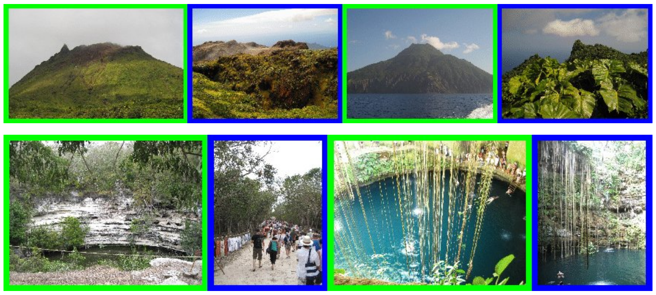


Figure 5. In-purpose: VIEW-TO VIEW-FROM – Volcanoes and Cenotes.

D. Topic view and object representation

The following sample excerpt tabulates a topic view (Table III) and shows the computed object representation (Figure 6) 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

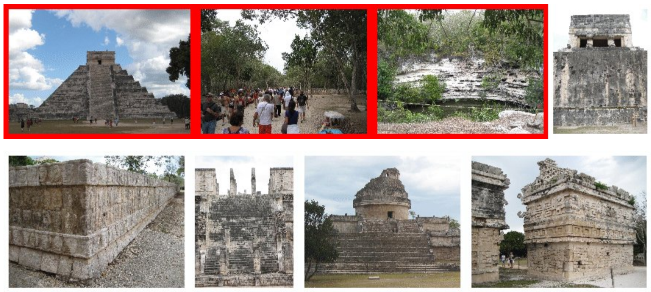


Figure 6. In-topic CONNECT – Kukulkán, Cenote, connected by Sacbé.

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 7), context of amphores (Figure 8), and geology information (Figure 9).

Figure 7. Sample COLLECTION – Precolombian Museum.

Figure 8. Sample CONTEXT – Pottery (amphores).

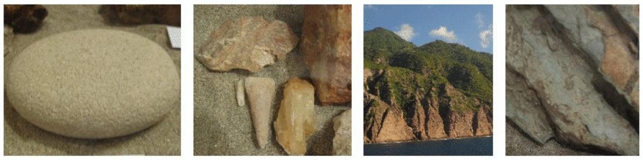


Figure 9. Sample DISCIPLINE – Geology (Caribbean limestone and tuff).

VII. CONCLUSION AND FUTURE WORK

Structuring and classification 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. The concept can be transferred to numerous applications in a very flexible way.

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. 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.

ACKNOWLEDGEMENTS

We are grateful to all national and international partners in the GEXI cooperations for the innovative constructive work and to the scientific colleagues at the Leibniz Universität Hannover, the Institute for Legal Informatics (IRI), and WWU Münster, sharing experiences on ZIV, HLRN, Grid, and Cloud resources and for participating in fruitful case studies as well as the participants of the EULISP Programme 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.), I.N.A.H. México, 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 (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, *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: http://dx.doi.org/10.5772/29337 [accessed: 2012-05-10].

[2] 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: 2012-02-26].

[3] "Geo Exploration and Information (GEXI)," 1996, 1999, 2010, 2012, URL: http://www.user.uni-hannover.de/cpr/x/rprojs/en/index.html#GEXI (Information) [acc.: 2012-02-26].

[4] 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.

[5] N. P. Service, "National Register of Historic Places Official Website, Part of the National Park Service (NPS)," 2012, NPS, URL: http://www.nps.gov/nr [accessed: 2012-03-18].

[6] "North American Database of Archaeological Geophysics (NADAG)," 2012, University of Arkansas, URL: http://www.cast.uark.edu/nadag/ [accessed: 2012-04-08].

[7] "Center for Advanced Spatial Technologies (CAST)," 2012, University of Arkansas, URL: http://www.cast.uark.edu/ [accessed: 2012-04-08].

[8] "Archaeology Data Service (ADS)," 2012, URL: http://archaeologydataservice.ac.uk/ [accessed: 2012-04-08].

[9] "Center for Digital Antiquity," 2012, Arizona State Univ., URL: http://www.digitalantiquity.org/ [acc.: 2012-01-08].

[10] "The Digital Archaeological Record (tDAR)," 2012, URL: http://www.tdar.org [accessed: 2012-01-08].

[11] 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].

[12] "LX-Project," 2012, URL: http://www.user.uni-hannover.de/cpr/x/rprojs/en/#LX (Information) [accessed: 2012-02-26].

[13] "Universal Decimal Classification Consortium (UDCC)," 2012, URL: http://www.udcc.org [accessed: 2012-02-19].

[14] C.-P. Rückemann, "Dynamical Parallel Applications on Distributed and HPC Systems," *International Journal on Advances in Software*, vol. 2, no. 2, 2009, ISSN: 1942-2628.

[15] C.-P. Rückemann, "Legal Issues Regarding Distributed and High Performance Computing in Geosciences and Exploration," in *Proceedings of the Int. Conf. on Digital Society (ICDS 2010 / CYBERLAWS 2010), February 10–16, 2010, St. Maarten, Netherlands Antilles*. IEEE CSP & 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: 2011-02-20].

[16] C.-P. Rückemann, "Beitrag zur Realisierung portabler Komponenten für Geoinformationssysteme. Ein Konzept zur ereignisgesteuerten und dynamischen Visualisierung und Aufbereitung geowissenschaftlicher Daten," Diss., WWU, Münster, Deutschland, 2001, 161 (xxii+139) S., 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: 2012-01-15].

[17] 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: 2012-03-25].

[18] 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: 2012-03-25].

[19] "Venice Mobility Project - Pedestrian Modeling," *Santa Fe Complex*, 2012, February, 2012, URL: http://sfcomplex.org/2012/02/venice-mobility-project-pedestrian-modeling [accessed: 2012-03-31].

[20] K. Perrin, "Archaeological Archives: Documentation, Access and Deposition. A Way Forward," *English Heritage*, 2002.

[21] 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/ [acc.: 2012-04-08].

[22] "Guides to Good Practice," 2012, ADS, URL: http://guides.archaeologydataservice.ac.uk/ [accessed: 2012-04-08].

[23] 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: 2012-03-25].

[24] "Archaeological Archives Forum (AAF)," 2012, URL: http://www.britarch.ac.uk/archives/ [accessed: 2012-04-08].

[25] C.-P. Rückemann, *Integrated Information and Computing Systems for Advanced Cognition with Natural Sciences*. IGI Global, Hershey, Pennsylvania, USA, 2012, in: Rückemann, C.-P. (ed.), ISBN: 978-1-4666-2190-9, DOI: 10.4018/978-1-4666-2190-9, (to appear).

[26] C.-P. Rückemann, "Integrating Information Systems and Scientific Computing," *Int. Journal on Advances in Systems and Measurements*, 2012, ISSN: 1942-261x, (to appear).

Nonlinear Transformation's Impact Factor of Cryptography at Confusion and Cluster Process

Lan Luo[1]
[1]*Networks and Intelligent Application of Block Cipher lab*
University of Electronic Science Technology of China,
Chengdu, China luolan@uestc.edu.cn

Tao Lu[1,4]
[4]Department of Civil and Structural Engineering,
School of Engineering, Aalto University, Aalto,
Espoo Finland tao.lu@aalto.fi

Zehui Qu[1,2]
[2]*Dipartimento di Informatica*
Università di Pisa, Pisa, Italy
zehui.qu@gmail.com

Qionghai Dai[1,5]
[5]*School of Information Science & Technology*
Automation Department, Tsinghua University, China
qhdai@tsinghua.edu.cn

Yalan Ye[1,3]
[3]*School of Computer Science and Technology*
University of Electronic Science Technology of China,
Chengdu, China yalanye@uestc.edu.cn

***Abstract*—For investigating the nonlinear character at the confusion and cluster process of cryptography, the nonlinear transformation's impact factor has been introduced. According to sorting result by different years, the amount of online cryptography is accounted. And then the nonlinear indexes, which are related to the outcome of amount reflecting the nonlinear character directly, are researched at the confusion process and the cluster process respectively. The nonlinear transformation's impact factor which includes both the private-key cryptography and public-key cryptography is studied with known nonlinear index at naïve Bayesian model, which combines with the networks environment fused in protocol whenever confusion or cluster process. To any networks environment, higher the nonlinear transformation's impact factor is, more popular the used cryptography is because more amounts of kinds of level users requiring stronger secure cryptography. So, the impact factor of the nonlinear transformation is a kind of cryptography's label indicating the suitable to application environment by suitable crowd. Contrarily, the extent of secure can measure up the nonlinear character of cryptography precisely.**

Keywords-nonlinear transformation; confusion and cluster; impact factor; Bayesian model

I. INTRODUCTION

Nonlinear transformation (NT) [1][2][3][4][5], which exists at any kinds of secure communication systems, is an important part of cryptographic study. The cryptography includes private key and public key cryptography. The private key cryptography is a secure process which uses one secret key to confuse and recover the information, and the private key process system is depend on a pseudorandom number generator, such as HASH [6][7][8], the block cipher[9][10][11][12], the stream cipher[13][14][15]. The public key cryptography is a process which is used two keys to ensure the information, and it is based on a mathematic problem, such as integer factorization and Elliptic curve [16][17][18][19]. Whether private or public key, the NT is evolved [20][21][22] from value tables supported by development of the information technology[23]. Since the Future Internet [24][25][26][27][28], wireless sensor[29][30], RFID[31][32] and quantum communication [33][34][35] are developed, the application of block cipher, stream cipher and HASH[36] is prevalence to ensure the communication's secure. The RSA [37], ECC[38] are used at mini-information secure condition, such as digital signature, key exchange schemes. The Lattice-based cryptosystem is simply introduced at the paper. The NT Impact factor (IF) of cryptography is a simple and popular way to value the NT influence at Bayesian model [39][40][41].

This paper focuses on the nonlinear transformation impact factor of cryptography, which is clustering with the different time, different scales of nonlinear transformation and different application environment. The rest of the paper is organized as follows. Section 2 contains the two inversed study directions which are confusion process and cluster process at cryptography. The nonlinear indexes and NT IF according to Bayesian model are described in section 3. Section 4 includes conclusion about the effect of NT IF.

II. CONFUSION AND CLUSTER PROCESS AT CRYPTO

The confusion and cluster process are the certain parts of encipher and decipher of information at cryptography. The whole cryptographic system is a nonlinear process because of the sharing of NT or nonlinear mathematic problem. The confusion process is the nonlinear part besides the key addition and linear diffusion process at secret key condition, and it is the whole operation process at the public key condition. Meanwhile the cluster process is the first step which is a simple differential sort process at whenever the secret key cryptographic system's condition of white box, gray box or black box, and is the exponent of the public key cryptographic. It shows the sort of online cryptography according to the change of time at figure 1. The published online cryptography amount has a jump as figure 2.

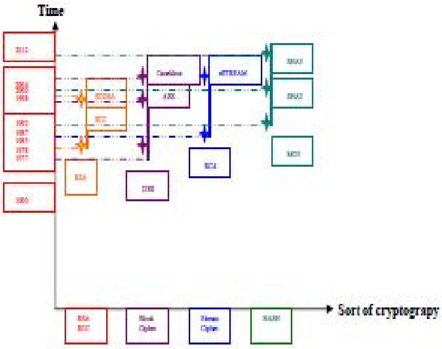


Figure 1. The sort of cryptography online according to time

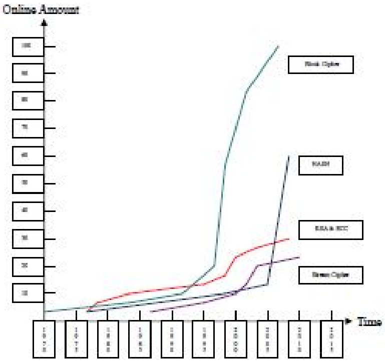


Figure 2. Online Cryptography Amount by Years

A. *Confusion and Cluster Process Precisely Depicted*

The confusion processes and cluster processes of cryptography sort out to 5 categories, which are confusion of block cipher, of stream cipher, of HASH, of RSA and of ECC in this paper. The confusion process of block cipher has a NT, which is from N bits to M bits. At the beginning of the published block cipher, the confusion process is constructed by 6 bites to 4 bits NT, which is activated 4 to 8 times and repeated 16 rounds, such as DES in 1976. IDEA is also an old block cipher which is 16 bits to 16 bites NT repeated 8.5 rounds in 1991. Updated block cipher is constructed by 8 bits to 8 bits NT activated 8 to 16 times and repeated 10, 12 or 14 rounds, such as AES in 1998. Camellia is a new block cipher which is the same nonlinear part similar as AES repeated 18 rounds in 2000. The stream cipher has none any nonlinear part at first, such as A5 used in the GSM cellular telephone standard. From November 2004 to April 2008, the project founded by EU ECRYPT network founded, eSTREAM, identify new stream ciphers suitable for widespread adoption. Salsa20, which is based on software, is one of the winners. The NT of Salsa20 is 128 bits to 128 bits repeated 20 rounds. Trivium based on hardware is one of the winners. Its NT is 228 bits to 228 bits at 4th rounds. Those are some nonlinear emblematical transformations.

The old HASH SHA1 is published at 1995, which is 96 bits to 96 bits NT repeated 80 rounds. SHA2 has a NT which is 96 bits to 96 bits activated 2 or 4 times repeated 64 or 80 rounds. A new SHA3 competition is an open competition from October 31, 2008 to the end of 2012, and the final round candidates have occurred on December 10, 2010. Keccak of finalists has a NT, which is 192 bits of 64 bit memory to 64 bits repeated 24 rounds. Skein of finalist has a NT, the designers call it MIX, which is 128 bits to 128 bits repeated 72 rounds or 256 bits to 256 bits repeated 80 rounds.

RSA and ECC are another typical type of cryptography comparing to block cipher, stream cipher and HASH, being called the public key cryptography. While the RSA patent expired in 2000, the NT of RSA is the naïve bit amount of prime number at ANSI X9.31 in 1998, which is 1024 bits. Meanwhile, the NT of ECC is the 192, 224, 256, 384, and 521 bits according to FIPS 186-3 in 2009. The Lattice based cryptography is a new mathematic problem appears with the post-quantum cryptography. The "ideal lattice" designed by Craig Gentry, which is announced by IBM at 6.2009, NL is depicted by the Lattice problem in n-dimensional Euclidean space R^n with a strong periodicity property.

B. *Nonlinear Transformations at a Confusion process*

The confusion process is to confuse the information to pseudo-random data which cannot be reversed at the useful-life time of the information. The different cryptography uses different nonlinear transformation to obtain the random. The block ciphers, being the popularity from public game AES, have active or non-active model at NT. If the box or the value table implement at NT part, the active number of box affects the nonlinear complexity and speed directly. More active boxes, more complexity the NT is. The mathematic problem NT depends on the exponent of the nonlinear function. There are some distinctions, which are the speed and secure influences, between all boxes activated and part of boxes activated at confusion process. There are some useful nonlinear factors at sub-key rolling in process. The information block and nonlinear complexity of sub-key are covered up by NT because they are far lower exponent than NT's. And then, the effect of NT is diffused with a linear array. By the way, a block cipher is constituted by iterative rounds which include information block, sub-key addition, NT and diffusion. The stream ciphers have a high speed character, which is usually suitable to the secure of wireless communication and a high-capacity bandwidth environment. The stream ciphers usually constitute by linear feedback shift register (LFSR) or nonlinear feedback shift register (NFSR) fusing by NT. If the NFSR is the part of a stream cipher, the resilience function, which has both better balance and better anti-differential analysis, is a better choice than bent function. The NT part of stream cipher can be just similar to a whole rounds or mini rounds of block cipher. And more and more simple NT fused directly to LFSR or NFSR is appearing, such as Trivium, Grain, or NUSH. Information is confused by a simple NT of stream cipher has lower secure but higher speed. HASH is a stream cipher or a block cipher adding kinds of digest function. The digest function is considered as a linear part so that the NT of HASH process refers to the stream or block cipher. So the NT of HASH is the same as the stream or block cipher.

The confusion process of public key cryptography is to ensure the mini-scale communication key's security. The mathematic confusion process is nonlinear because exponent is more than 1. The NT of RSA is up to 210 in 2010. Then the successor ECC has a popular confusion process at the equation $y^2 = x^3+ax+b$. So the NT of ECC is considered as 3^2. Until now, the idea model of the Lattice based confusion process is a no known bounds iteration of vectors at least 2 equivalence dimensions. The NT of lattice based confusion process is $(\text{number at least } 2)^n$, the n is unknown.

C. Nonlinear Transformations at a Recovery Process

A recovery process is to unveil the cipher step by step at the condition of White-box, grey-box and black-box. If the confusion process is totally published, the recovery process is at a white-box situation. If there only are confusion date or some parts of confusion process, the recovery process is at a black-box or a grey-box situation. The NT at a recovery process based on a white-box has to operate carefully rounds by rounds, or just operates the reversed process of the mathematic problem. At this situation, the NT is strictly equal to the reverse of NT at confusion process. According the grey-box and black-box conditions, the cluster process is necessary to congregate the confusion date. By the NT of cluster process, the data is sorted by some factors, such as the linear index, the nonlinear index, characters of Pseudo-Randomness or avalanche, including by other mathematic ways and means. Then those sorts are the results of first glimpse of the recovery process, in which the grey-box is covered to the white-box and the black-box to grey-box. The black-box is equivalent to several grey-boxes which also are considered as the scale of white-box based on the results. When the cluster data has a certain sameness with a certain white-box, the grey-box can be equal to the white-box at the scale. In conclusion, the black boxes mix with white boxes according to results of NF cluster at recovery process can be converted to grey boxes, which are prepared to be white.

So, the cluster process of NT can recovery white-box completely and converse the deep color box to undertone box in basis of confusion date clustering.

III. Nonlinear Indexes Based on Bayesian Model

Bayesian inference is a rational engine for solving such problems within a probabilistic framework, and consequently is the heart of most probabilistic models of nonlinear indexes the cryptography. The nonlinear indexes, which include the private-key cryptography and public-key cryptography are sorted at naïve Bayesian model, whenever it's confusion or cluster process. The simple NT index reflect the nonlinear character itself meanwhile the NT IF combine the networks or cryptography protocols into NT index.

A. Nonlinear transformations index at cryptography

The nonlinear transformations index at cryptography is a kind of rough description about the strength of cryptography. The NT index of block cipher is the product of the active boxes number of a block and exponent of nonlinear function. For example, the NT index of standard AES which has 16 actives s-boxes at each of 10 rounds is that product of 8 and 160. The NT index of stream cipher is the sum of each part of nonlinear functions' exponent. The stream cipher Grain has a NT index which is the product of 3 and 6. The NT index of HASH evolved from the certain cipher, which is block cipher or stream cipher, is the same count method as the certain cipher. The NT index of public-key cryptography, such as RSA or ECC, is that 1024 or 9 referring to the standard. The ECC has the homothetic nonlinear character as private-key cryptography.

If only consider the NT index, the block cipher is equal to many times of stream cipher, such as NT index of AES is about 72 times of Grain. Meanwhile, RSA is almost equal to 120 times of ECC. Thus, standard AES can be secure the 72 times bandwidth data the same as Grain. The signature with RSA is equal to 72 signatures of ECC. The figure3 shows the NT IF by years at different networks environment.

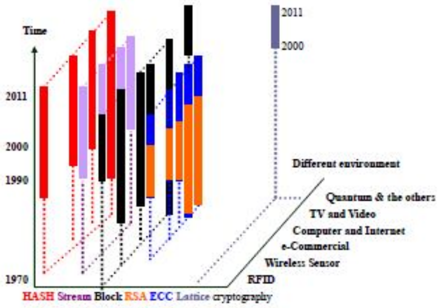


Figure 3. NT IF at different environments according to time

B. The Nonlinear Transformations Impact Factor Based on Naïve Bayesian Model

The NT impact factor is related to NT index according to the application of cryptography at the naïve Bayesian model. The NT index reflects the nonlinear character directly and NT IF combines with the networks environment fused in protocol. So T_0 is the NT index which is the cryptography's nonlinear measure and the T_1 is that is the different networks. Naive Bayesian networks are identified with theories at the lowest, most concrete level of the abstraction hierarchy, level T_0. The categorization grammars are typically identified with the T_1-level theories that define hypothesis spaces of T_0-level structures and assign prior probabilities to those hypotheses, thereby guiding inferences about the naive network structure T_0 mostly likely to have given rise to some observed dataset d. A Bayesian learner evaluates a naive network hypothesis T_0 based on its posterior probability:

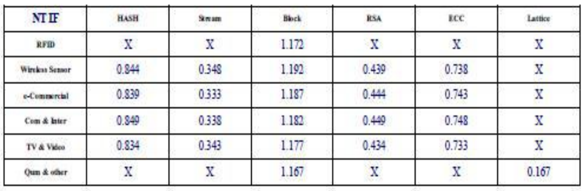
$$\boldsymbol{P\ (T_0|d,T_1) = （P(d|T_0)P(T_0|T_1)）/P(d|T_1)} \quad (1)$$

where the denominator is

$$\boldsymbol{P\ (d|T_1) =} \sum \boldsymbol{P(d|T_0)P(T_0|T_1)} \quad (2)$$

The naive grammar T_1 specifies a probabilistic process for generating naive-network hypotheses. With such a Bayesian model, the NT IF of cryptography is discussed in a causal way which is considered as naive condition. NT IF is the iteration of Bayesian model result and the amount of the cryptography. There is one kind of NT based on ciphers' application among different networks which occurs between private-key cryptography and public-key cryptography. So the NT IF is identical as the cryptography amount online. The table1 is the NT IF result considered the impact by the frequency of same networks environment.

TABLE I. NT IF OF CRYPTOGRAPHY AT DIFFERENT ENVIRONMENT IN 2011 ACCORDING TO BAYESIAN MODEL



NT IF	HASH	Stream	Block	RSA	ECC	Lattice
RFID	X	X	1.172	X	X	X
Wireless Sensor	0.844	0.348	1.192	0.439	0.738	X
e-Commercial	0.839	0.333	1.187	0.444	0.743	X
Com & Inter	0.849	0.338	1.182	0.449	0.748	X
TV & Video	0.834	0.343	1.177	0.434	0.733	X
Qum & other	X	X	1.167	X	X	0.167

IV. CONCLUSION

The nonlinear character of both private-key cryptography and public-key cryptography is expressed by the nonlinear transformation's impact factor simply. At confusion process, the NT IF is a complexity of nonlinear character. At cluster process, the NT IF is the appearance color of nonlinear box or the known degree of a cryptography algorithm. At any networks, higher the NT IF is, the more popularity the cryptography is used. The reason is that more amounts of users require stronger secure level. Furthermore, the NT's IF of cryptography at confusion and cluster process based on Bayesian model, which is an advanced description of the nonlinear character according to the different application environments, demonstrates that higher NT IF is a suitable wider width-band data communication. The NT IF can be a label of a cryptography system indicating the suitable crowd and suitable application environment. Contrarily, the extent of cryptography's secure can be measure up the NT impact factor's precision deeply.

ACKNOWLEDGMENT

This research is partly financially supported by the zygx2010J068. We would like to deeply thank the various people who provided us with very useful and helpful suggestions. The authors sincerely thanks for YuQi He, the academician of Academy of Sciences of the United States and Department of Automation Tsinghua University and Harvard University, who is our tutor of the Bayesian model at Sciencenet.CN.

REFERENCES

[1] Svetla Nikova, Vincent Rijmen and Martin Schläffer, Nonlinear transformations S-box Noekeon the other design of hardware Secure Hardware Implementation of Nonlinear Transformations in the Presence of Glitches, Journal of Cryptology, Vol. 24, Number 2, April 2011, Pages 292-321

[2] D. R. Stinson and J. L. Massey, An infinite class of counterexamples to a conjecture concerning nonlinear resilient transformations. Journal of Cryptology, 1995, Vol. 8, Number 3, Pages 167-173

[3] Stanislav V. Smyshlyaev, Perfectly Balanced Boolean Transformations and Golić Conjecture, Journal of Cryptology, 3 July 2012, Pages 464-483

[4] Carlisle Adams and Stafford Tavares, The structured design of cryptographically good s-boxes, Journal of Cryptology, 1990, Volume 3, Number 1, Pages 27-41

[5] Nenad Dedić, Gene Itkis, Leonid Reyzin and Scott Russell, Upper and Lower Bounds on Black-Box Steganography, Journal of Cryptology, 2009, Vol. 22, Num 3, Pages 365-394

[6] Lars R. Knudsen, Xuejia Lai and Bart Preneel, Attacks on Fast Double Block Length Hash Transformations, Journal of Cryptology, 1998, Volume 11, Number 1, Pages 59-72

[7] ebastiaan Indesteege and Bart Preneel, Practical Collisions for EnRUPT. Journal of Cryptology, 2011, Volume 24, Number 1, Pages 1-23

[8] David Cash, Dennis Hofheinz, Eike Kiltz and Chris Peikert, Bonsai Trees, or How to Delegate a Lattice Basis. Eurocrypt'10 Proceedings of the 29th Annual international conference on Theory and Applications of Cryptographic Techniques Pages 523-552

[9] Debra L. Cook, Moti Yung and Angelos D. Keromytis, Elastic block ciphers: method, security and instantiations, International Journal of Information Security, 2009, Volume 8, Number 3, Pages 211-231.

[10] Joan Daemen and Vincent Rijmen AES proposal[R]: Rijndeal, NIST, FIPS PUB 197, 11. 2001

[11] Matsui, M., Nakajima, J., and S. Moriai, A Description of the Camellia Confuseion Algorithm, RFC 3713, April 2004.

[12] Serge Vaudenay, Decorrelation: A Theory for Block Cipher Security, Journal of Cryptology, 2003, Volume 16, Number 4, Pages 249-286

[13] T.W. Cusick, C. Ding, A. Renvall, Stream Ciphers and Number Theory, Published: APR-1998, ISBN 10: 0-444-82873-7, Imprint: NORTH-HOLLAND

[14] Matthew Robshaw and Olivier Billet, New Stream Cipher Designs The eSTREAM Finalists, Lecture Notes in Computer Science, Volume 4986, 2008

[15] Daniel J. Bernstein, The Salsa20 Family of Stream Ciphers, Lecture Notes in Computer Science, 2008, Volume 4986, Pages 84-97

[16] Steven D. Galbraith, Xibin Lin and Michael Scott, Endomorphisms for Faster Elliptic Curve Cryptography on a Large Class of Curves, Journal of Cryptology, Volume 24, Number 3 / July 2011, Pages 446-469

[17] Johan Håstad and Mats Näslund, Practical Construction and Analysis of Pseudo-Randomness Primitives, Journal of Cryptology, 2008, Volume 21, Number 1, Pages 1-26

[18] S. Micali and C. P. Schnorr, Efficient, perfect polynomial random number generators, Journal of Cryptology, 1990, Volume 3, Number 3, Pages 157-172

[19] Omer Barkol, Yuval Ishai and Enav Weinreb, On d-Multiplicative Secret Sharing, Journal of Cryptology, 2010, Volume 23, Number 4, Pages 580-593

[20] Eran Tromer, Dag Arne Osvik and Adi Shamir, Efficient Cache Attacks on AES, and Countermeasures, Journal of Cryptology, 2010, Volume 23, Number 1, Pages 37-71

[21] M. Bellare, A. Boldyreva, L. Knudsen and C. Namprempre, On-line Ciphers and the Hash-CBC Constructions, CRYPTO '01 Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology, Pages 292 - 309

[22] David M. Goldschlag, Stuart G. Stubblebine and Paul F. Syverson, Temporarily hidden bit commitment and lottery applications, International Journal of Information Security, 2010, Volume 9, Number 1, Pages 33-50

[23] Dibyendu Chakrabarti, Subhamoy Maitra and Bimal Roy, A key pre-distribution scheme for wireless sensor networks: merging blocks in combinatorial design, International Journal of Information Security, 2006, Volume 5, Pages 105-114

[24] Renbin Xiao, Tinggui Chen and Chunhua Ju, Research on Product Development Iterations Based on Feedback Control Theory in a Dynamic Environment, International Journal of Innovative Computing, Information and Control, Volume 7, Number 5(B), May 2011, Pages. 2669-2688

[25] Serap Ataya, Marcelo Maserab, Challenges for the security analysis of Next Generation Networks, Information Security Technical Report, Vol. 16, Issue 1, February 2011, Pages 3-11

[26] William Walker, Mobile telephony security compromises, Information Security Technical Report, Volume 15, Issue 3, August 2010, Pages 134-136

[27] Allan Tomlinson Corresponding Author Contact Information, Po-Wah Yau, John A. MacDonald, Privacy threats in a mobile enterprise social network, Information Security Technical Report, Volume 15, Issue 2, May 2010, Pages 57-66

[28] Roman, R.E, Alcaraz, C., Lopez, J. The role of Wireless Sensor Networks in the area of Critical Information Infrastructure Protection, Information Security Technical Report, Volume 12, Issue 1, 2007, Pages 24-31

[29] Svendsen, N.K.a , Wolthusen, S.D. Connectivity models of interdependency in mixed-type critical infrastructure networks, Information Security Technical Report, Volume 12, Issue 1, 2007, Pages 44-55

[30] Igure, V.M., Laughter, S.A., Williams, R.D. Security issues in SCADA networks, Computers and Security, Volume 25, Issue 7, October 2006, Pages 498-506

[31] Arne Tauber, A survey of certified mail systems provided on the Internet, Computers & Security, Volume 30, Issues 6-7, September-October 2011, Pages 464-485

[32] Roberts, C.M, Radio frequency identification (RFID), Computers and Security, Volume 25, Issue 1, February 2006, Pages 18-26

[33] Alhazmi, O.H., Malaiya, Y.K., Ray, I. Measuring, analyzing and predicting security vulnerabilities in software systems, Computers and Security, Volume 26, Issue 3, May 2007, Pages 219-228

[34] Miron Abramovici, A solution for on-line trust validation, Anaheim, CA, USA, June 09-June 09, ISBN: 978-1-4244-2401-6, 2008 IEEE International Workshop on Hardware-Oriented Security and Trust

[35] Jens-Peter Kaps, Gunnar Gaubatz, Berk Sunar, Cryptography on a Speck of Dust, Computer, vol. 40, no. 2, Feb. 2007, doi:10.1109/MC.2007.52, Pages 38-44

[36] Philip O'Kane, Sakir Sezer, Kieran McLaughlin, "Obfuscation: The Hidden Malware," IEEE Security and Privacy, vol. 9, no. 5, Sep./Oct. 2011, Pages 41-47

[37] Romain Giot ,Mohamad El-Abed, Baptiste Hemery, Christophe Rosenberger, Unconstrained keystroke dynamics authentication with shared secret, Computers & Security, Vol. 30, Issues 6-7, September-October 2011, Pages 427-445

[38] Gary S.-W. Yeo and Raphael C.-W. Phan, On the security of the WinRAR confuseion feature, International Journal of Information Security, 2006, Volume 5, Pages 115-123C

[39] Yuanyuan Wang, Yunming Ye, Xutao Li, Michael K. Ng and Joshua Huang, Hierarchical Information-Theoretic Co-Clustering for High Dimensional Data, International Journal of Innovative Computing, Information and Control, Volume 7, Number 1, January 2011, ISSN 1349-418X, Pages 487-500

[40] Vladimir S. Udaltsov, Jean-Pierre Goedgebuer ,Laurent Larger, Jean-Baptiste Cuenot, Pascal Levy, William T. Rhodes , Cracking chaos-based confuseion systems ruled by nonlinear time delay differential equations, Physics Letters A 308 Pages 54–60

[41] Massimiliano Zanin1, Alexander N. Pisarchik, Boolean Networks for Cryptography and Secure Communication, Nonlinear Sci. Lett. B, Vol. 1, No.1, 2011, Pages 25-32

Decoupling Modeling from Complexity Based on a Freedom-to-Act Architecture

Udo Inden
Cologne University of Applied Sciences
Research Centre for Applications of Intelligent Systems (CAIS),
Cologne, Germany
E-Mail: udo.inden@fh-koeln.de

Sergej Naimark
Controlling Chaos Technologies GmbH
Hannover, Germany
E-Mail:
s.naimark@controlchaostech.de

Claus-Peter Rückemann
Leibniz Universität Hannover / Westfälische Wilhelms-Universität Münster (WWU) / North-German Supercomputing Alliance (HLRN),
Münster, Hannover, Germany
E-mail: ruckema@uni-muenster.de

Abstract—**Focusing on business systems we discuss that self-organizing, increasing operations' complexity is escaping from capabilities of modeling. This urges to decouple modeling from emergent domain complexity as well as management to eventually overcome control illusions inherent to conventional approaches. In a first step; we show that, on some conditions, agent-based modeling and multi-agent architectures are able of solving the problem. Conditions include aspects of service-based modeling, of semantic modeling and of freedom-based agent's behavior. The concepts are relevant for design, management and computing of highly complex operations.**

Keywords-complexity of modeling; agent-based modeling; freedom-to-act architecture; service-oriented architecture; semantic modeling.

I. Introduction to the Problem

Models reduce complexity of a real-world domain with regard to a particular purpose of control. Concepts in this paper are based on experience in highly complex environments, where struggling modeling complexity is part of daily business in strategic, tactical or real-time management. The development of operations complexity shows that improvements of modeling techniques did not ease this job because both, the complexity of operations and of modeling are coupled self-organizing developments.

From experiment, we learned that agent-based modeling (ABM) [37] and multi-agent technology (MAT) [29] [30] offer options of decoupling modeling from domain complexity. We argue that it is necessary if control about complex systems is to be maintained. ABM or MAT are well known. In spite of their advantages in handling complexity, they failed joining the mainstream of Information and Communication Technologies.

The reason is that managements and software developers consider emergent behavior to be an intimate enemy of the control of systems. In contrary, we argue that emergent behavior is the last resort of control of complex systems. Namely, we suggest changing the view at the control of complex systems which by principle cannot be reduced to models. The term control illusion [33] illustrates the difference between the views.

The following examples from aviation industry provide insight into the complexity operations' systems:

Airlines plan the service of aircrafts as sequences of flights executed in a period of time. The cost-efficiency of these '*rotations*' and the service quality they deliver to customers depend on a manifold of interacting factors spanning air- and ground operations and involving thousands of aircrafts and flights of hundreds of airlines or numerous supporting services. Well established international proceedings, synchronize flight plans global networks. Before a flight this abstract plan is to be particularized and confirmed.

But in execution, plans are troubled by a constant floor of interference which easily can get out of control [1]. The challenge lies in the continuous process of adapting to reality by correcting, mitigating or recovering active plans. In large cases ten thousands of autonomous actors and legacy systems are to be re-synchronized in almost real-time – repeatedly because the solution of the next problem may affect the solution of previous ones.

The 'time-to-volume' of industrial series production begins with the start of development of products and production processes and ends with reaching stable output as planned for amortization of invest. But, there are trade-offs, e.g., more engineering time drives costs and postpones the product launch, while less effort drives quality risk emerging from *butterfly effects* (non-linear behavior) or from *black swans* (long-tail risks).

In the cases of B787 or A380 such effects delayed ramp-up and planned volumes for years [2] [3]. For ramp-up, Boeing implemented a *virtual ramp-up system* (VRS, a simulator and planner) covering major stakeholders and components in the supply-chain – except the *fasteners* for carbon-fiber parts. Apparently, these tiny parts were ignored to limit model complexity. The supplier failed and fasteners became a problem. As example of *black swans* [5] the uncontained engine failure of a Trent 900 engine in Quantas A380, flight QF32, November 4th 2010 [4], may serve.

In airliner as in airframer business problems increase: Air traffic, thus the number of aircrafts and flights, is expected to more than double by 2020 while in America or Europe air- and ground infrastructures are lacking behind [34]. Airlines' competition gets harder and new competitors of Boeing and Airbus appear in emerging markets. Solutions are asked to increase service capabilities, reduce costs or CO_2 footprints, and get to volume faster. These examples suggest that architectures need to support, respectively, adapt to

- the pace of change, driven by competition that defines opportunity windows to adapt to change or affects trade-offs between costs and quality of modeling.
- the need of catching up with ignored or not captured aspects in terms of butterflies or black swans.
- the fact that more and more details matter (resolution of object) and become source or target of events (resolution of time: more events per unit of time) [6].

These aspects refer to the *law of requisite variety* saying that control fails if controllers' complexity does not match the complexity of the system to be controlled [15]. In result of developments sketched above models tend to fail delivering their core service that is reducing complexity with regard to a particular purpose. In consequence complexity of modeling is to be decoupled from complexity of domains.

This paper is organized as follows: Section two shows how mainstream architectures relate to complexity and its increase. In Section three, basic principles of the alternative freedom-to-act architecture are analyzed. After a summary, Section fife sketches selected aspects of further work.

II. Complexity in Function-, Process-, and Service-oriented Modeling Architectures

In the mainstream, different regimes of operations management and modeling developed: function-, process-, and service-orientation (with internet-based automation). Coming from its history in knowledge management also semantic modeling is about to enter the mainstream.

A. *Encapsulating Complexity in Organizational Silos: The Function-oriented Regime*

Modeling as effort and tool of operations management is initially bound to the work of Taylor [7] and operations strategies of Ford [8]. This thinking uses functional specialization to grow expertise of engineers or administrators (knowledge workers) and to compensate the lack of educated workers for the assembly lines by extreme simplification.

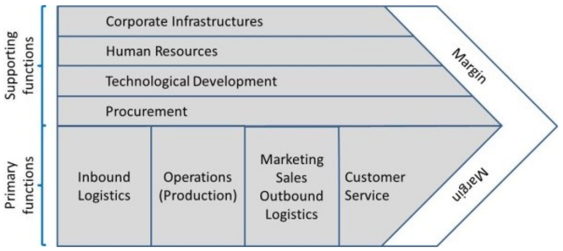


Figure 1. The Value Chain.

Figure 1 depicts a model of the '*value chain*', a later concept introduced by Porter [9]. It shows that value is produced by specialized departments differentiated in those directly contributing to the creation of value (lower group) and supporting ones related to a firm's infrastructures (facilities…, later also IT) or administration (upper group).

The picture also has a financial interpretation: its expanse represents the financial flow passing the organization. Reading the grey block as costs spent for direct and indirect operations, hopefully a positive difference to revenue is left, a profit: the spike of the arrow. In this light function-orientation obviously creates self-referential silo-behavior: if resources are to be distributed (more staff) or costs to be cut (the contrary) routinely competition appears: Who is more important, who to be blamed for failure? Also, careers are built on affiliation to silos. Accordingly, the value creating process is marked by silo-driven discontinuity turning into high coordination effort and long processing-times.

B. *Tackling the Complexity of Interactions The Process-oriented Regime*

In functional models, business processes are implicit. In order to overcome disadvantages they had to be made explicit. The motivation was induced by competition, again starting from car industry: When Taiichi Ohno, CTO of Toyota, visited Ford, he learned that silos or radical simplification of assembly jobs produce problems rather than solutions: High simplification turns into a waste of talent and silos into self-inflicted complexity [10]. Particularly, they obstruct the view at real challenges: effectiveness (value delivered) and efficiency (costs) of operations.

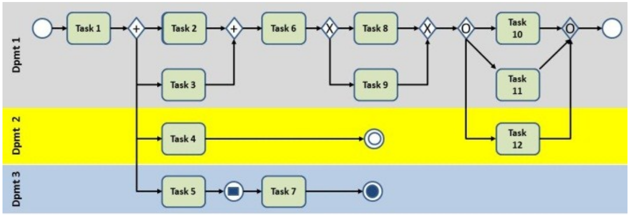


Figure 2. Business Process Modeling Notation.

Instead, Ohno invented strategies avoiding redundancy (*muda*) like *Just in Time* delivery replacing inventories or continuous improvement (*kaizen*) exploiting knowledge at *all* workplaces. Later these ideas were united as *Lean Management*. They aim at unobstructed flows of orders, material, or information as well as purposeful collaboration across fields of tasks and networks of suppliers. Workers, formerly just repeating simple jobs, became autonomous and creative partakers in implementing and improving operations. Knowledge and intelligence became strategic resources.

An MIT study [10], uncovering advantages of Lean Management, became a wake-up call to established car makers. Finally, the new strategy changed the rules of competition and formed a new *fitness landscape* [27]. Initially, managers in the USA or Europe misunderstood the call to become lean as a call for cutting cost. But, in fact, it was a call to change minds towards integrative thinking and modeling. Rather than optimizing functionality in the silos, the new heading geared towards the efficiency of functions linked across business processes and improving value propositions – a far more complex job than managing silos.

Organizations had to learn how to create new value from this. Self-inflicted complexity was to be exchanged by value-driven (money-making) complexity. It asked taking care for interdependencies beyond boxes and to enrich responsibility of knowledge- and of assembly-line workers: Managerial excellence is marked by capabilities of model-literacy and self-management on *all* hierarchical levels [11].

So, models of functions and processes became representations of complex interdependent activity. In parallel, ICT became a driver of model complexity since automation asked elaborating models with high precision and detail. Figure 2 depicts a diagram accordingly to the BPMN (Business Process Modeling Notation) [35], an advanced standard, capturing organizational structures in horizontal lanes (remains of the silos) as well as structures and rules of proceedings and interaction (connectors or auxiliary information).

It answers the question: Who does what (why), where, when, how and with what and whom? Functions now are embedded as physical or intellectual resources [12]. With growing vertical (along managerial hierarchy) and horizontal (same level of activity) integration of automation current ICT covers the value chain depicted in Figure 1, while process-cost accounting [13] enabled new strategies of controlling economics of managing operations complexity.

But, scale and complexity of models grew (consider the complexity of operations landscapes drafted in the introduction) and in accordance to Ashby's Law of requisite variety [14], richness and heterogeneity of detail or the manifold of interfaces did not turn into economies of scale but became drivers of costs and risks to run out of budgets. Complexity started escaping from modeling capabilities.

C. Modularizing and Encapsulating Functionality
The Regime of Services

There are solutions to this problem: modularization and virtualization. Modules with standardized interfaces reduce variety and hide details of the functionality they provide. So, challenged by faster change and increasing demandingness in markets as well as by increasing costs, automotive industry introduced *platforms* that decouple the variety of car-models from the variety of parts they are built from.

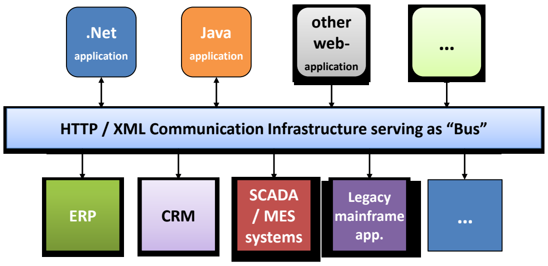


Figure 3. SOA-based Application Architecture.

This concept of *functionality* provided by a *module* also was adopted by ICT in service-oriented architectures (SOA) for services collaborating via the Internet (Figure 3). Promises reach from risk reduction or portability and re-usability in software development to integrated, intelligent operations like in the vision of the *Internet of Things and Services* [15] [16]. And, as we shall see later, in a general view, everything can be seen as provider or demander of service.

Based on service description standards [17], web services are capable of autonomously composing complex dynamic networks involving things (in the simplest case via radio tags, in future also enabled by embedded multi-core computing capacity), legacy systems as well as human users in the roles of operations' supervisors or IT operators.

However, due to latency times or high customization, legacy IT is hard to be integrated into service landscapes or to be decomposed into sub-services [18]. The competition of big players (IBM, Oracle, SAP, etc.) in direction will fragment web-standards. Or, the vulnerability of Internet-based services and dependencies on intermediaries raise security concerns. Yet, in spite of achievements, also SOA does not sustainably reduce the complexity of models [19] [20].

To not get lost in variety Volkswagen implemented a new platform, the *Modular Transverse Matrix*. It reduces the number of modules by up to 90 % across 10 brands (Seat, VW … Audi, Porsche, Bentley) [21]. However, it is hard to believe that this is applicable to examples in aviation industry. And, by and by, markets also will coerce VW to accept and accommodate new variety. Accordingly, complexity will reconquer modularized operations and related models or applications' architectures [22] [23].

D. Introducing Meaning
The Regime of Semantic Modeling and Ontologies

Adopting Internet-based communication and cooperation semantic models became relevant: Languages are sets of tools to create meaning of signs or symbols in communities. So far, however, technology only can interpret sequences of signs obeying syntactic rules, e.g., as command to delete a text. There is no understanding of meaning.

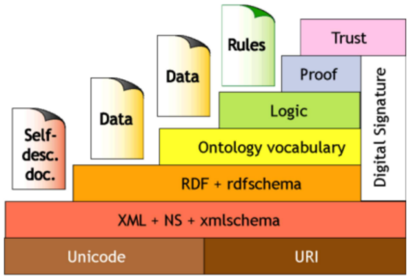


Figure 4. The Semantic Layer in the W3 Architecture [24].

The idea of the semantic web [25] is to encode semantic information into web-pages. In the W3-architecture of Berners-Lee (Figure 4) [24] ontologies provide the *vocabulary*, more generally, the knowledge enabling to recognise and associate services. The common vocabulary enables modelling relations to other objects and properties relevant to relations.

Reading meaning, a sequence of signs would not only be just a code for deterministic execution of commands, but for understanding *know-why* and context, i.e., *the content of the service it provides*: Which idea is leading the search? What is a function about? Does it fit into a particular scene? What is the meaning of or responses to events in contradictive context (costs, quality, security, operations' footprints)?

Ontologies, among others, enable capturing local knowledge about objects related to operations like orders or resources. Given that these local models are globally consistent (non contradictive), complete (no relevant aspects missing) and clear they conceptually and practically support integrating large and massively distributed operations' systems [26] and a designer of a web service can describe a service without knowing other services.

On the other side, semantic interoperability needs a non-trivial degree of accuracy of the model. It is obvious that this problem increases with the complexity to be captured. Although semantic modeling is a great step for itself, it does not decouple from complexity.

III. Implicit Aspects of Decoupling: Towards Freedom-to-Act-Architectures

A. Decoupling Modeling from Domain Complexity

The example of Boeing's or Airbus ramp-ups shows that the approach of modeling domain complexity forces a trade-off between simplification and risk that butterflies, long-tail risks can materialize (*black swans*) or open events may affect operations (*openness to environments*) each speedily grow to a 7-, 8-, or even 9-digit € problem. In a complex domain simplification is evidence that this complexity is out of control of modeling and management. It is a bad bet. Decoupling by agent-based modeling (ABM) [37] is the alternative.

A first step is realizing that complexity is a property of dynamic systems [29] [30]. It emerges from degrees of freedom of interacting elements (or agents) the system consists of. These agents may act autonomously (executing individual decisions for achieving individual objectives with individual resources like humans or *things that think*) or not (simple things just connected by sensing technology).

Decoupling relies on a division of labor amid ABM and Multi-agent Systems (MAS). Instead of modeling operations' complexity, ABM focuses on the behavior of agents and the framework of interaction complexity emerges from. The knowledge required is in the relationships and properties of objects or in target functions, rules and protocols of agents' communication. It can be captured in ontologies as local knowledge from or also directly modeled by the people working in the domain. Software-agents can read ontologies and MAS organize interaction. Thus, complexity is not in the model, but emerges in the MAS. Since ABM is scalable to resolution of objects [5] or open to any new object, it also enables capturing butterfly effects or black swans.

Occasionally, as shown by the sequence of paradigms discussed in Section II, there also is a revolutionary and irreversible change of sets and settings of agents and of patterns of interaction. So, the success of the Lean Regime does not allow returning to Ford's strategies. This *evolutionary aspect* of operations' systems is explained by the theory of *fitness landsca*pes [27] as discriminating change of control requirements and capacities of self-reproducing (autopoietic) systems [28]. But, although it may require a thoroughly review of models, it still can be handled by principles of ABM.

B. The Definition of Freedom-to-Act (FTA) Architectures

Dynamics of systems formed by agents can be described as the change of states of agents. These states are results of previous activity and, at the same time, resources of subsequent action; they imply degrees of freedom future states emerge from. This view we consider to be relevant for effectively modeling and managing complex dynamic systems.

Since the behavior of complex systems hardly is reducible to models and to be predicted, they ask to adapt to unexpected, thus unplanned change. Therefore, FTA are the most decisive resource or control parameter of managing complex systems and the viability of any solution depends on the physical (feasibility of the solution) and the legal (legitimacy) availability or exploitability of FTA.

For example, agents representing 'busy' service trucks in airport operations will not respond to another order, except there are FTA allowing to shift current jobs in time or to transfer them to other agents. Agents representing parts to be supplied to an assembly line may have the state 'delayed' and ask for mitigating action, i.e., other agents to take action in reach of *their* FTA. Or, due to a satisfying crash-test of new material, agents representing the respective objects in engineering may release contingency budget by reducing the value of the event risk "readiness for manufacturing delayed" and by this release the FTA of other agents.

C. Lean Modeling, Service-orientation and Semantic modesty

In agent-based modeling, ontologies do not only serve as dictionaries but primarily as frameworks of agents' behavior. They need capturing dimensions of acting like space (aircraft negotiate new routes) or organizational affiliation of staff (experts may be sent for supporting suppliers).

Models also need support sensitivity to events (aircraft passed control-point; CF-component is ready for manufacturing) as well to context in terms of domains of operations (engineering or administration) or to possibly contradicting objectives (reducing costs versus reducing environmental footprints). Also, connectivity to sensors or other sources is required.

Undisputedly, semantic models can be very complex and under conditions of fast change *the challenge is keeping ontologies complete and consistent in order to enable MAS of continuously providing viable solutions in spite of frequent updates*. One problem may be the *manifold of relationships between objects*, another one the *sophistication of semantics*. To prevent complexity from returning through backdoors, we therefore, suggest two principles of modeling: service-orientation and semantic modesty.

As discussed, *service-oriented architectures* support modularization and encapsulation of functionality. Mainly they are used for modeling web-services. *The same strategy can be applied in ontologies by organizing models of agents' behavior in a service-oriented way.* On that base, agents form dynamic networks by offering and consuming services on virtual markets which allow controlling activity by cost, price, or margins [30].

Service-oriented modeling can significantly contribute to the clarity of ontologies since *any* relation may be modeled as a service-relationship, since FTA can be understood as capabilities or needs to provide or consume a service and since any property of agents can be arranged accordingly. For example, a truck may offer transport services and demand services of gas stations – both substantiated by FTA in terms of maximum payload or level of fuel. Drilling deeper, payload is a service offered by structural components of the truck …

This concept also works in less usual cases. So, the relation *table "has" legs* (*leg "is part of" table*) translates into:

"*Legs provide the service of keeping distance from ground by 72 cm*" and into sequences like: "*user agent asks for table of heights of 72 cm*" → "*table agent asks …*". This example from ergonomics and logistics of modularized office furniture system in a large organization may look strange, but it works and provides clarity.

Semantic sophistication is about risks of modeling complexity *into* ontologies. Certainly, there are applications that cannot avoid this problem like a semantic search engine that may have to discern the meaning of "time" in philosophical, physical, economical, or sociological contexts. The problem increases with distance from models in natural sciences, engineering or direct business operations. In the latter 'time' is the clock-time planned or elapsed between events. Already in more abstract economic contexts like *time to volume* (achieving the crestline of production) and *time to amortization* (progress of effective sales) the meaning complicates. New managerial regimes introduce new concepts of time, like 'synchronization' which is highly relevant to lean management but far less for Henry Ford's functional silos.

Consequently, by *limiting the variety of relations* and being *modest* in terms of semantic complexity the global consistency of local modeling is significantly easier to validate. Like in Lean Management, it abandons self-inflicted complexity and compares to the VW Matrix which at least for the time being abandoned 90% of variance.

Semantic, service-oriented agent-based modeling also is *scalable to increasing resolution* of object and time [5]. If another 'butterfly' or 'black swan' is identified to affect the behavior of the system it can be added as another agent providing and consuming services as well as owning the properties related to these services. And subsequently the MAS will process the model.

D. Processing Freedom-to-Act Architectures and the Example of Ontology-based Multi-Agent Systems

Why don't wheeled animals exist? Because wheels do not provide animals with FTA required sustaining in their environments. Life is the machine producing survival or extinction by processing real animals' FTA. Alike, the leanness of firms is processed by markets returning profit or loss.

In real-time business operations, FTA are explicitly processed in a mode inofficially called *improvisation*. Experienced, focused and observing dispatchers or operators at any time know the FTA of "their" resources. If one fails in a tight situation they know alternatives that can serve for mitigation or remedy (compare idle slots in Figure 5). The same works in MAS: FTA-awareness of dispatchers is replaced by collaborating agents, each by definition knowing its FTA and the proceeding of exploiting them at the best.

Multi-agent systems are *the* model of software for processing FTA, whether they are formed by software agents acting in the local memory of a server or across a grid of servers, by agents acting in the Internet (web-services, things like aircraft2aircraft [34] or car2car communication in future traffic management scenarios [36]) or in collaboration with human analysts, planners or deciders.

In MAS, agents process semantic models of relations, properties or other aspects of the framework like objectives, metrics, negotiation or reporting protocols that shape agents' behavior. With some simplification, relations between agents compare to relations between actors playing a role and ontologies compare to the role scripts. Performance's quality relies on both, the quality of scripts and the talent of actors comparing to the quality of code and properties of agents.

Instead of fighting complexity, the peer2peer architecture of MAS take advantage of it by exploiting FTA, i.e., disposable capacity. If the behavior of the system at least is statistically predictable, an optimal, e.g. cost minimizing plan can be delivered that, except in terms of contingency buffers, avoids idleness of resources (*muda*). Disposable capacity is avoided and no exploitable FTA are left.

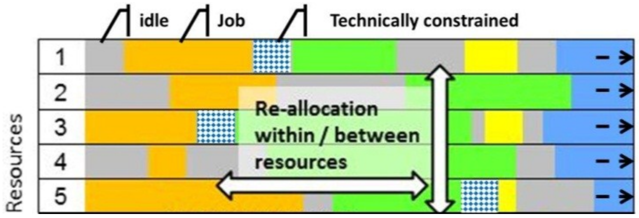


Figure 5. Gantt Diagram with Jobs and Idle Time.

The Gantt Diagram shown in Figure 5 displays the allocation of resources to jobs as well as idle slots which an optimal plan tries to avoid. But, in another view, these idle slots also represent a part of the relevant spectrum of FTA: disposable time, a major resource of adapting to unplanned events. Again, there are counter-intuitive aspects:

- On one hand, objectives of optimization are incompatible with the unpredictability of complex systems since the validity and viability of any optimum relies on the reliability of underlying assumptions.
- On the other hand, muda to be avoided accordingly to principles of optimization is the major resource of adaptiveness to unexpected events. Thus, principles of optimization may detract resources required to adapt to unexpected events.

These problems significantly increase with complexity: As in reality, FTA (agent's disposability) are floating in a non predictable way. And the more complex, thus uncertain the scene the more likely there is a need for redundancy that may serve as resource to a solution.

IV. Summary

In the light of Ashby's Law [14] '*competing'* can be conceived as effort to achieve, maintain or increase control in a competitive environment, i.e., to dispose of freedoms to act competitors cannot control. Therefore, competition in direction drives complexity. That way Ford's mass-production system (standing for all car-makers of this type) with its almost strict decomposition of work exceeded complexity of car-by-car garage production.

Then, overcoming self-inflicted complexity of silos, Toyota invented lean manufacturing systems, again more complex than the regime they attacked. Consider demands of strategies like Kaizen or Just-in-Time (JIT) for process-spanning thinking, model literacy or continuous learning on all hierarchical levels: Lean management needs, employs and develops organizational and collaborative intelligence.

Since competition does not stop driving complexity, platform strategies answered in operations as in modeling (SOA) or ICT, among others for automation of operations (web services). The Internet of things and services or the semantic web are next steps.

Modeling architectures are running after this still accelerating development. To maintain chances of effective modeling it becomes paramount decoupling them from domain complexity by agent-based as well as lean concepts. Multi-agent systems are able of processing these models and exploiting FTA in response to unexpected events.

V. Selected Topics of Future Work

The view at FTA should serve as bridge between traditional and future handling of complexity. Among others, MAS, the reference used here, likely failed joining the mainstream because of a lack of such bridges in the thinking of users and developers [31]. In the following, selected aspects of further proceeding will be sketched.

A. The Acceptance Problem: Trust in a Black-Box?

There are serious technical problems and security concerns which finally may find acceptable solutions. But, the most fundamental problem is not about technology or automation but about organizational integration.

In business environments, conceiving and tackling complexity as a resource commonly interferes with conventional concepts of governance and management: Still the ideas of calculability and optimizability (full control) prevails and not principles of continuous and potentially experimental adaptation and approximation. Redundancy is not considered to be a resource.

With major EU industry, we currently are developing strategies on integrated risk management across large-scale operations landscapes. The approach is driven by competition and complexity that reduces time windows to be passed for amortization of capital-intensive projects and hardly leaving alternatives to new, ABM- and FTA-related strategies.

B. Improving Modeling and Auto-code Technology

There are ontology editors and debugging tools available and the strategy of reducing semantic complexity has the potential of substantially simplify the development of ontologies. But, there is still a long way from ontology to effective code. As an example, let us take the fasteners neglected in Boeing's Virtual Ramp-up System. Including such aspects at a later point of time, may need implementing new classes of objects providing respectively asking new services.

On their own resources, users may easily implement new classes of objects and services into ontologies. Implementing it into a multi-agent system requires software developers. Therefore, strategies, architectures and tools have to be elaborated enabling users (more) directly, at the best without developer support, implementing new classes objects into processing systems (e.g., MAS).

We call this objective "WYKIWYG" – what you know is what you get. It may have an impact in terms of model driven software engineering or possible automated code generation and with this increase acceptance of users.

C. Criticality Management, an Application Example

Criticality is a control parameter of complex system defined as the scale-free point of a phase transition, e.g., from liquid to solid, from stable to unstable (a pile of sand) or from able to unable of response to unexpected events, disposing or not of respective FTA. To effectively manage criticality it is to be estimated.

As a parameter of complexity, also criticality emerges in operations, and is to be modeled as global property (like the temperature of a solid object). Air traffic or manufacturing systems are dynamic networks formed by large numbers of agents. In simulation or real operations' control respective FTA can be logged. Stochastic models could be explored to estimate criticality and working it up for management. These aspects are very closely related to research on risk management mentioned.

D. Convergence with HPC-Problems

There is serious indication that modeling and computing of business operations' on one and domains and issues of High Performance or High End Computing (HPC, HEC) on the other side converge. Operations' models may be smaller as, e.g., climate models.

But, in terms of distribution and non-linearity these applications compare to typical HPC applications. In terms of the variety of agents (including autonomous ones) and the manifold of parameters they may even be more complex. In both domains, FTA are a relevant concept. Reversely, HPC addresses real-time collaboration and learning in a way comparable to a project on intelligent manufacturing ramp-up, we shall start end of 2012 [32] [5].

References

[1] Inden, U., Tieck, St., and Rückemann, C.-P. (2011). Rotation-oriented Collaborative Air Traffic Management. In: C.-P. Rückemann, W. Christmann, S. Saini, & M. Pankowska, (Eds.), Proceedings of The First International Conference on Advanced Communications and Computation, INFOCOMP, October 23-29 2011. Pp. 25-30, ISBN: 978-1-61208-161-8.

[2] Aero International (2009). Boeing 787 – Der schwierige Weg zum Erstflug. Aero International 2009, Vol. 4. Pp. 40-43.

[3] Kinsley-Jones, M. (2010). Giant steps – Has the latest Airbus A380 production ramp worked. Flightglobal, July 12th 2010, Retr. Feb. 13th 2011. http://www.flightglobal.com/news/articles/farnborough-giant-steps-has-the-latest-airbus-a380-production-revamp-343814/.

[4] Ostrower, J. (2010). A380 fleet grounded following Trent 900 failure. Flightglobal Blog, Nov. 4th 2010. Retr. May 1st 2012, http://www.flightglobal.com/blogs/flightblogger/2010/11/qantas-a380-fleet-grounded.html.

[5] Taleb, N. N. (2007). Black Swans and the Domains of Statistic. The American Statistician Association, Vol. 61, Issue 3, 2007. DOI: 10.1198/000313007X219996. Pp. 1-3.

[6] Müller-Stewens, und G., Fleisch, E. (2008). High-Resolution-Management: Konsequenzen des Internet der Dinge auf die Unternehmensführung. Führung & Organisation 77 (5), Pp. 272-281.

[7] Taylor, F. W. (1911). Principles of Scientific Management. New York and London, Harper & brothers.

[8] Hughes, T. P. (2004). American Genesis: A Century of Invention and Technological Enthusiasm, 1870-1970. The University of Chicago Press. IBN: 0-226-35927-1.

[9] Porter, M. E. (1985). Competitive Advantage: Creating and Sustaining superior Performance. The Free Press, New York (1998). ISBN 0-684-84146-0.

[10] Womack, J., Jones, D., and Roos, D.: The Machine that changed the World – The Story of Lean Production. Harper Collins, New York 1990, ISBN 978-0-06-097417-6.

[11] Ohno, Taiichi (1995), Toyota Production System: Beyond Large-scale Production, Productivity Press Inc., ISBN 0-915299-14-3.

[12] BPMN standard: http://www.bpmn.org/. Retr. Mai 12th 2012

[13] Mayer, R. (1998). Kapazitätskostenrechnung: Prozesskostenrechnung, Lösungsansatz für indirekte Leistungsbereiche, Vahlen, München. ISBN 3-8006-2366-8.

[14] Ashby W.R. (1956). An Introduction to Cybernetics. London, U.K.: Chapman & Hall Ltd. Pp. 206–212.

[15] Ten Hompel, M. et al. (2008). Künstliche Intelligenz im Internet der Dinge: Die Zukunft der Materialflusssteuerung mit autonomen Agenten. Jahrbuch der Logistik 2008. Pp. 24-29.

[16] Karnouskos, S., Savio, D., Spiess, P., Guinard D., and Trifa V., Baecker O. (2010). Real-world Service Interaction with Enterprise Systems in Dynamic Manufacturing Environments. In: Artificial Intelligence Techniques for Networked Manufacturing Enterprises Management. ISBN 978-1-84996-118-9, Springer, Pp. 423–457. Retr. Sept. 22nd 11.

[17] UDDI standard: https://www.oasis-open.org/committees/tc_home.php? wg_abbrev=uddi-spec. Retrieved May 12th 2012

[18] Weiss, O. (2001). ERP-Systeme müssen flexibler werden. Computerwelt Sept. 9th 2011. Retr. October 17th 2011. http//www.computerwelt.at/?id=251&tx_ttnews[tt_news]=45815.

[19] den Haan, J. (2007). Model-Driven SOA. Sept. 13th 2007. http://www.theenterprisearchitect.eu/archive/2007/09/13/model-driven-soa. Retr. Jan. 8th 2012.

[20] de Groot, R. (2008). Top 10 SOA Pitfalls: SOA does not solve complexity automatically. May 19th 2008. http://blog.xebia.com/2008/05/19/top-10-soa-pitfalls-6-soa-does-not-solve-complexity-automatically/. Retr. Jan. 8th 2012.

[21] VW (2012). Volkswagen introduces Modular Transverse Matrix. Retr. Feb. 12th 2012. http://www.volkswagenag.com/content/vwcorp/info_center/en/themes/2012/02/MQB.html.

[22] Muhammad, S.S., Myers, D., and Sanchez C.O. (2011). Complexity Analysis at Design Stages of Service Oriented Architectures as a Measure of Reliability Risks. In: Milanovic N., Engineering Reliable Service Oriented Architecture: IG Global. ISBN13: 9781609604936. Pp. 292-314.

[23] Tran, H., Zdun, U., and Dustdar, S. (2007) View-based and Model-driven Approach for Reducing the Development Complexity in Process-Driven SOA. International Conference on Business Process and Services Computing, volume 116 - Lecture Notes in Informatics. P. 105—124.

[24] Berners-Lee, T. (2000). Presentation. Semantic Web on XML. XML 2000. Slide 10. Retr. Jan. 24th 2007 http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html

[25] Berners-Lee, T., Hendler J., and Lassila O. (2011). The Semantic Web, A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientific American Feature Article: The Semantic Web.

[26] Stuckenschmidt, H. (2009). Debugging OWL Ontologies – A Reality Check. In Petrie Ch.: Semantic Web Service Challenge: Proceedings of the 2008 Workshops. Stanford Logic Group, Computer Science Department. Stanford University. Retr. Dec. 12th 2010. http://logic.stanford.edu/reports/LG-2009-01.pdf.

[27] Kauffman, St. (1995). At Home in the Universe. Oxford University Press, New York. P. 200 ff.

[28] Maturana, H., and Varela, F. (1992). The tree of knowledge: biological roots of human understanding (1984). Boston, MA: Shambhala Publications, Inc.

[29] Skobelev, P. (2011). Bio-Inspired Multi-Agent Technology for Industrial Applications. In: Alkhatheb F., Al Maghayreh E., Doush A.: Multi-Agent Systems – Modeling, Control, Programming, Simulations and Applications. Pp. 495-522. InTech, ISBN 978-953-307-174-9.

[30] Rzevski, G. (2011). A practical Methodology for Managing Complexity. Emergence: Complexity & Organization.13 (1-2), Pp. 38-56.

[31] Vrba, P., Kadera, P., Jirkovsky, V. Obitko, M., and Marik, V. (2011). New Trends of Visualization in Smart Production Control. In: Marik V. et al.: Holonic and Multi-Agent Systems for Manufacturing, HoloMAS 2011. Springer, Berlin ISBN 978-3-642-23180-3. Pp. 72-83.

[32] ARUM – Adaptive Ramp-up Management. European Research Framework Program, Project 312056. Planned Start: Sept 1st 2012. Project Management: EADS.

[33] Sloman, St. A., and Fernbach, P.M. (2011). Human representation and reasoning about complex causal systems. Journal Information, Knowledge, Systems Management. IOS Service. Volume 10, Number 1-4 / 2011, Chapter 5. Pp. 85-99.

[34] SESAR, Single European Sky ATM Research: http://www.eurocontrol.int/sesar/public/standard_page/overview.html. Retr. Nov. 14th 2009

[35] On Business Process Model and Notation (BPMN). http://www.omg.org/spec/BPMN/1.2/. Retr. Jan. 19th 2012

[36] Car-to-Car Communication Consortium. http://www.car-to-car.org/. Retrieved May 12th 2012

[37] Allan, R.J. (2009). Survey of Agent Based Modeling and Simulation Tools. Retr. June 21st 2012. http://epubs.cclrc.ac.uk/work-details?w=50398.

A Neural Network Ultrasonic Sensor Simulator for Evolutionary Robotics

Christiaan J. Pretorius*, Mathys C. du Plessis†, Charmain B. Cilliers†
*Department of Mathematics and Applied Mathematics
†Department of Computing Sciences
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa
emails: {cpretorius, mc.duplessis, charmain.cilliers}@nmmu.ac.za

***Abstract*—Evolutionary Robotics is concerned with using simulated biological evolution to automatically create controllers for robots. Simulation, which reduces the amount of real-world testing, is typically used to accelerate the evolution process. However, the creation of robotic simulators is a difficult and time-consuming process which requires expert knowledge. As an alternative to manual simulator creation, this paper describes the use of Neural Networks to act as simulators for an ultrasonic distance sensor in the Evolutionary Robotics process. The creation of the simulator Neural Networks is discussed and motivated. The simulators are evaluated by means of a comparison with test data. Finally, the simulators are validated by evolving a controller for an obstacle avoiding robot using the simulator Neural Networks. The experimental results show that Neural Networks can indeed be used to simulate an ultrasonic sensor in the Evolutionary Robotics process.**

Keywords-Robotics; Genetic Algorithms; Neural Networks; Simulators.

I. Introduction

Great advances have been made in recent years in the field of robotics. Robots are becoming cheaper, with more hardware capabilities and faster onboard computing [1]. A robot's behaviour is determined by a *controller*. The controller continuously receives input from the robot's sensors and gives output in the form of commands, for example motor speeds [2]. The underlying implementation of a controller depends on the application domain of the robot.

The manual creation of a controller by human experts is a time-consuming and complicated task, which may be infeasible due to the complexity of the robotic task [3]. The unstructured, noisy and dynamic nature of the real world environments in which robots are required to function, adds to the difficulty of creating controllers [4]. The cost of creating controllers, which currently constitutes up to a third of total expenses [5], will increase in future as robotic hardware continues to advance and the tasks required from the robots become more complex [6].

Evolutionary Robotics (ER) is a field that aims to simplify the creation of controllers by means of the principles of biological evolution [7]. Engineers can use ER to create complex controllers with minimum human input. The ER process is an extension of the theory behind Evolutionary Algorithms (EAs) to the realm of robotic controllers. A population of candidate controllers compete for the ability to produce offspring, based on the effectiveness of each controller.

Simulation has been used extensively in the ER process to avoid having to evaluate the performance of candidate controllers in the real world. The creation of these simulators may in itself be extremely time-consuming or infeasibly complex. This paper consequently reports on a different technique of creating robot simulators, based on Neural Networks (NNs). NNs have previously been used as controllers [8], but are not commonly used as simulators in the ER process. Previous research by the current authors have shown that NNs can be used as simulators in the ER process for motion simulation [9], and for modeling various sensors [10][11].

The focus of this paper is on NN simulators for an ultrasonic distance sensor. The ultrasonic sensor used in the study differs from previously modeled sensors in that it is less reliable and that its functioning is intermittent. These deficiencies require a slightly different treatment when simulating this sensor.

The remainder of this paper is structured as follows: Section II details the ER process and describes the role of simulation in the ER process. Section III provides related work on using NNs as simulators. Section IV gives a brief overview of robotic controllers. The experimental robot that was used in this study is described in Section V. Section VI describes the NN simulators that were used in this study along with the data acquisition approach. The training of the NNs is discussed in Section VII. An analysis into the accuracy of the trained NNs is given in Section VIII. The NN simulators are validated in Section IX by their use in the ER process to evolve a controller. Conclusions are drawn in Section X.

II. ER and Simulation

The ER process, in the context of this study, is a technique that can be used to automatically create controllers for robots by means of artificial intelligence. The basic procedure is as follows:

1) Randomly create a set (referred to as a population) of controllers for the robot.
2) Evaluate the effectiveness of each controller (known as the fitness of each controller). Stop the process if an adequate controller has been found.
3) Create offspring from the current population, giving preference to the more fit controllers by means of:
   - Mutations (Small random changes to each controller)
   - Crossovers (Combinations of subcomponents of parent controllers)
4) Replace the current population with the offspring population.
5) Return to Step 2.

Step 2 is typically the most time-consuming of the ER process as it requires the candidate controllers to be transferred to a real-world robot to evaluate their performance. The evaluation of controllers in the real world may not be possible, as a large number of controllers have to be repeatedly evaluated [1][7][12]. Furthermore, certain controllers could lead to erratic robotic movements which may potentially damage the robot hardware [1]. Evolution in simulation has been used by several researchers to avoid the time-consuming task of real-world evaluation [12][13][14].

The computational complexity of simulators must allow for relatively fast evaluation of controllers [15], while still approximating the real robotic environment [16]. Simulators have been created by several researchers [6][12][13][17][18][19][20] and can be categorised in three classes [11]:

- **Physics-based simulators** mathematically model the physical behaviour of robotic components. This type of simulation is typically accurate [21], although physics simulators use complex physics models [22] and their construction requires a considerable amount of human input [23]. Furthermore, physics models often contain simplifications or approximations of the real world, which could lead to factors like friction and inertia being overlooked [14].
- **Empirical models** make use of data collected from the real world [2][12][14]. This approach has the advantage of capturing the fuzzy characteristics of the robotic components [12]. A disadvantage of empirical models is that elementary data analysis techniques are often used in their construction, for example, basic interpolation (although more advanced techniques have been investigated [19][20]).
- **Hybrid models** combine physics and empirical models by utilising experimental data to optimise the parameters of the physics model [13]. A disadvantage of this modeling technique is that assumptions from the physics model are inevitably incorporated into the simulator.

The challenges in existing robotic simulators is the motivation for an alternative simulation scheme, namely simulators implemented using NNs.

III. Neural Network Simulators

Artificial Neural Networks are used to model the biological brain in software and consequently harness the computing power of the biological neurons by training the network to perform certain tasks. This research involves the use of NNs as robotic simulators in the ER process.

NNs are well suited for use as robotic simulators because of their noise tolerance and generalisation ability. The environment that is to be modeled is typically employed to create training data used in the construction of the NNs. Large amounts noise is inevitably present in the training data due to inaccuracies in the acquisition process [13]. NNs are known to be noise tolerant [24][25], which makes their application to this domain appropriate.

Training data inherently contains only a sample of the set of all states of the environment. Movements of a robot through its environment can only be sampled at discrete intervals. The number of samples are also constrained by the amount of time that is available to create training data. The ability to generalise over training data [26] makes it possible for NNs to interpolate to unseen environmental states.

NNs have been used as simulators in non-ER fields, for example, to produce realistic graphic animations [27], to animate a robotic arm [28], and to model the environmental interaction of a sonar sensor of an experimental robot [29].

To the best of the authors' knowledge, however, no previous investigations have been conducted into the usage of NNs as simulators in the ER process, apart from previous works by the authors. The current authors have previously demonstrated the use of NN simulators in the ER process for motion simulation [9] and modeling of several sensors, including: light sensors [10], touch sensors, tilt sensors and gyroscopic sensors [11]. This study demonstrates that Simulator Neural Networks (SNNs) can also be used for ultrasonic sensors.

IV. Robotic Controllers

This study is focused on a controller to perform obstacle avoidance. The goal of such a controller is to move a robot through a scene to a destination point as fast as possible without colliding with any obstacles [8][13][30][31]. A NN controller has previously been evolved to perform the obstacle avoidance task [8].

The controllers that were evolved in the current study were evaluated in the real world on an experimental robot which was created to perform the obstacle avoidance task.

V. Experimental Robot

The Lego® Mindstorms NXT [32] robotic components were used in this study to construct the experimental robot.

The components that were used in this study include the *central micro-computer*, an *ultrasonic sensor* and *servo motors*.

An *Obstacle Avoidance Robot (OAR)* was constructed from the NXT components. Two motors connected to wheels and two castor wheels were used to create the differential steering of the OAR. The orientation of the robot was obtained from a compass sensor during training, while the ultrasonic sensor was used to determine the distance between the robot and obstacles in the scene. SNNs were constructed to model the OAR's motion and sensor functioning.

VI. SNN Parameters and Data Acquisition

Empirically obtained training data was used to construct the SNNs for motion and ultrasonic sensor simulation. The ultrasonic sensor's readings were recorded as random commands were sent to the robot. This information was later extracted to create training data for the SNNs. The training data for the motion SNNs were obtained through motion tracking. Discretised time steps of 400ms were used as the time frame in which simulation was performed.

A. Motion Simulation

The SNNs for the motion simulation of the robot were described in [10][11] and are consequently only briefly mentioned here. The robot operated on a horizontal slip-resistant surface in a coordinate system that moves and rotates with the robot. Three separate SNNs were created to simulate changes in the x and y coordinates, and the orientation angle (Δx, Δy and $\Delta\theta$, respectively).

The current and previous motor speeds of the motors were given as inputs to each SNN. The inertia of the robot was thus taken into consideration by including previous motor speeds. Frames from a camera mounted on the ceiling were analysed to track the position of the robot. The tracking data concerning the orientation change angle was supplemented using the compass sensor to obtain averaged values. The state changes in response to random commands, in terms of orientation angle and position, were used to generate a set of training data for the motion SNNs.

B. Ultrasonic Sensor Simulation

To simplify the construction of the SNNs for the ultrasonic sensor, the assumption was made that all obstacles in the robot's environment have straight and perpendicular edges and that any obstacle could thus be represented simply by its bounding rectangle. The parameters used in the ultrasonic sensor SNNs are indicated in Figure 1. The distance, D, between the ultrasonic sensor and the relevant obstacle was taken into account, as well as the orientation angle, α, between the pulse emitted from the sensor and the normal to the relevant edge of the obstacle in question.

The ultrasonic sensor sometimes does not return a reading at all if the emitted pulse does not find its way back to the receiver on the sensor. Testing suggests that the ultrasonic sensor only successfully produces a value roughly 80% of the time when it is at random positions and orientations relative to obstacles. In order to model the functioning of the ultrasonic sensor realistically, it was thus deemed necessary to model the probability of the ultrasonic sensor returning a value. The assumption was made that the probability of the ultrasonic sensor successfully producing a reading depended on D and α (Figure 1).

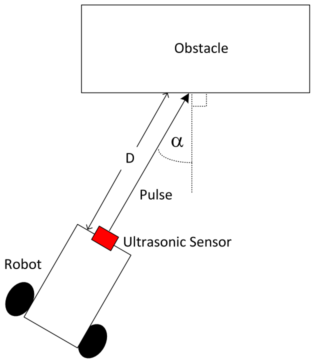


Figure 1. Parameters used in ultrasonic sensor models

Two different SNNs were thus employed to model the operation of the ultrasonic sensor. The mapping expected from each of these SNNs is shown in equations (1) and (2).

$$SNN_{ultra\,prob} : \{D, \alpha\} \rightarrow \{ultra_{prob}\} \quad (1)$$

$$SNN_{ultra\,value} : \{D, \alpha\} \rightarrow \{ultra_{value}\} \quad (2)$$

The first of the two ultrasonic sensor SNNs was used to predict the probability of the ultrasonic sensor producing a value ($ultra_{prob}$), given as inputs the distance and orientation angle to a relevant obstacle. This probability was expressed in the range [0, 1]. In the event of the ultrasonic sensor producing a reading, the second SNN would then be used to predict the actual reading produced by the ultrasonic sensor ($ultra_{value}$), again given as inputs the distance and orientation angle.

The random movement commands given during the motion tracking phase were used to move the robot through a scene containing various solid, straight-edged obstacles. By making use of the known position and orientation of the robot at any given point in time as well as the ultrasonic sensor values recorded to onboard memory, data could be parsed relating the probability of the sensor producing a value and this value itself to various distances from and orientations relative to obstacles. This would provide training data for the ultrasonic SNNs.

The sources of errors in the training data, i.e., human errors, inconsistencies in the motor control and noisy functioning of the ultrasonic sensor, were reduced as much as possible, for example, by repeating all physical measurements twice. Nonetheless, a considerable amount of noise remained which could potentially make the effective training of the SNNs challenging.

VII. Network Training

The SNNs were trained using a Genetic Algorithm (GA), although other optimisation techniques may also have been used. Each individual in the GA encoded potential weight values for a candidate SNN directly. A population of 100 randomly initialized chromosomes were used. A tournament among 33 individuals were used to select offspring. Simulated Binary Crossover [33] occurred with a probability of 80%, while mutations (normally distributed random values were added to each component) occurred with a probability of 5%. Training was halted when an improvement of less than 0.1% was found in the inverse of the mean-squared error of the fittest individual in the population over a period of 300 generations of the GA.

A Feed-Forward Neural Network (FFNN) with a single hidden layer was used as topology for the SNNs. Appropriate values for the number on neurons in each hidden layer were experimentally determined [34]. Each hidden layer contained an equal number of summation and product units, while output layers contained only summation units. Table I lists the activation functions used by each SNN, along with the number of hidden neurons that were employed.

Table I
Details of each SNN

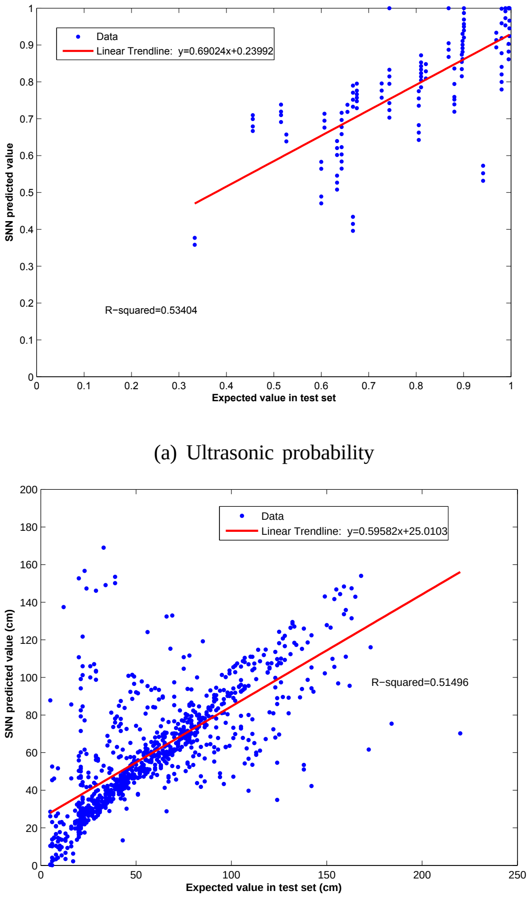
SNN	Activation Function		
	Hidden Layer	Output Layer	Number Hidden
Orientation Angle	Linear	Linear	4
X-coordinate	Linear	Linear	20
Y-coordinate	Linear	Linear	20
Ultrasonic Probability	Linear	Ramp	10
Ultrasonic Value	Linear	Linear	40

VIII. SNN Training Accuracy

The training accuracy of the ultrasonic SNNs is illustrated in Figure 2. Each graph gives the output of the SNN plotted against empirically obtained values from the real world. The empirically obtained values used to perform this analysis were not used during the network training phase and were thus previously unseen by the SNNs. Each graph contains a trendline with its associated equation and R^2-value.

Figure 3 gives three-dimensional plots of the SNN outputs based on the distance and orientation of the sensor with respect to the obstacle. Two views of each surface are shown from different viewing angles. The plots also contain data from the test set for comparison.

Figures 2 and 3 illustrate that the SNNs generally trained relatively well (although the R^2-values are relatively low).



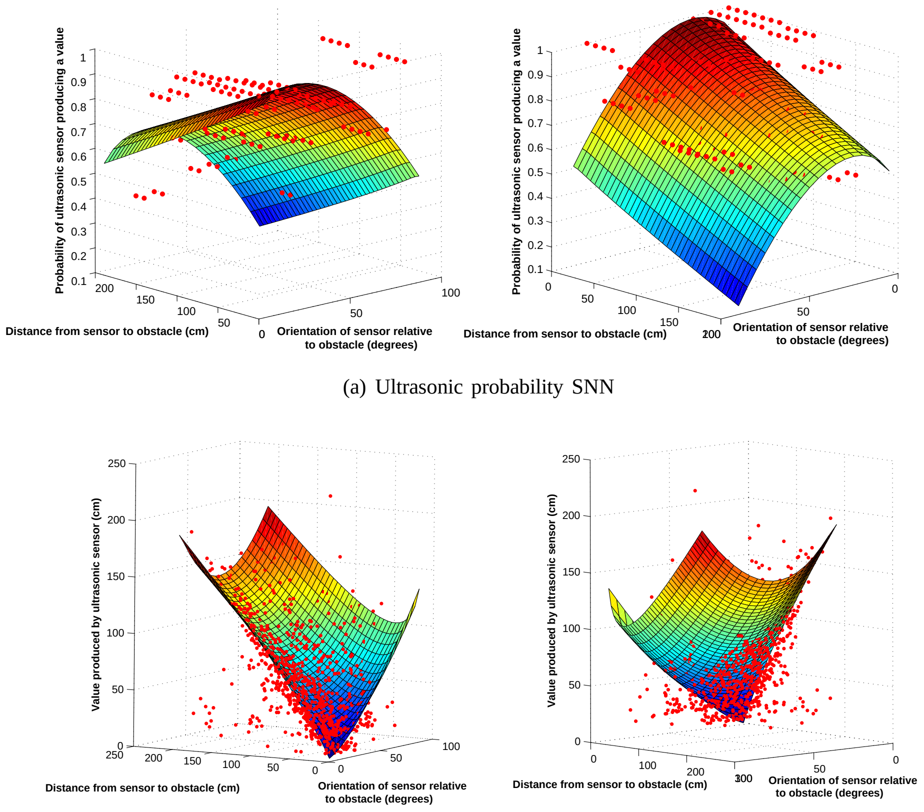
(a) Ultrasonic probability

(b) Ultrasonic value

Figure 2. Comparison of expected and SNN predicted values for ultrasonic sensor SNNs of the OAR

The noise in the ultrasonic sensor value SNN test data can easily be seen in Figure 3(b) (see discussion on this figure later in this section). Visible in Figure 2(b) is a vertical line where the expected value is roughly 20cm. This can be attributed to the fact that the ultrasonic sensor was sometimes seen to produce an erroneous value of roughly 20cm regardless of the actual distance of the sensor from an obstacle. Taking the noise levels present in training data for the ultrasonic sensor SNNs into account, the accuracy obtained by these SNNs is reasonable.

The three-dimensional plots shown in Figure 3(a) indicate that a relatively good fit was obtained for the ultrasonic probability SNN. The general trend predicted by this SNN is as would be expected: The probability of the ultrasonic sensor firing successfully (that is, the emitted pulse returning to the receiver of the ultrasonic sensor) is highest when the sensor is close to an obstacle and facing it head-on (that is, it has a small orientation angle). This probability decreases as the obstacle gets further away and the relative angle between



(a) Ultrasonic probability SNN

(b) Ultrasonic value SNN

Figure 3. Outputs for the SNNs for various values of distance and orientation (test data included)

the robot and the obstacle becomes bigger, since the emitted pulse would have a low probability of reaching the receiver of the ultrasonic sensor under these circumstances due to it being scattered.

High levels of noise can be seen in the test data for the ultrasonic sensor value SNN in Figure 3(b). These noise levels were almost certainly also present in the training data for this SNN, and are probably caused by inaccuracies in the sensor itself. The surface produced by the SNN does not match the test data very accurately, but this is probably due to the erroneous test data. A trend can clearly be seen in the noise level present in the test data. When the orientation angle is small (that is, the ultrasonic sensor faces the obstacle almost head-on) the noise levels are low and the value produced by the ultrasonic sensor is roughly equal to the real-world distance between the sensor and the obstacle as determined through motion tracking. As this orientation angle increases, however, more noise is gradually introduced in the test data with the value produced by the ultrasonic sensor diverging more and more from the distance determined by motion tracking. These large levels of noise from the training data of the ultrasonic value SNN almost certainly impacted negatively on the training accuracy of said SNN.

IX. Controller Evolution

The actual effectiveness of the SNNs in the ER process could, however, only be demonstrated by evolving a controller in simulation using the SNNs and consequently successfully transferring the controller to the real world. A simple array-based obstacle avoidance controller was consequently evolved in simulation using the SNNs. Four different obstacle avoidance tasks where set by choosing four different initial positions and orientations for the OAR in a scene containing four stationary obstacles (refer to Figure 4). The goal of each of the four evolved controllers was to guide the OAR through the scene to a predetermined target point by using inputs from the ultrasonic sensor.

The obstacle avoidance task was previously solved using an open-loop time-based controller which did not take any sensory inputs into account [11]. However, in the current study the controller was not allowed to transition to new motor speeds based on time and was thus forced to use input from the ultrasonic sensor. Ultimately, the effectiveness of the evolved controller would be entirely dependent on the accuracy of the real-world sensor and of the developed SNNs.

A command set containing 4-tuples of the form (mot_1, mot_2, $bigorsmall$, $ultval$) was evolved. A given pair of motor speeds (mot_1 and mot_2) was maintained until the ultrasonic sensor produced a value (the sensor would sometimes not produce a value at all) and this value was larger than or smaller than (as determined by a boolean value *bigorsmall*) a threshold value ($ultval$). The next pair of motor speeds in the next tuple would then be executed. This process was continued until the end of the command set was reached. Each controller contained six of these 4-tuples.

A. Evolution Procedure

A total of four controllers were evolved to guide the robot to the target position from each of the four positions from which the robot was started. These controllers were evolved entirely in simulation, using the motion and ultrasonic sensor SNNs. The different tuples of the controllers were directly encoded as individuals in the algorithm. Identical parameters were employed in the ER process as was used in the GA described in Section VII.

The fitness function (F_{obs}), used to quantify the quality of each potential solution, is given in Equation (3) [11].

$$F_{obs} = \begin{cases} \frac{0.5}{dist_{crash}} & \text{if the robot crashed} \\ \frac{1}{dist_{final}} & \text{otherwise} \end{cases} \tag{3}$$

The value $dist_{final}$ is calculated as the Euclidean distance from target point to the robot's final position. The value $dist_{crash}$, which is used only when the robot crashed into an obstacle, is the Euclidean distance from the target point to the impact point.

The fitness function thus assigned high fitness values when the robot stopped close to the target position without colliding with obstacles. Controllers that caused collisions close to the target position were favoured over those that caused collisions far from the target position.

The SNNs were used to produce a simulated path for each controller in the ER population. Each controller was assigned a fitness based on the simulated path, using equation (3). Evolution was terminated after 1000 generations.

Randomness was incorporated into the ultrasonic sensor SNNs in that the ultrasonic probability SNN predicts the probability of the ultrasonic sensor producing a value. As a result of the randomness present in the ultrasonic sensor simulation, any given controller in the ER population would thus produce different behaviours in simulation when run multiple times. In order to thus produce a robust controller which would take into account the firing and non-firing of the ultrasonic sensor, the fitness of each controller in the ER population was determined by running each controller five times in simulation and summing the fitnesses produced in each of these five runs. The task to be performed by a controller evolved in this way would thus be not only to reach the target position as accurately as possible, but also to do this in spite of the noise present in the ultrasonic sensor.

B. Results and Discussion

The quality of the evolved controllers were investigated by executing the controllers on the real-world robot. The successful execution of the obstacle avoidance task in the real world is of paramount importance, as this determines whether SNNs can be successfully used in the ER process.

Figure 4 compares the paths followed by the simulated robot and the real-world robot for each of the four starting points considered. Three paths are illustrated in each case. Motion tracking was used to determine the real-world paths of the robot (indicated by solid lines in Figure 4).

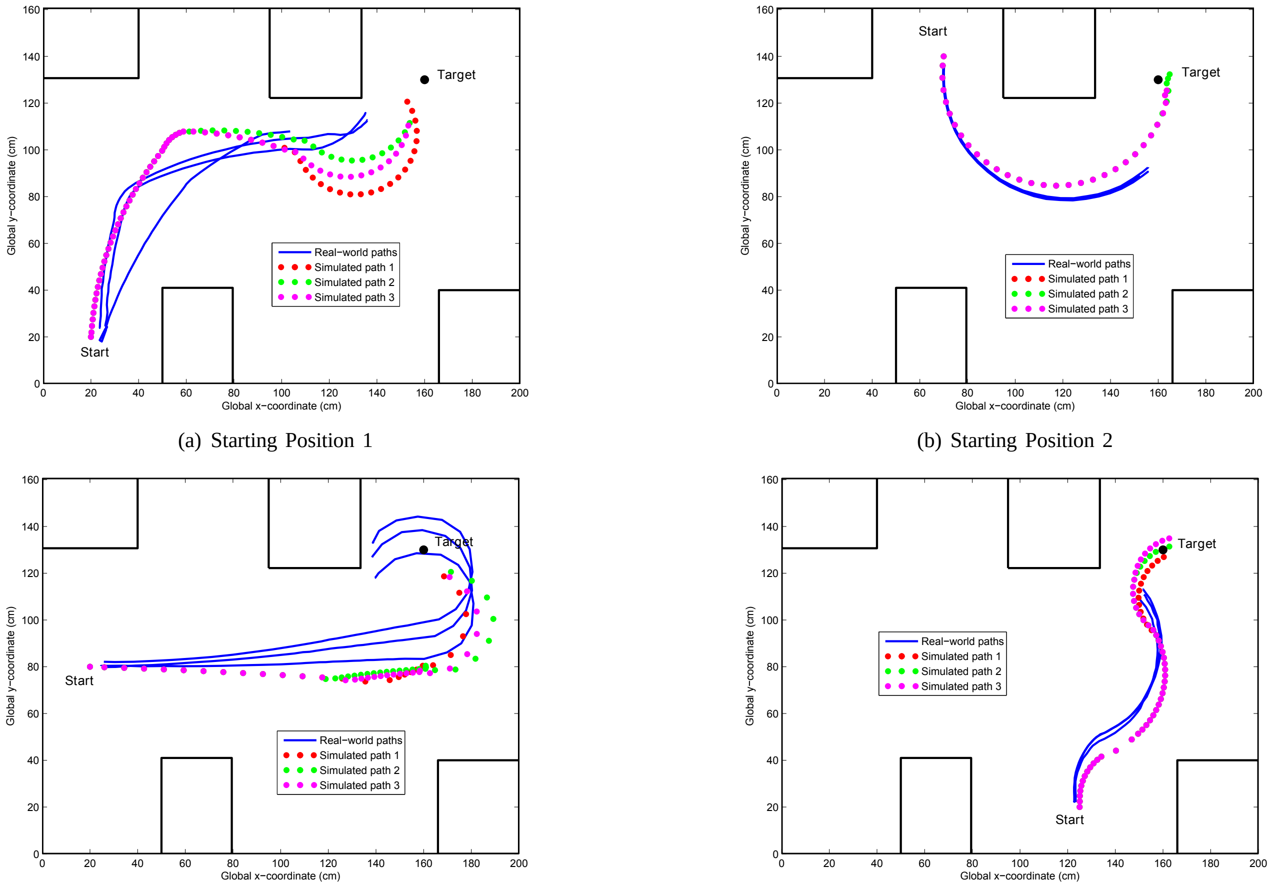
Results obtained for the ultrasonic sensor-based controllers show a relatively good correspondence between the simulated and real-world behaviours, although some discrepancies are evident. Notably, the simulated paths shown in Figure 4(a) can be seen to differ considerably from the real-world paths. This resulted from the robot crashing into an obstacle. In this case the SNNs thus failed to accurately model the robot's behaviour, although more accurate results were seen for the remaining three starting points.

Large amounts of noise present in the functioning of the ultrasonic sensor could have contributed to the differences between simulation and the real world. The fact that relatively consistent paths were observed for the evolved controllers in the real world from each starting position (roughly the same path was taken in all three real-world runs) indicates that the ER process succeeded in evolving controllers which could perform their task adequately. This is in spite of the ultrasonic sensor sometimes not producing a value and readings from this sensor generally containing large amounts of noise. The results thus indicate that the ultrasonic sensor SNNs (especially the ultrasonic probability SNN) represented the functioning of the ultrasonic sensor with reasonable accuracy.

X. Conclusion and Future Work

The main goal of this study was to determine whether SNNs can be successfully used to simulate an ultrasonic sensor in the ER process. The results of this study indicate that this could indeed be achieved, and provide evidence that the basic technique proposed in [9][10][11] can be extended to create simulators for more complex sensors.

Controllers were successfully evolved using the created SNNs. This suggests that, despite the limited and noisy training data, the SNNs managed to generalise. Readings from sensors will inevitably contain noise which results in unpredictable real-world behaviour. This is evident, for example, in the operation of the ultrasonic sensor. By creating the ultrasonic sensor probability SNN, a simple method has thus been suggested to model such unreliable sensors. Reasonable results were obtained using this method,



(a) Starting Position 1

(b) Starting Position 2

(c) Starting Position 3

(d) Starting Position 4

Figure 4. SNN predicted and real-world paths for the ultrasonic sensor-based controllers

in that controllers evolved in simulation using the ultrasonic probability SNN performed reasonably when executed on the real-world OAR. SNNs thus provide a method for modeling unreliable sensors.

A benefit of using SNNs as simulators is that their use do not require an in-depth analysis and understanding of the environment to be modeled and the underlying mechanics. This is a considerable advantage of the applied approach in comparison to more traditional approaches. Furthermore, the investigated approach has the advantage of implicitly incorporating the flaws and imperfections of the robotic hardware. It is not certain how well the applied approach will scale to other more complex robot systems. Future investigations can therefore explore the potential usage of SNNs to model other robotic sensors with large quantities of noise in their readings, or robots which need to function in highly dynamic environments.

References

[1] D. A. Sofge, M. A. Potter, M. D. Bugajska, and A. C. Schultz, “Challenges and opportunities of evolutionary robotics,” in *Proceedings of the Second International Conference on Computational Intelligence, Robotics and Autonomous Systems*, 2003.

[2] S. Nolfi and D. Parisi, “Evolving non-trivial behaviors on real robots: An autonomous robot that picks up objects,” in *Proceedings of the 4th Congress of the Italian Association for Artificial Intelligence on Topics in Artificial Intelligence*. London: Springer Verlag, 1995, pp. 243–254.

[3] I. Harvey, P. Husbands, and D. Cliff, “Issues in evolutionary robotics,” in *Proceedings of the Second International Conference on Simulation of Adaptive Behavior: From Animals to Animats 2*. Cambridge: MIT Press, 1993, pp. 364–373.

[4] R. A. Brooks, “New approaches to robotics,” *Science*, vol. 253, no. 5025, pp. 1227–1232, Sep 1991.

[5] W. Van de Velde, “Toward learning robots,” *Robotics and Autonomous Systems*, vol. 8, no. 1-2, pp. 1–6, 1991.

[6] L. A. Meeden and D. Kumar, “Trends in evolutionary robotics,” in *Soft Computing for Intelligent Robotic Systems*, L. Jain and T. Fukuda, Eds. New York: Physica-Verlag, 1998, pp. 215–233.

ISBN: 978-1-61208-226-4

[7] D. Floreano, P. Husbands, and S. Nolfi, "Evolutionary Robotics," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Berlin: Springer Verlag, 2008.

[8] D. Floreano and F. Mondada, "Automatic creation of an autonomous agent: Genetic evolution of a neural-network driven robot," in *Proceedings of the Third International Conference on Simulation of Adaptive Behavior: From Animals to Animats 3*. Cambridge: MIT Press, 1994, pp. 421–430.

[9] C. J. Pretorius, M. C. du Plessis, and C. B. Cilliers, "Towards an artificial neural network-based simulator for behavioural evolution in evolutionary robotics," in *Proceedings of the 2009 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists*. New York: ACM, 2009, pp. 170–178.

[10] ——, "A neural network-based kinematic and light-perception simulator for simple robotic evolution," in *IEEE Congress on Evolutionary Computation*, 2010, pp. 1–8.

[11] ——, "Simulating robots without conventional physics: A neural network approach," *submitted to Journal of Intelligent and Robotic Systems*, 2012.

[12] H. H. Lund and O. Miglino, "From simulated to real robots," in *Proceedings of IEEE Third International Conference on Evolutionary Computation*, 1996.

[13] N. Jacobi, P. Husbands, and I. Harvey, "Noise and the reality gap: The use of simulation in evolutionary robotics," in *Proceedings of the Third European Conference on Advances in Artificial Life*. London: Springer Verlag, 1995, pp. 704–720.

[14] O. Miglino, H. H. Lund, and S. Nolfi, "Evolving mobile robots in simulated and real environments," *Artificial Life*, vol. 2, pp. 417–434, 1996.

[15] O. Miglino, K. Nafasi, and C. E. Taylor, "Selection for wandering behavior in a small robot," *Artificial Life*, vol. 2, no. 1, pp. 101–116, 1995.

[16] R. A. Brooks, "Artificial life and real robots," in *Proceedings of the First European Conference on Artificial Life*. Cambridge: MIT Press, 1992, pp. 3–10.

[17] J. Teo, "Robustness of artificially evolved robots: What's beyond the evolutionary window?" in *Proceedings of the Second International Conference on Artificial Intelligence in Engineering and Technology*, Kota Kinabalu, Sabah, Malaysia, 2004, pp. 14–20.

[18] V. Tikhanoff, A. Cangelosi, P. Fitzpatrick, G. Metta, L. Natale, and F. Nori, "An open-source simulator for cognitive robotics research: The prototype of the iCub humanoid robot simulator," in *Performance Metrics for Intelligent Systems Workshop, National Institute of Standards and Technology*, 2008.

[19] K. M. A. Chai, C. K. I. Williams, S. Klanke, and S. Vijayakumar, "Multi-task Gaussian process learning of robot inverse dynamics," in *Advances in Neural Information Processing Systems 21, Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems*, 2008, pp. 265–272.

[20] T. Kyriacou, U. Nehmzow, R. Iglesias, and S. A. Billings, "Accurate robot simulation through system identification," *Robotics and Autonomous Systems*, vol. 56, no. 12, pp. 1082–1093, 2008.

[21] S. Carpin, T. Stoyanov, and Y. Nevatia, "Quantitative assessments of USARSim accuracy," in *Proceedings of Performance Metrics for Intelligent Systems Workshop*, 2006.

[22] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *In IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004, pp. 2149–2154.

[23] J. Rieffel, F. Saunders, S. Nadimpalli, H. Zhou, S. Hassoun, J. Rife, and B. Trimmer, "Evolving soft robotic locomotion in PhysX," in *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference*. New York: ACM, 2009, pp. 2499–2504.

[24] I. A. Basheer and M. Hajmeer, "Artificial neural networks: Fundamentals, computing, design, and application," *Journal of Microbiological Methods*, vol. 43, no. 1, pp. 3–31, 2000.

[25] S. I. Gallant, *Neural Network Learning and Expert Systems*. Cambridge: MIT Press, 1993.

[26] S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. New Jersey: Prentice Hall, 2008.

[27] R. Grzeszczuk, D. Terzopoulos, and G. Hinton, "Neuroanimator: Fast neural network emulation and control of physics-based models," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. New York: ACM, 1998, pp. 9–20.

[28] P. O. Moreno, S. I. Hernandez Ruiz, and J. C. R. Valenzuela, "Simulation and animation of a 2 degree of freedom planar robot arm based on neural networks," in *Proceedings of the Electronics, Robotics and Automotive Mechanics Conference*. Washington, DC: IEEE Computer Society, 2007, pp. 488–493.

[29] T. Lee, U. Nehmzow, and R. J. Hubbold, "Mobile robot simulation by means of acquired neural network models," in *Proceedings of the 12th European Simulation Multiconference on Simulation - Past, Present and Future*, 1998, pp. 465–469.

[30] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," in *Proceedings of the 2000 Congress on Evolutionary Computation*, 2000, pp. 256–263.

[31] J. Kodjabachian and J. Meyer, "Evolution and development of neural controllers for locomotion, gradient-following, and obstacle-avoidance in artificial insects," *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 796–812, 1998.

[32] "LEGO.com MINDSTORMS : Home," retrieved: August, 2012. [Online]. Available: mindstorms.lego.com

[33] K. Deb and R. Agrawal, "Simulated binary crossover for continuous space," in *Complex Systems*, 1995, pp. 115–148.

[34] C. J. Pretorius, "Artificial neural networks as simulators for behavioural evolution in evolutionary robotics," Master's thesis, Nelson Mandela Metropolitan University, 2010.

Biometric Security Systems for Mobile Devices based on Fingerprint Recognition Algorithm

Michał Szczepanik, Ireneusz Jóźwiak
Institute of Informatics,
Wrocław University
of Technologies
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
Email: {michal.szczepanik, ireneusz.jozwiak}@pwr.wroc.pl

***Abstract*—In this paper, we propose a selective attention algorithm which increases the reliability of biometrics security system based on fingerprint recognition. We compare the existing fingerprint recognition algorithms and test our own algorithm on fingerprints database which changes in the structure as a result of physical damage. We propose new selective attention algorithms, which help to detect the most sensitive to damage areas, and add it as step of fingerprint analyses for the fingerprint recognition procedures. We also propose a new algorithm, which does not require complex hardware systems, so it can be applied in new smart mobile devices, which restrict unauthorized access to sensitive data or other user resources. The main goal of this work is to demonstrate the applicability of the developed algorithm in mobile devices.**

Keywords-biometric; fingerprint; minutia group; selective attention algorithms.

I. Introduction

Nowadays, mobile phones, tablet PCs and other mobile devices are an excellent source of data about users. They are used as a remote office, a tool for bank account management, email management, as well as entertainment like social media, real time social games, etc. The only applicable security method is a four-digit pin, which is usually easy to guess or break. Computational capabilities of current devices are not as limited as two years before [17]. Today, mobile phones have the computational capabilities similar to personal computers which could be bought one or two years ago; so, why should they not use better security systems, such as biometrics systems. The accuracy of a fingerprint verification system is critical in a wide variety of civilian, forensic and commercial applications such as access control systems, credit cards, phone usage, etc. The main problem, from an economic point of view, can be the size of the used system hardware and its cost, and therefore it cannot be too extensive and advanced. Fingerprints are the most widely used biometric feature for personal identification and verification in the field of biometric identification [11][19]. Most important for designing the system is the effectiveness of fingerprint recognition algorithms, which depends mainly on the quality of digital fingerprint images input and fingerprint's physical damage [9]. Current mobile devices typically use as collateral for a four digit pin code or face recognition system, which doesnt work when is too darker. Most fingerprint recognition algorithms are not immune to damage, so they are not use in mobile devices. The main problem, which we would like to solv,e is the stability of the recognition systems with respect to the capability to deal with fingers damage, will make the system more useful to the user.

In Section II, we explain the basic parameters and and measures for the safety and usability of biometric systems. In Section III, we briefly analyze existing algorithms and how they work. In Sections IV and V, we present preprocessing and method for comparing fingerprints by our algorithm. In the section The quality of the algorithms, we present first test of our algorithm on public fingerprints database. In the next section, we explain how we represent data for our algorithm. The most important section of this paper is The experiment, in which we present results of tests, which are done on real mobile devices.

II. Quality assessment of biometric algorithms

There are two most important performance metrics for biometric systems [17]: False Accept Rate (FAR), also called False Match Rate (FMR), is the probability that the system incorrectly matches the input pattern to a non-matching template from the database. It measures the percent of invalid inputs which are incorrectly accepted. False Reject Rate (FRR), also called False Non-Match Rate (FNMR), is the probability that the system fails to detect a match between the input pattern and a matching template from the database. It measures the percent of valid inputs which are incorrectly rejected. They can be presented mathematically as:

$$FAR(T) = \int_{Th}^{1} p_i(x)dx \tag{1}$$

$$FRR(T) = \int_{0}^{Th} p_i(x)dx \tag{2}$$

ISBN: 978-1-61208-226-4

where Th is the value of a threshold used in the the algorithm. Both FAR and FRR are functions of a threshold T. When T decreases, the system has more tolerance to intraclass variations and noise; however, FAR increases. Similarly, if t is lower, the system is more secure and FRR decreases.

III. History and existing solutions

First mobile phone with fingerprint recognition system was developed by Siemens in 1998 [19]. Since that time, more than 100 phone models had such a protection [15]. Unfortunately, the biggest problem of those systems was usability. Every day, people are exposed to cuts, wounds and burns; therefore, it is important that the algorithms are resistant to this type of damage. The current fingerprint recognition systems for mobile devices usually use one of the algorithms: Minutiae Adjacency Graph (MAG), Elastic minutiae matching (EMM), Delaunay Triangulation (DT), Pattern-Based Templates (PBT) [10]. The most popular algorithms are based on local and global structures represented by graphs like in MAG [16]. In this type of algorithms, local structures to find corresponding points to align feature vector are used first, then global structures are matched [4][5]. This type of algorithm was used by He and Ou [6], Ross et al. [16]. They also use thin-plate spline (TPS) model to build an average deformation model from multiple impressions of the same finger. Owing to iteratively aligning minutiae between input and template impressions, a risk of forcing an alignment between impressions originating from two different fingers arises and leads to a higher false accept rate. Typically, a minutia matching has two steps:

- Registration aligns fingerprints, which could be matched, as well as possible.
- Evaluation calculates matching scores using a tolerance box between every possibly matched point (minutiae) pairs.

The EMM algorithm typically uses only global matching where each point (minutia) which has a type, like end point or bifurcation, needs to be matched to a related point in the second fingerprint image. Based on elastic deformations which are used to tolerate minutiae pairs that are further apart because of plastic disrotations, and therefore to decrease the False Rejection Rate, so in most popular algorithms authors increase the size of bounding boxes [13] to reduce this problem, but as side effect they got higher False Acceptation Rate (FAR). In this type of algorithms, for elastic match, TSP [1] also can be used, which provides better performance than only one parameter of deformation.

The DT algorithm [2][14] is the most popular version of MAG, so it was tested as separate algorithm. Its structure based on triangulation connects neighboring minutiae to create triangles, such that no point (minutia) in P is inside the circumcircle of any triangle in DT(P). DT algorithm analyzes the structure of points identically as minutiae adjacency graph algorithm, so it also is not resistant to typical injury of physical fingerprint (see Figure 1).

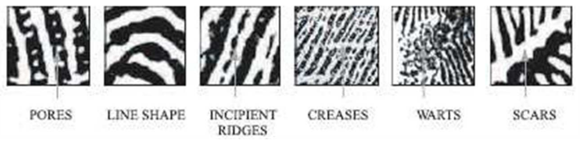


Figure 1. Typical damages on fingerprint

The Pattern based algorithms [3] compare the basic fingerprint patterns (like arch, whorl, and loop between a previously stored template and a candidate fingerprint. This algorithm requires that the images be aligned in the same orientation and in the same scale. To do this, the algorithm finds a central point in the fingerprint image and centers on it, and after that, scales to the same size of the fingerprints ridge. In a pattern-based algorithm, the template contains the type, size and orientation of patterns within the aligned fingerprint image. The candidate fingerprint image is graphically compared with the template to determine the degree to which they match. Due to the storage of the original picture for the algorithm there is a high risk that this image can be read from the memory card reader or fingerprints database.

IV. Fingerprint recognition algorithm based on minutia' groups

The proposed solutions, in contrast to other algorithms, are more resistant to damage.

A. Fingerprint recognition algorithm based on minutes groups

For older low-resolution readers it is required to detect the areas of correct scanning of the fingerprint. First step of image analysis is the search for the imprint area including the exclusion of areas containing significant damage [18]. Fingerprint image is represented by a gray scale image that defines the area of forced application fingerprint for the reader (see Figure 2).

$$I_{fp}(i,j) = < 1, 255 > \tag{3}$$

The operation that converts a grayscale image into a binary image is known as binarization. We carried out the binarization process using adaptive thresholding. Each pixel is assigned a new value (1 or 0) according to intensity mean in a local area and the parameter t_g which excludes poorly read fingerprint areas from the analysis (see Figure 3).

$$B_{fp}(i,j) = \begin{cases} 1 for I_f p(i,j) \geqslant t_g \\ 0 for I_f p(i,j) < t_g \end{cases} \tag{4}$$

The last step is creating the fingerprint mask based on the binarized image. The Mask for the area of a square (X, Y), which size is 2.5 wide edges, is determined by two

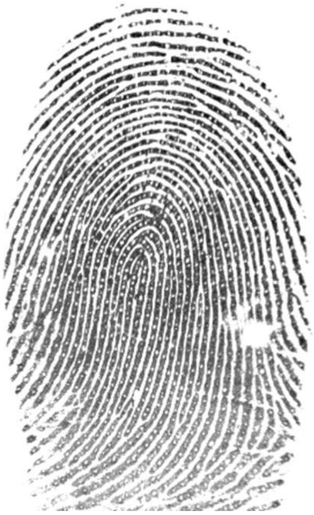


Figure 2. Original image (Source: own work)

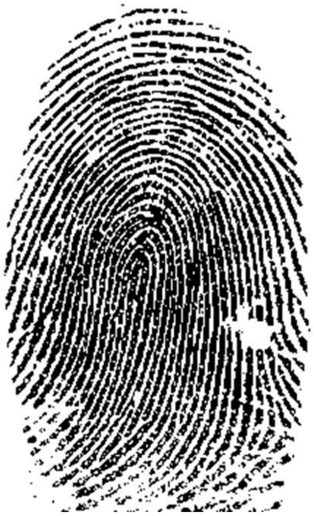


Figure 3. Image after binarization (Source: own work)

parameters p_{lo}, which is a limitation that excludes areas with an insufficient number of pixels describing the image, and p_{hi} excludes blurred areas, such as moist (see Figure 4).

$$F_fp(X,Y) = \begin{cases} I_fp(X,Y) for F_{img} \geqslant p_{lo} \wedge F_{img} \leqslant p_{hi} \\ 0 \; otherwise \end{cases} \tag{5}$$

Where F_{img} is

$$F_{img} = \sum_{i \in X} \sum_{j \in Y} Bfp(i,j) \tag{6}$$

Created mask is used for finding the most damaged area in the fingerprint image.

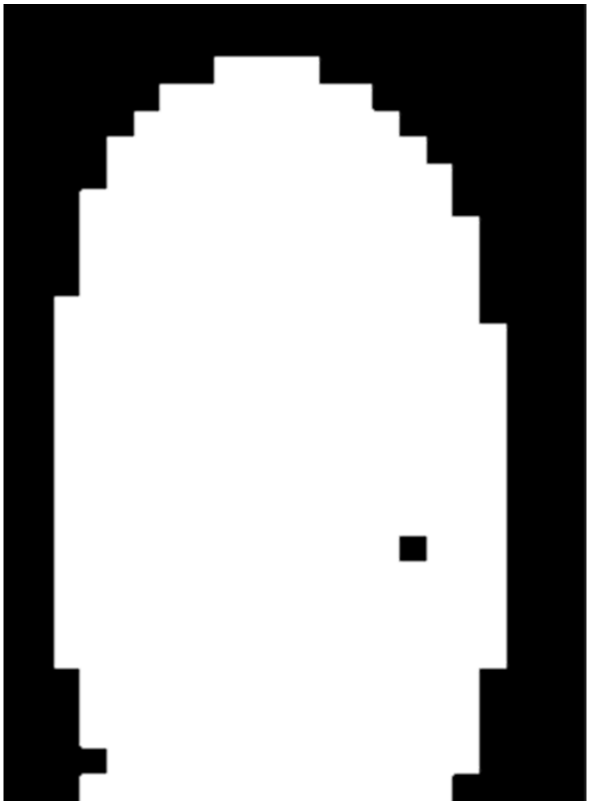


Figure 4. Mask for fingerprint with detecting damage (Source: own work)

B. Detecting features and leveling of the damage in segmentations

Standard leveling of damage is carried out by calculating the variance of points and the analysis of brightness. Based on these two parameters, the frequency of furrows is calculated, which is used for each fingerprint image. After applying Gabor filter [12] to highlight the pits and valleys, it uses segmentation in accordance with its size, 2.5 width of segment furrow, the image is redrawn. After that process fingerprints are continuous and lint. In contrast to the literature, the algorithm does not require additional transformations to find the minutiae, such as converting the width of 1px furrows. It does not require information about the orientation of minutiae; it only requires the data about its position. Therefore, the resulting image is used to find the edge - the minutiae are located at the intersection of the edge of the furrows. The problem of fingerprint recognition is a complex process, even in laboratory conditions; therefore, if used as a system to control access to the mobile devices, it should be insensitive to certain natural changes or damages in physical structure of fingerprints, which can include: incomplete fingerprint, fingerprint parts which can be injured or burned, blurred, partly unreadable or rotation. In order to detect the most sensitive to damage areas, we use neural network with selective attention technique [7]. This type of

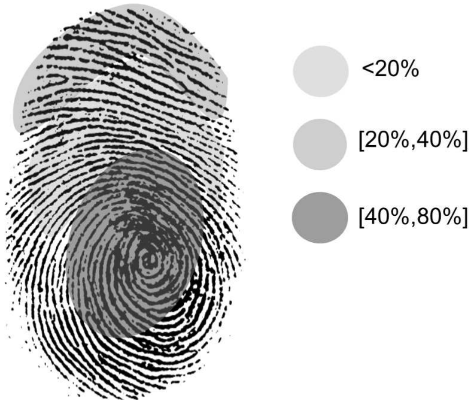


Figure 5. Mask of fingerprint's areas vulnerable to damage. (Source: own work)

neural network is more like an analysis done by a human [8]. This allows us to create a mask of areas vulnerable to damage (see Figure 5).

We created 15 different masks, broken down by the type of fingerprints core also known as fingerprint patterns (arch, whorl and loop) and the type of finger (thumb, index finger, middle finger, ring finger, small finger). Basing on this mask we created a filter, which we use to compare fingerprints where specific minutiae are weighted in the decision process and their score is based on the location on the fingerprints.

V. Comparison of fingerprints

Minutiae image is divided into segments, each segment is corresponding minutiaes group is described by parameters (x, y, nom), where x and y are the coordinates, and nom determines the number of minutiae in the group. Additionally, one implementation uses an additional parameter specifying the probabilities of damage in a given segment, which is estimated by a neural network, based on the distribution of areas rejected by the mask described by the formula. Current algorithm implementation searches small groups of minutiae that that contain up to 5 minutiae (see Figure 6). Then, based on the neighboring groups (max 4) creates a new large group (see Figure 7). For each, the orientation parameters and the number of characteristic points are recalculated. The last step is to create a matrix of Euclidean distances between the largest groups.

When comparing the use of two parameters: dx - the distance defining the difference between groups in the pattern and tested fingerprint, px - the threshold of damage occurrence probability (determined by whether the group is under consideration in the analysis), we decide which groups should be compered and we set the priority for them. After that we do the comparison of the groups, which are divided according to the priority, that is defined by the number of minutiae in the group and selective attention (SA) algorithms, which are based on probabilities of damage in a group segment. This provides quick verification of whether the analyzed fingerprint is consistent with the pattern.

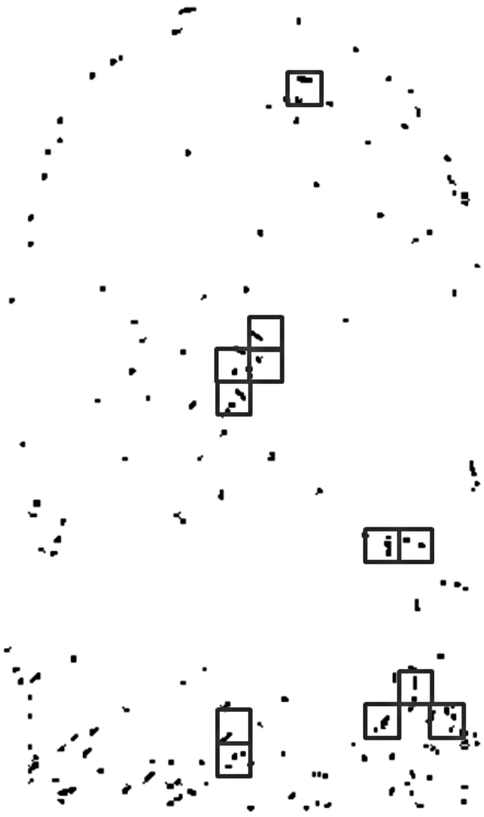


Figure 6. Fingerprint devided into segments. (Source: own work)

Table I
The result of experiment using FVC2004 database

	FAR	FRR
MAG	0.82%	0.65%
EMM	1.23%	1.15%
PBTA	0.15%	1.73%
MGM64	6.60%	0.54%
MGM32	3.23%	0.32%
MGM32_SA	0.38%	0.09%

VI. The quality of the algorithms

First test was done using FVC2004 [12] fingerprint databases. For each of four databases, a total of 120 fingers and 12 impressions per finger (1440 impressions) were gathered. Unfortunately, most of the publicly available databases of fingerprints do not include the problem of physical damage, so additionally small damage such as cuts and burns has been generated on each sample. In most cases artificially applied damages cover 5-20% of the fingerprint. For 10% of the samples they cover approximately 50% of the area to simulate severe damage.

Most algorithms cannot process fingerprints with severe

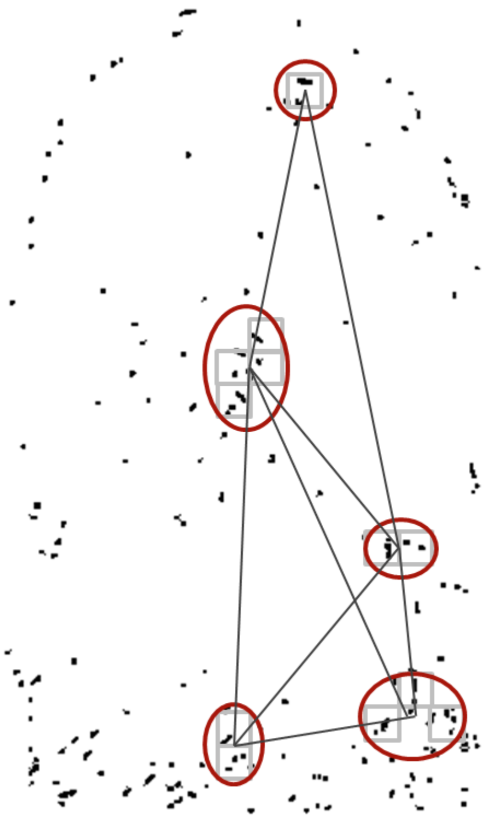


Figure 7. Detecting relations between minutiaes groups. (Source: own work)

physical damage correctly. Also, the proposed one has proven to have a very dangerous level of False Acceptation Rate. After applying the selective attention algorithm, fingerprint recognition algorithm improved its performance and reliability. The proposed algorithm has been developed in such a way, that it uses the property of a damage map, so its results have improved the most.

VII. Classification and data managment used in the algorithm

The developed algorithm is based on minutiae groups, where each group is basically represented by the coordinates - x, y and the number of minutiae - nom contained in the group. Group covers an area equal to 2.5 the width of the furrow and its coordinates are in the middle of the square which is boundaring this area. Number of minutiae in the group describes its priority. Additionally, a stored parameter defines the probability of damage - p_d in the area represented by the group. In conclusion the group is defined as follows:

$$M_{group} : \{x, y, nom, p_d\} \tag{7}$$

Based on these data, a matrix of Euclidean distances between the groups is created. Data on the characteristic point is limited to its weight (nom) and the probability of damage p_d. Finally we obtain:

$$M_{group}(I) : \{nom_I, (p_d)_I)\} \tag{8}$$

$$M_{group}(I, J) : dist(M_{group}(I), M_{group}(J)) \tag{9}$$

where $dist(M_{group}(I), M_{group}(J))$ is Euclidean distances between the group I and J. Data stored for analysis to prevent reproduction of the original fingerprint image. Additional storage parameters to estimate the damage allow us to better match fingerprints in the event of damage.

Table II
The result of experiment on Samsung Galaxy SII

	FAR	FRR	Average time to decision in ms
MAG	0.80%	0.63%	138
EMM	1.15%	1.20%	127
PBTA	0.25%	1.70%	130
MGM64	6.75%	0.55%	127
MGM32	3.43%	0.42%	128
MGM32_SA	0.43%	0.18%	132

Table III
The result of experiment on HTC Widefire S

	FAR	FRR	Average time to decision in ms
MAG	0.80%	0.63%	225
EMM	1.15%	1.20%	220
PBTA	0.25%	1.70%	232
MGM64	6.75%	0.55%	215
MGM32	3.43%	0.42%	228
MGM32_SA	0.43%	0.18%	225

VIII. The experiment

The tests were conducted on two devices: Samsung Galaxy SII (see Table II) and HTC Widefire S (see Table III). For test authors used real fingerprints. Due to the nature of work, we use real fingerprints. Each user was exposed to frequent damage of fingerprints, like cuts and burns presented on Figure 1. All algorithms were compared using 112 different fingerprints and each had 10 samples.

In the test, we use not optimal algorithm, because first implementation was done in Android SDK, not in NDK, which allow developer to use more code optimizations, but need much more time to implement it for specific devices. The implementation in Android NDK is planned in future work. This implementation should also allow us to create a dynamic mask of the damaged sectors and the most vulnerable to damage area of the fingerprint for a specific user and not only use a general mask, which is hardcoded in our current implementation to reduce memory usage. Our algorithm can be used on mobile devices because its decision time is very similar to other algorithms; however, other parameters are much better.

IX. Conclusion

In this paper, we proposed a new step for fingerprint-matching approach, which is based on selective attention. Inserted mask can be hardcoded in the algorithm or generated in real time by neural network, but it required devices with better performance. With a hardcoded mask we can provide significant improvement in algorithms we with a low performance cost. The proposed solution can be used by everyone who is exposed to damage of fingerprints. The system can also be applied to protect access to important data or premises, which are very important for mobile device users.

Acknowledgment

Experiment is co-financed by the European Union under the European Social Fund.

References

[1] A.M. Bazen and S.H Gerez, ”Fingerprint Matching by Thin-plate Spline Modelling of Elastic Deformations.”, in Pattern recognition, vol. 36 (8), 2003, pp. 1859-1867

[2] G. Bebis , T. Deaconu, and M. Georgiopoulos, ”Fingerprint Identification Using Delaunay Triangulation”, in Proc. IEEE International Conference on Intelligence, Information, and Systems, 1999, pp. 452-459

[3] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, ”Fingerprint Classification by Directional Image Partitioning”, in IEEE Transactions on Pattern Analysis Machine Intelligence, vol.21, no.5, 1999 pp. 402-421

[4] S. Chikkerur, V. Govindaraju, and E. N. Cartwright, ”K-plet and coupled bfs: A graph based fingerprint representation and matching algorithm.”, in LNCS vol. 3832, 2006, pp. 309315

[5] C. Grzeszyk, ”Forensic fingerprint examination marks.” (in Polish), Wydawnictwo Centrum Szkolenia Policji, Legionowo 1992

[6] Y. He, Z. Ou, ”Fingerprint matching algorithm based on local minutiae adjacency graph”, in Journal of Harbin Institute of Technology 10/05, 2005, pp. 95-103

[7] M. Huk, ”Sigma-if neural network as the use of selective attention technique in classification and knowledge discovery problems solving”, in Annales Universitatis Mariae Curie-Skodowska. AI Informatica. vol. 5, 2006, pp. 121-131

[8] M. Huk, ”Learning Distributed Selective Attention Strategies with the Sigma-if Neural Network”, in Advances in computer science and IT / ed. by D. M. Akbar Hussain. Vukovar : In-Teh, 2009. pp. 209-232

[9] A. Hicklin, C. Watson, and B. Ulery, ”How many people have fingerprints that are hard to match”, NIST Interagency Report 7271 ,2005

[10] L. Hong, Y. Wan, and A. K. Jain, ”Fingerprint image enhancement: Algorithm and performance evaluation” in IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 1998, pp 777-789.

[11] A. K. Jain, A. Ross, and K. Nandakumar, ”Introducing to biometrics”, Spinger 2011

[12] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, ”Handbook of Fingerprint Recognition, 2nd Edition”, Springer 2009

[13] S. Pankanti, S. Prabhakar, and A.K. Jain, ”On the individuality of fingerprints”, in Proceedings of Computer Vision and Pattern Recognition (CVPR), 2001, pp. 805-812

[14] G. Parziale and A. Niel ”A fingerprint matching using minutiae triangulation.”, In: Proc. of International Conference on Biometric Authentication (ICBA) , Springer, vol. 3072, 2004, pp. 241-248

[15] N. K. Ratha and V. Govindaraju, ”Advances in Biometrics: Sensors, Algorithms and Systems”, Springer 2007

[16] A. Ross, S.C. Dass, and A.K. Jain, ”A deformable model for fingerprint matching”, in Pattern Recognition 38(1), 2005, pp. 95103

[17] A. Ross, K. Nandakumar, and A.K. Jain, ”Handbook of Multibiometrics (International Series on Biometrics)”, Springer 2011

[18] M. Szczepanik and R. Szewczyk, ”Fingerprint identification algorithm (in Polish)”. KNS, vol. 1, 2008, pp. 131 -136

[19] J.L. Wayman, A.K. Jain , D. Maltoni, and D. Maio, ”Biometric Systems. Technology, Design and Performance Evaluation, 1st Edition.”, Springer 2005

3D Measures Computed in Monocular Camera System for Fall Detection

Konstantinos Makantasis, Anastasios Doulamis, Nikolaos F. Matsatsinis
Computer Vision & Decision Support Laboratory,
Technical University of Crete,
73100, Chania, Greece
konst.makantasis@gmail.com, adoulam@cs.ntua.gr, nikos@ergasya.tuc.gr

Abstract—**Traumas resulting from falls have been reported as the second most common cause of death. For this reason, computer vision tools can be exploited for detecting humans' fall incidents. In this paper, we propose a fast, real-time computer vision algorithm capable to detect humans' falls in complex dynamically changing conditions, by exploiting the motion information in the scene and 3D space's measures. This algorithm is using a single monocular low cost camera and it requires minimal computational cost and minimal memory requirements that make it suitable for large scale implementations in clinical institutes and home environments. The proposed scheme was tested in complex and dynamically changing visual conditions and as proved by the experiments it has the capability to detect over 92% of fall incidents.**

Keywords-machine vision; image motion analysis; features extraction; subtraction techniques

I. INTRODUCTION

Life expectancy in developed countries is increasing and population is aging. However, the quality of life, especially for elderly, is associated with their ability to live independently and with dignity, without having the need to be attached to any person in order to live a normal life and fulfill daily living. According to medical records, falls are the leading cause of injury-related visits to emergency departments and the primary etiology of accidental deaths in persons over the age of 65 years. The mortality rate for falls increases dramatically with age in both sexes and in all racial and ethnic groups, making this one of the most important problems that hinders these people's ability to have such an independent life, making necessary the presence and monitoring of their daily activities by care-givers.

For this reason, a major research effort has been conducted in the recent years for automatically detecting persons' falls. One common way is through the use of specialized sensors, such as accelerometers, floor vibration sensors, barometric pressure sensors, gyroscope sensors, or combination/fusion of them [1][2][3][4][5]. However, most of the previous techniques require the use of specialized wearable devices that should be attached to human body and thus their efficiency relies on the person's ability and willingness to wear them. On the other hand, a more research challenging alternative is the use of visual cameras, which is however, a prime research issue due to the complexity of visual content (illumination variations, background changes and clutter) and the fact that a fall should be discriminated than other ordinary humans' activities, like sitting and bending. Vision-based systems present several advantages as they are less intrusive, installed on building (not worn by users), they are able to detect multiple events simultaneously and the recorded video can be used for post verification analysis. Towards this direction, some works exploit 2D image data like for instance [6][7][8][9]. These works exploit foreground object's shape as well as its vertical motion velocity in order to detect a fall incident. H. Qian *et al.* [10] are based on human anatomy according which each part of the human body occupies an almost fixed percentage in length relative to body height, in order to train a classifier capable six indoor human activities, including fall incidents. However, none of these works exploit 3D information to increase system robustness. A 3D active vision system based on Time of Flight (ToF) cameras is proposed in [11]. Although, this work doesn't take into account the orientation of motion of the moving blob, and the measures that are provided by the camera could be affected by reflectivity objects properties and aliasing effects when the camera-target distance overcomes the non-ambiguity range. Multi-camera systems have been also proposed in [12], to exploit stereo vision. 3D processing, though more robust than a 2D image analysis in terms of fall detection and discrimination of a fall than other daily humans' activities; require high computational cost making these systems unsuitable for real-time large scale implementations.

In this paper, a new innovative approach is presented that exploits, on the one hand, monocular cameras to detect in real-time fall incidents in complex dynamically changing visual conditions and, on the other, it is capable to exploit actual 3D physical space's measures, through camera calibration and inverse perspective mapping, to increase system robustness. Due to its minimum computational cost and minimum memory requirements, it is suitable for large scale implementations, let alone its low financial cost since simple ordinary low-resolution cameras are used, making it

ISBN: 978-1-61208-226-4

affordable for a large scale. In contrast to other 2D fall detection methods [6][7][8], our system is very robust for wider range of camera positions and mountings, as is proven by the experiments.

The rest of this paper is organized as follows: in Section 2 problem formulation is presented. Section 3 presents 2D and 3D measures for features extraction. In Section 4 experimental results along with the fall detection algorithm are presented, and, finally, Section 5 concludes this work.

II. Approach Overview

Humans' fall incidents can be characterized by motion features that are very discriminative in the fall detection context and in humans' posture. Information about humans' posture can be derived by the actual width-height ratio, and it is valid that in a 3D space this ratio is bigger in value when a fall event occurs than the same ratio with humans in standing position. The most commonly used feature to detect a fall is that of vertical motion velocity, which, besides fall incidents discrimination, is also able to provide useful information about fall intensity and thus possible injuries. Vertical motion velocity V, during a sequence of frames, can be expressed by (1).

$$V = \sum_{i=k-m}^{k} h_a(i) - h_a(i-1) \qquad (1)$$

where $h_a(k)$ stands for the actual height of a human in 3D space at the k^{th} image frame (time). Vertical motion velocity is calculated over a time window of length m to estimate the speed of the motion which is also an evident of how severe would be a fall. Index k denotes the current frame for processing. We choose to use actual humans' height, measured in physical world units (e.g., cm, inches), and not their projected height being measured in pixel units, since this yields a more robust performance not be affected by cases where the human is far away or very close to the camera. To measure the actual height, however, we need to exploit 3D information. In addition, actual height can provide information about the moving object, in a way that the system becomes capable to discriminate if the moving object might be a human or something else, like a pet.

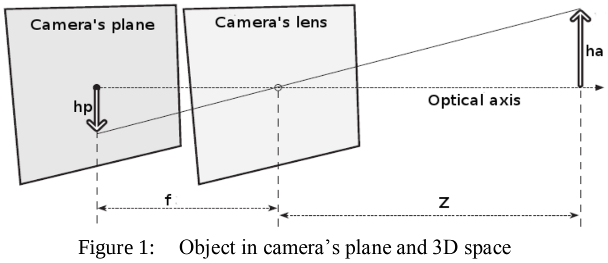


Figure 1: Object in camera's plane and 3D space

Width-height ratio computation requires, firstly foreground extraction (Section III.A), to extract the foreground object, which initially is unknown and secondly information about its left-most, right-most, top-most and bottom-most points, to calculate its projected height and projected width, as explained in Section III.B.

Vertical motion velocity, V , computation requires knowledge of the actual height of foreground object in 3D space. Representation of an object in camera's plane is presented in Figure 1. From Figure 1, it appears that the actual height of foreground object can be given through (2), if camera's focal length f, distance Z between the camera and foreground object and foreground object's projected height h_p are known.

$$h_a = Z\frac{h_p}{f} \qquad (2)$$

The projected height can be obtained by the use of a foreground detection algorithm (Section III.B), the focal length can be obtained through camera calibration, as this process provides information about camera's geometry, and the distance between the camera and the foreground object can be obtained through the construction of a reference plane that is the orthographic view of the floor, as explained in Section III.C.

III. 3D Measures For Fall Detection

This Section presents 2D and 3D measures used for features extraction, as well as, the fall detection algorithm.

A. Foreground Extraction

For foreground extraction we use the iterative scene learning algorithm described in [8]. This algorithm, unlike the classic background subtraction techniques, which fail in large scale implementations because of their computational cost and memory requirements, is computationally efficient and has the ability to operate properly in real-time and in complex, dynamic in terms of background visual content, and unexpected environments.

It exploits the intensity of motion vectors along with their directions to identify humans' movements. For motion vectors estimation, the "pyramidal" Lucas-Kanade algorithm [13] was used, which has the ability to catch large motions by using an image pyramid. Motion vectors estimation is followed by the creation of a binary mask in order to indicate areas of high motion information.

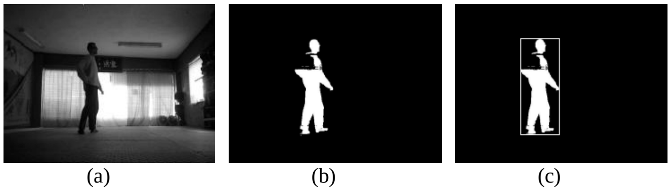

(a) (b) (c)

Figure 2: (a) original frame, (b) extracted foreground, (c) minimum bounding box

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

This information is used as a computationally efficient background/foreground updating mechanism that updates the background at every frame instance by using the intensity of motion vectors within an area. If motion vectors' intensity is greater than a threshold then this area is denoted as foreground, otherwise it is denoted as background.

B. *2D Foreground Object Width-Height Ratio*

Width-height ratio estimation requires information about the projected width and projected height of foreground object. In order to estimate the projected width and projected height of foreground object a minimum bounding box that includes the foreground object was created. Figure 2 shows foreground extraction and minimum bounding box for a captured frame. By using the four corners of the bounding box the left-most, right-most, top-most and bottom-most points of foreground object can be estimated and width-height ratio can be expressed by (3).

$$R = \frac{w_p}{h_p} = \frac{p_{rm} - p_{lm}}{p_{tm} - p_{bm}} \qquad (3)$$

where w_p and h_p are projected width and projected height and p_{rm}, p_{lm}, p_{tm}, p_{bm} are the left-most, right-most, top-most and bottom-most points of foreground object respectively.

C. *Estimation of 3D Measures for Detecting Falls*

As mentioned before, vertical motion velocity computation requires camera calibration as this process relates camera measurements with measurements in the real, three dimensional, world according to (4). This relation is a critical component in any attempt to find the dimensions of an object in a three dimensional scene.

$$q = MQ,\ \ q = \begin{bmatrix} x \\ y \\ w \end{bmatrix}, M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} X \\ Y \\ W \end{bmatrix} \qquad (4)$$

where q is a point on camera's plane, Q is the same point in three dimensional world and M is camera's intrinsic matrix. The two parameters, c_x and c_y, have to be introduced to model a possible displacement between the principle point and the center of the imager, while two different focal lengths are used because the individual pixels on a typical low-cost imager are rectangular rather than square. Our approach to camera calibration is derived from [14], which tries to determine optimal values for intrinsic parameters based on image observations of a known target.

Besides camera calibration, vertical motion velocity computation requires the construction of a reference plane that is the orthographic view of the floor. This construction is a perspective transformation, which can be though as a specific case of projective homographies. As described in [15], an affine space R^n is transformed to a projective space P^n by the following mapping:

$$(x_1, x_2, \dots, x_n)^T \to (x'_1, x'_2, \dots, x'_n, x'_{n+1})^T = (x_1, x_2, \dots, x_n, 1)^T$$

and the inverse mapping, from the projective space P^n to the affine space R^n, is given as:

$$(x'_1, x'_2, \dots, x'_n, x'_{n+1})^T \to (x_1, x_2, \dots, x_n)^T =$$

$$= (\frac{x'_1}{x'_{n+1}}, \frac{x'_2}{x'_{n+1}}, \dots, \frac{x'_n}{x'_{n+1}})^T$$

where $x'_{n+1} \neq 0$.

For a projective space P^n, a projective homography is defined as a nonsingular matrix $H_{(n+1)x(n+1)}$. A point $\boldsymbol{x}$ is projectively transformed to $\boldsymbol{x'}$ as follows:

$$\boldsymbol{x}' = H\boldsymbol{x}, \quad \boldsymbol{x}, \boldsymbol{x}' \in P^n \qquad (5)$$

where $\boldsymbol{x}$ denotes pixel coordinates in the homogeneous coordinates and $\boldsymbol{x'}$ is a new position of a pixel in the wrapped output image.

By using perspective transformations, any parallelogram can be transformed to any trapezoid, and vice versa. In our case, we want to transform the camera's plane to a reference plane that represents the orthographic view from above of the camera's plane. Then according to the inverse perspective mapping algorithm described in [16], $\boldsymbol{x'}$ and $\boldsymbol{x}$ can be expressed by the following relations:

$$\boldsymbol{x'} = [x' \quad y' \quad 1]\ and\ \boldsymbol{x} = [x \quad y \quad 1] \qquad (6)$$

where $x, y, x^{'}, y'$ represent Cartesian coordinates on image plane and reference plane respectively, homography matrix $H = [h_{ij}]$can be normalized so to have $h_{33} = 1$ and through (6) equation (5) is expressed as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad (7)$$

This equation that represents a perspective transform requires at least four non-collinear points in order to be solved. By using observations of a known target, a larger set of points can be found and this equation can be solved in a least square sense. The quality of the transformation is measured by the Back Projection Error [16], associated with H(8).

$$E = \sum_{i=1}^{n} \left(x'_i - \frac{h_{11}\, x_i + h_{12} y_i + h_{13}}{h_{31} x_i + h_{32} y_i + h_{33}} \right)^2 + \left(y'_i - \frac{h_{21}\, x_i + h_{22} y_i + h_{23}}{h_{31} x_i + h_{32} y_i + h_{33}} \right)^2 \qquad (8)$$

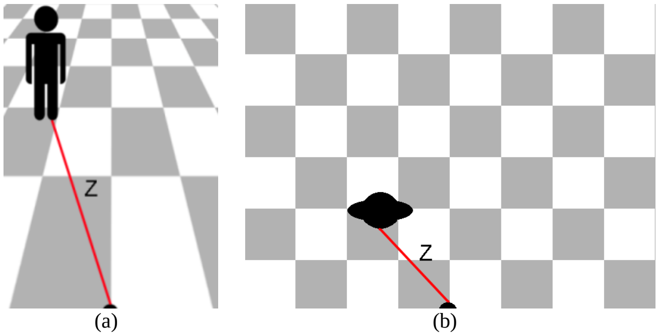


Figure 3: (a) camera's plane, (b) reference plane

Figure 3 shows both camera's and reference planes, while Figure 4 shows input images along with their inverse perspective transformation output images. To approximate the distance Z between foreground object and camera, we use the bottom-most point of foreground object, p_{bm}. As shown in Figure 3(b), on the reference plane the relation between camera's natural units (pixels) and the units of the physical world (cm) is linear and thus distance Z is straightforwardly calculated.

This results in a simple model and a single solution in which a point in the physical world (X, Y, Z) with actual height h_a is projected on the image plane with projected height h_p in accordance with (9). However, the appearance of errors during perspective transformations affects the actual height estimation, as it depends on distance estimation on created reference plane.

$$h_a = Z\frac{(h_p - c_y)}{f_y} \quad (9)$$

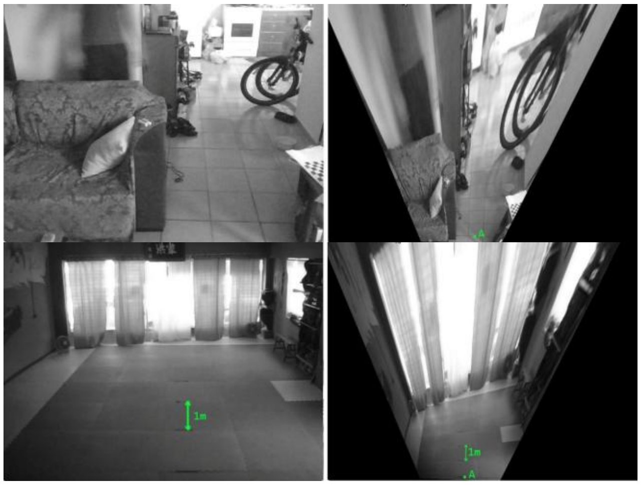


Figure 4: Input images and their inverse perspective transformation mappings

In order to use (1), actual height has to be approximated for every captured frame. Because of the motion of foreground object, errors in the calculation of its height may occur. Let us denote as $\hat{h}(i)$ this approximate height of the foreground object at the current frame of analysis i. In our approach, to reduce accumulation of the approximation errors to the following frames to process we use a heuristic iterative methodology, which updates the foreground height taking into account previous height information and the current one, yielding to a robust approximate solution, denoted as $h(i)$, which is computed by (10). This iterative procedure requires an initial value of $h(i)$ which in our case is set to average height for adult males, e.g., 175cm.

$$h(i) = \lambda h(i-1) + (1-\lambda)\hat{h}(i) \quad (10)$$

where λ is a parameter that regulates the importance of $\hat{h}(i)$ to the iterative procedure. For our experiments, λ is set to 0.8, since this value yields the more reliable performance.

By using this form for every captured frame, $h(i)$ converges to the actual height of foreground object. In order to reduce wrong estimations when a fall event occurs and height is significantly decreases, height $h(i)$ is being updated only if $\hat{h}(i)$ is bigger and smaller than a threshold (in our case ±20 cm). Figure 5 shows the approximation of foreground object's actual height. The gray line represents the approximation of foreground object's actual height for the first 1,000 frames of system operation, while the horizontal line represents its actual height.

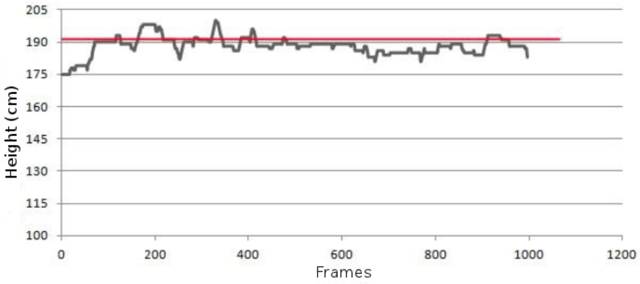


Figure 5: Actual height approximation

IV. Experimental Results

The application was developed on a PC with 4GB RAM and a dual-core Intel processor at 2.1GHz. The camera that was used was a simple USB webcam with 640x480 pixels resolution. The code was written in C by using OpenCV library. By using this hardware, this algorithm operates in real time at 14fps. In quad-core computers, the time can be reached up to 17fps.

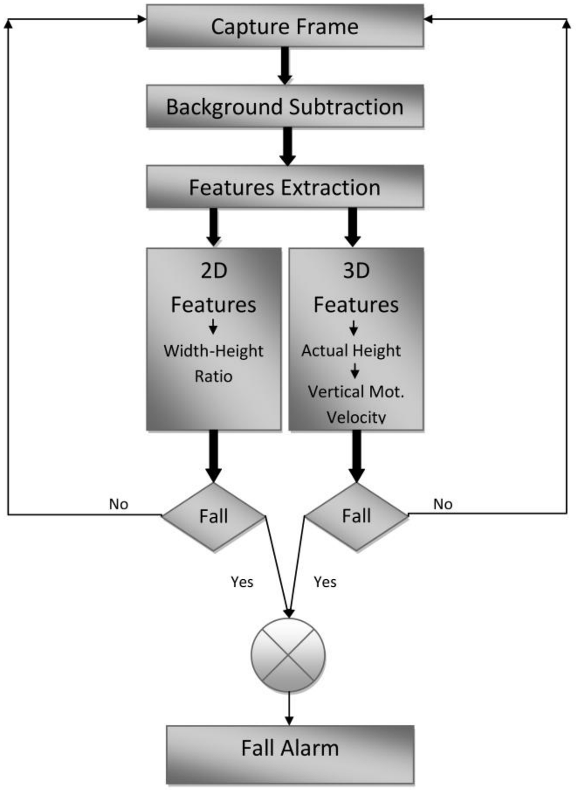
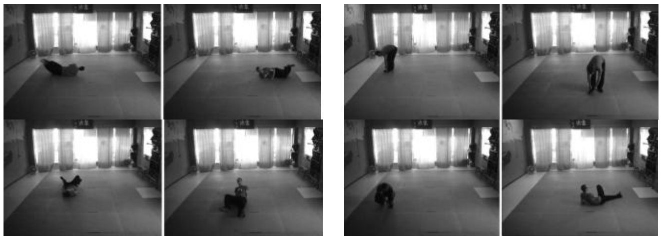


Figure 6: Workflow diagram of presented fall detection scheme

The workflow of the system is presented in Figure 6. For every captured frame, initially, the background subtraction algorithm takes place. The output of this algorithm leads to the extraction of the foreground, and thus, the features that are used by the fall detection algorithm (vertical motion velocity and width-height ratio). The fall detection algorithm, firstly checks if the width height ratio suggests a fall. If this feature suggests a fall, then the vertical motion velocity is calculated and compared with a threshold relative to the real height of the foreground object measured in cm. If this feature suggests a fall too, then a fall alarm occurs. By measuring vertical motion velocity in cm, the performance of the system is not affected by cases where the foreground object is far away or close to the camera.



(a) (b)

Figure 7: (a) Falls in every direction according to the camera position, (b) normal everyday activities that look like a fall, like bending and lying down

During the experimentation process one person simulated falls, in every direction according to the camera position and normal every day activities, that may look like falls; but, they are not real falls; see Figure 7. Because of the nature of the algorithm two different variables affect its performance; the width-height ratio of foreground object and its vertical shifting during a sequence of frames. Figure 8 and Figure 9 describe the performance of the system, concerning on successful fall detection, when the camera was placed at the height of 260cm. In the diagram in Figure 8, we keep constant the value of vertical motion velocity threshold, while in the Figure 9 we keep constant the value of width-height ratio threshold.

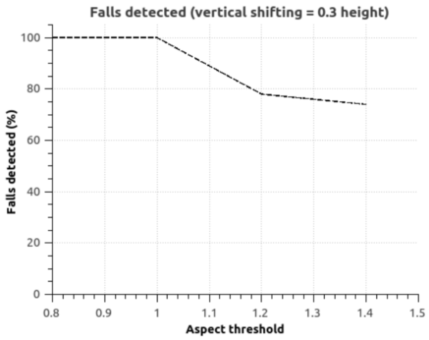


Figure 8: Performance in regard to width-height ratio when camera placed at 260cm

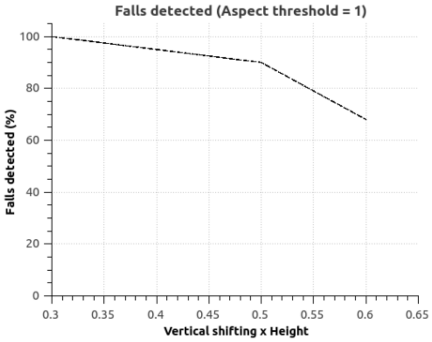


Figure 9: Performance in regard to vertical shifting when camera placed at 260cm

TABLE I: PERFORMANCE WHEN CAMERA PLACED AT DIFFERENT HEIGHTS

Camera's height		**Proposed system**	**System of** [8]
40cm	Falls detected	92.8%	92.8%
	Wrong detections	4	4
220cm	Falls detected	92%	72%
	Wrong detections	6	6
260cm	Falls detected	96%	73%
	Wrong detections	2	2

ISBN: 978-1-61208-226-4

Table I summarizes its performance and compares it with system performance proposed in [8]. As this system is an expansion of [8], it was expected to perform better. Performance of system proposed in [8] was affected by the camera’s position, in contrast, this system performs well for a much wider range of placements that permits a camera mounted in a higher position, favoring fall detection process by providing better coverage with less obstacles inserted into its field of view. This comparison was performed by using the same demo video as input into both fall detection systems.

V. Conclusion and future work

This paper presented a fall detection scheme that exploits 3D measures by using a single monocular camera. The proposed scheme has the capability to operate in real-time and to detect over 92% of fall incidents in complex and dynamically changing visual conditions, while it presents low false positive rate. Its minimal computational cost and memory requirements, let alone its low financial cost since simple ordinary low resolution cameras are used, making it affordable for a large scale.

This algorithm makes the assumption that only one person is present in the scene; so, primary priority in to-do list is its evolution in a way that it will operate properly when more than one person are present in the scene and even in crowded conditions.

Through our proposed scheme, besides the contribution to humans’ fall problem, significant measures of a 3D scene can be calculated that can reveal much more information which might be useful in different kind of applications.

VI. Acknowledgments

The research leading to these results has been supported by European Union funds and national funds from Greece and Cyprus under the project “POSEIDON: Development of an Intelligent System for Coast Monitoring using Camera arrays and Sensor Networks” in the context of the inter-regional programme INTERREG (Greece-Cyprus cooperation) – contract agreement K1 3 1017/6/2011.

References

[1] S. Wang, J. Yang, N. Chen, X. Chen, and Q. Zhang, “Human activity recognition with user-free accelerometers in the sensor networks,” in *Neural Networks and Brain, 2005. ICNN B ’05. International Conference on*, 2005, vol. 2, pp. 1212 –1217.

[2] M. N. Nyan, F. E. H. Tay, and E. Murugasu, “A wearable system for pre-impact fall detection,” *Journal of Biomechanics*, vol. 41, no. 16, pp. 3475–3481, 2008.

[3] T. M. Le and R. Pan, “Accelerometer-based sensor network for fall detection,” in *Biomedical Circuits and Systems Conference, 2009. BioCAS 2009. IEEE*, 2009, pp. 265 –268.

[4] F. Bianchi, S. J. Redmond, M. R. Narayanan, S. Cerutti, and N. H. Lovell, “Barometric Pressure and Triaxial Accelerometry-Based Falls Event Detection,” *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 18, no. 6, pp. 619 –627, Dec. 2010.

[5] Y. Zigel, D. Litvak, and I. Gannot, “A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound - Proof of Concept on Human Mimicking Doll Falls,” *Biomedical Engineering, IEEE Transactions on*, vol. 56, no. 12, pp. 2858 –2867, Dec. 2009.

[6] N. Doulamis, “Iterative motion estimation constrained by time and shape for detecting persons’ falls,” in *Proceedings of the 3rd International Conference on PErvasive Technologies Related to Assistive Environments*, New York, NY, USA, 2010, pp. 62:1–62:8.

[7] Z. Fu, E. Culurciello, P. Lichtsteiner, and T. Delbruck, “Fall detection using an address-event temporal contrast vision sensor,” in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, 2008, pp. 424 –427.

[8] A. Doulamis and K. Makantasis, “Iterative Scene Learning in Visually Guided Persons’ Falls Detection,” in *19thEUSIPCO*, Barcelona, 2011, pp. 779–783.

[9] H. Foroughi, A. Rezvanian, and A. Paziraee, “Robust Fall Detection Using Human Shape and Multi-class Support Vector Machine,” in *Computer Vision, Graphics Image Processing, 2008. ICVGIP ’08. Sixth Indian Conference on*, 2008, pp. 413 –420.

[10] H. Qian, Y. Mao, W. Xiang, and Z. Wang, “Home environment fall detection system based on a cascaded multi-SVM classifier,” in *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, 2008, pp. 1567 –1572.

[11] G. Diraco, A. Leone, and P. Siciliano, “An active vision system for fall detection and posture recognition in elderly healthcare,” in *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, 2010, pp. 1536 –1541.

[12] N. Thome, S. Miguet, and S. Ambellouis, “A Real-Time, Multiview Fall Detection System: A LHMM-Based Approach,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 11, pp. 1522 –1532, Nov. 2008.

[13] B. Lucas and T. Kanade, “An Iterative Image Registration Technique with an Application to Stereo Vision,” presented at the IJCAI81, 1981, pp. 674–679.

[14] J. Heikkila and O. Silven, “A four-step camera calibration procedure with implicit image correction,” in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, 1997, pp. 1106 –1112.

[15] B. Cyganek and J. P. Siebert, “Front Matter,” in *An Introduction to 3D Computer Vision Techniques and Algorithms*, John Wiley & Sons, Ltd, 2009, p. i–xx.

[16] A. Bevilacqua, A. Gherardi, and L. Carozza, “Automatic Perspective Camera Calibration Based on an Incomplete Set of Chessboard Markers,” in *Computer Vision, Graphics Image Processing, 2008. ICVGIP ’08. Sixth Indian Conference on*, 2008, pp. 126 –133.

FlexRay Static Section Scheduling Using Full Model

Rim Bouhouch, Houda Jaouani, Wafa Najjar, Salem Hasnaoui

SysCom Laboratory
National Engineering School of Tunis
Tunis, Tunisia
{rim.bouhouch@yahoo.fr, jouani_houda@yahoo.fr, wafa_najjar@yahoo.fr, salem.hasnaoui@enit.rnu.tn}

Abstract—**In this paper, we propose a new scheduling method for FlexRay static segment tasks, on the node level. This method handles the periodic communicating tasks transmitted on the static segment of the bus and takes into account the effect of the disruptive tasks, such as interruptions, on the response time. In this context, our scheduling method is based on the full model that we evaluate the performance by calculating the response time of the communicating tasks using as application model the SAE benchmark.**

Keywords-Scheduling; FlexRay Bus; Periodic Tasks; Full Model; Worst Case Response Time.

I. Introduction

FlexRay [1] is a new communication system that offers reliable and real-time capable high-speed data transmission between electrical and mechatronic components to map current and future innovative functions into distributed systems within automotive context. Thanks to its several features, this communication protocol is meeting safety critical applications performance requirements (flexibility, fault-tolerance, determinism, high-speed, etc.). Therefore, FlexRay is emerging as a predominant protocol for in-vehicle x-by-wire applications (i.e., drive-by-wire, steer-by-wire, brake-by-wire, etc.). As a result, there has been a lot of recent interest in timing analysis techniques in order to provide bounds for the message communication times on FlexRay. The real-time Data Distribution Service (DDS) based on the subscription-publication paradigm offers a clear distinction between the communicating tasks by classifying them into DataReaders and DataWriters and that helps insuring the delivery of the right data on the right time. One of the most interesting combinations would be the use of DDS on top FlexRay networks; but the challenge remains in the scheduling of the DataWriter provided by the applications and according to DDS specification, to meet the DataReaders Deadlines.

In this paper, we provide a scheduling method for periodic tasks on the static segment, based on the full model taking into account all the disruptive events and their effect on the response time of the Writers evaluated by the WCRT.

In the first section, we present the related work dealing with scheduling in the FlexRay bus; in the second section, an overview of the FlexRay network and its features is given; the third section is dedicated to scheduling parameters in the bus and in the static section; the fourth section presents the response time calculation using the full model; in the fifth section we present the application model on which we have performed our tests, and, in the last section, we present the results of our tests.

II. Related work

Tasks in real-time networks such as FlexRay [1] or CAN [14] are scheduled according to a static or a dynamic scheduling method. A static scheduler is a time triggered scheduling based on the Time Division Multiple Access (TDMA) [14], where each participant is granted a specific fixed interval in a repetitive time window. TDMA scheduling guarantees a deterministic transfer of messages, but has the disadvantage that the bandwidth is not used efficiently. A dynamic scheduling is an event triggered scheduling where participants can only send information if an event occurs, such as new data is ready for transmission.

Our previous researches [2] were interested in scheduling for the Data Distribution Service (DDS) architecture over CAN. We have developed in each node a local scheduling component, the Earliest Deadline First (EDF) scheduler. The latter, sends scheduling parameters of tasks to the global scheduling system. Then information is sent to a distributed information collection service called the System Information Repository (SIR). In [3], we have presented how DDS API is implemented on top of FlexRay Driver. In [4], we have presented a combined scheduling method that can be applied for both static and dynamic scheduling in FlexRay.

Related studies to this research include time triggered, event triggered and automobile protocols.

First studies [5] illustrate how a window-based analysis technique can be used to find Worst-Case Response time of a task. It considers bursty sporadic activities, where tasks arrive sporadically but then execute periodically for some bounded time.

Hagiescu *et al.* [6] proposes an analytical framework for compositional performance analysis of a network of Electronic Controller Unit (ECU) that communicates via a FlexRay bus. The main contribution was a formal model of the protocol governing the static segment of FlexRay.

In this paper, we focus our interest on the static segment of FlexRay and propose a new scheduling method that handles all the disruptive tasks and their effects on the response time, to evaluate the deadline of the communicating tasks.

III. FlexRay Networks

FlexRay has been developed by the FlexRay consortium since 2000 for safety related applications in the automotive industry [1]. It is today applied in real-time application and as a replacement of CAN when higher data rates are required.

FlexRay has been developed to support x-by-wire applications such as steer-by-wire or brake-by-wire. These are replacements of the traditional mechanical and hydraulic control systems through electronic control systems.

FlexRay features two communication channels, each with a data rate of 10 Mbits/s, and payloads of frames up to 254 Bytes. Furthermore, the communication is time triggered in contrast to the event triggered CAN protocol. This is why FlexRay guarantees fixed communication latencies and a global synchronous time basis for all participating electronic control units.

A. Topologies

A FlexRay cluster consists of several nodes and two communication channels, channel A and channel B. In order to provide reliable communication, a node must be connected to both communication channels. To reduce cost using only one channel can be sufficient.

FlexRay supports both bus and star topologies. To increase the communication distance between two nodes they have to be connected via star couplers [7].

B. Hierarchical Network Timing

The communication scheme of a FlexRay cluster is built up of communication cycles that are repeated over again from startup of the network until it is shutdown. A communication cycle consists of the network communication time and the network idle time.

The communication time includes a mandatory static segment, an optional dynamic segment, and the symbol window.

In the static segment, deterministic communication ensures constant latency. FlexRay adheres to a time division multiple access method (TDMA), which means that there are equally sized slots and that the point of time is fixed when a frame is transmitted on the channel.

In the dynamic segment event driven communication takes place. This is usually used for low priority data, for example for the transmission of diagnosis information [8].

C. Electronic Control Unit (ECU)

The software application is executed on a host processor which is connected to a dedicated communication controller that executes the FlexRay protocol. The transmission from digital signals of the communication controller to analog signals on the bus is accomplished by the bus driver.

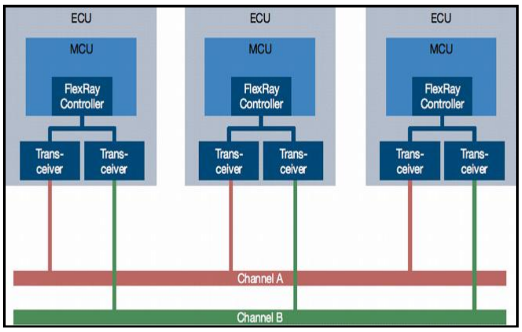


Figure 1. FlexRay Node (ECU)

IV. Scheduling Parameters in FlexRay Networks

A. FlexRay Bus

FlexRay is a real time communication bus [1] designed to operate at speeds of up to 10 Mbits/s. He was developed by a consortium that includes automobile builders. It offers time-triggered and an event triggered architecture. Data is transmitted in payload segment containing between 0 and 254 bytes of data, 5 bytes for the Header segment and 3 bytes for the trailer segment. The topology may be linear bus, star or hybrid. This bus contains two channels; each node could be connected to either one or both channels.

FlexRay bus contains a static segment for time triggered messages and a dynamic segment for event triggered messages. In time triggered networks, nodes only obtain network access at specific time periods, also called time slots. In event triggered networks nodes may obtain network access at any time instant. The static (ST) segment and the dynamic (DYN) segment lengths can differ, but are fixed over the cycles. Both the ST and DYN segments are composed of several slots. The first two bytes of the payload segment are called message ID, this is used only in dynamic segment. The message ID can be used as a filterable data.

In this paper, we will study the transmission parameters of DDS nodes on a FlexRay bus. During any slot, only one node is allowed to send on the bus, and that is the node which holds the message with the frame identifier (Frame ID) equal to the current value of the slot counter. There are two slot counters, corresponding to the ST and DYN segments, respectively. The assignment of frame identifiers to nodes is static and decided offline, during the design phase. Each node that sends messages has one or more ST and /or DYN slots associated to it. The bus conflicts are solved by allocating offline one slot to at most one node, thus making possible for two nodes to send during the same ST and DYN slot. FlexRay allows the sharing of the bus among event driven (ET) and time driven (TT) messages.

For a distributed system based on FlexRay, task scheduling can be SCS (Static Cyclic Scheduling) or FPS (Fixed Priority Scheduling). For the SCS tasks and ST messages, the schedule table could be built. For FPS tasks and DYN messages, the worst-case response times had to be determined.

B. Communication Cycle

The FlexRay protocol organizes time into communication cycles, every cycle is organized into four parts, segments of configurable duration: The static segment is used to send critical, real-time data, and is divided into static slots, in which the electronic control units (ECUs) can send a frame on the bus. These frames consist of a header, payload and trailer and are assigned to the slots according to a static, TDMA-based schedule. Channel idle time is enforced between frames to prevent overlapping consecutive frames. The dynamic segment enables event-triggered communication. The lengths of the mini slots in the dynamic segment depend on whether or not an ECU sends data. The symbol window is used to transmit special symbols, for example to start up the FlexRay cluster. The network idle time interval is used by the nodes to allow them to correct their local time bases in order to stay synchronized to each other.

The length of an ST slot is specified by the FlexRay global configuration parameter gdStaticSlot. The length of the DYN segment is specified in number of mini-slots gNumberOfMinislots.

C. Static segment parameters

In a general communication process, response time can be divided in four pieces, as shown in Fig. 1: generation delay, queuing delay, transmission delay and reception delay [9].

Generation delay is started when the transmitting node received the request of sending from a frame until the data is written into the buffer and ready for being sent. Queuing delay is started when generation delay ended until the frame acquires the occupation of the bus and begins to be sent. Transmission delay is the time during which the frame is being transmitted on the bus. Reception delay is started when the frame gets off the bus and goes into the receiving node until the frame accomplishes its task.

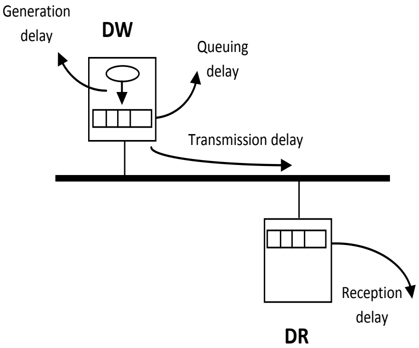


Figure 1. Communication Model between DataReader and DataWriter

Note that the generation delay and reception delay are not related to the FlexRay network characteristics, but related to the given MCU performance. Therefore, these two parts of delay should not be taken into account. In FlexRay protocol the average response time R_m of a given frame is the sum of queuing delay average (t_m) and transmission delay average (C_m):

$$R_m = t_m + C_m \quad (1)$$

Since the static segment is transmitting at fixed time points in each FlexRay communication cycle without any queuing delays, the response time can be approximated by Cm.

$$R_m = C_m \quad (2)$$

Transmission delay Cm refers to the time interval between being on the bus and completion of sending process. It depends on the frame itself as well as bus parameters.

$$C_{m,s} = [TSS + FSS + FES + t_d + (HS + TS + S_m) \times (8 + BSS)]\tau_{bit} \quad (3)$$

TSS is the Transmission Start Sequence (3~15 bits). FSS is the Frame Start Sequence (1 bit). FES is the Frame End Sequence (2 bits). t_d is the delay related to sending and receiving nodes, which is around 2~3 bits. S_m represents the data field length (number of bytes) of the data frames. In addition, two BSS (Bit Start Sequence) are added before each byte. The constant “8” added to the data field length Sm refers to the sum of the FlexRay Header Segment (HS: 5) and Trailer Segment (TS: 3) lengths (number of bytes). Finally, τ_{bit}refers to the one bit transmission delay.

V. Response Time calculation

The full model is inspired from the FPS (First Priority Scheduling) approach [10], which is the most widely used approach in the computing world. In this case, each task has a fixed static priority, which is ECU pre-run-time. The runnable tasks are executed in the order determined by their priority, knowing that in real-time systems, the “priority” of a task is derived from its temporal requirements, not its importance to the correct functioning of the system or its integrity.

The full model was conceived to be used in an industrial context [10], the temporal overheads of implementing the system must be taken into account such as:

- Context switches (one per job)
- Interrupts (one per sporadic task release)
- Real-time clock overheads

In this case, the Response time equation is rather than:

$$R_i = C_i + \sum_{j\in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j \quad (4)$$

where $hp(i)$ is the set of tasks with priority higher than task i, C_i is the worst case computation time of the task i and T_j is the minimum time between task releases, jobs or task period.

The new equation is:

$$R_i = CS^1 + C_i + B_i + \sum_{j\in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil (CS^1 + CS^2 + C_j) \quad (5)$$

where the new terms CS^1 and $CS^2$are the cost of switching to the task, and the cost of switching away from it. And the term B_i is the cost of the task worst case blocking time.

The cost of handling interrupts is:

$$\sum_{k\in\Gamma_s} \left\lceil \frac{R_i}{T_k} \right\rceil IH \quad (6)$$

where Γ_s is the set of sporadic tasks and IH is the cost of a single interrupt (which occurs at maximum priority level).

There is also a cost per clock interrupt, a cost for moving one task from delay to run queue and a (reduced) cost of moving groups of tasks

Let CT_c be the cost of a single clock interrupt, $\Gamma_p$be the set of periodic tasks, and CT_s be the cost of moving one task. The following equation can be derived

$$R_i = CS^1 + C_i + B_i + \sum_{j\epsilon hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil \left(CS^1 + CS^2 + C_j\right) + \sum_{k\epsilon\Gamma_s} \left\lceil \frac{R_i}{T_k} \right\rceil IH + \left\lceil \frac{R_i}{T_{clk}} \right\rceil CT_c + \sum_{g\epsilon\Gamma_p} \left\lceil \frac{R_i}{T_g} \right\rceil CT_s \qquad (7)$$

Within the static segment a static time division multiple access scheme is applied to coordinate transmissions. In the static segment all communication slots are of identical, statically configured duration and all frames are of identical, statically configured length. In order to schedule transmissions each node maintains a slot counter state variable *vSlotCounter* for channel A and a slot counter state variable *vSlotCounter* for channel B. Both slot counters are initialized with 1 at the start of each communication cycle and incremented at the end boundary of each slot.

In the Implementations of the FlexRay bus, the periodic and safety-critical data is scheduled on the static time-triggered segment so the tasks in the static segment are periodic tasks that have the same priority per communication cycle.

Considering these facts the equation (7) applied on the static segment context becomes:

$$R_i = CS^1 + C_i + B_i + + \sum_{k\epsilon\Gamma_s} \left\lceil \frac{R_i}{T_k} \right\rceil IH + \left\lceil \frac{R_i}{T_{clk}} \right\rceil CT_c + \sum_{g\epsilon\Gamma_p} \left\lceil \frac{R_i}{T_g} \right\rceil CT_s \qquad (8)$$

VI. Application model

To illustrate the utility of our Comprehensive Scheduling Strategy, we have chosen to work within a platform of a vehicular network based on the SAE standard. In this system, a set of network processors subsystems produces routing data. This data must be distributed along the vehicular network.

In fact, we will apply the studied approaches on a new vehicle benchmark developed in [11] and based on the SAE Benchmark [15]. We added to the original benchmark a number of nodes and messages to better represent the complexity of today's vehicles and to model some added options responsible for improving vehicle safety, reliability, cost, and luxury.

However, this Benchmark was designed to best fit the CAN network and it needs major modifications to be adapted to the FlexRay protocol. Hence, later in this paper, we will explain how to introduce adjustments to that model and we will apply our scheduling algorithm and present our results for the new model. The resulting architecture is composed of 15 nodes connected by the FlexRay bus. According to the FlexRay specification, each node consists of a host (CPU) that processes incoming messages and generates outgoing messages, a communication controller (CC) that independently implements the FlexRay protocol services, and a two-way controller-host interface (CHI) that serves as a buffer between the host and the CC.

The main goal of the proposed architecture is to insure better performance of the vehicular network and to guarantee the arrival of the right data on the right time by meeting the tasks deadline. The framework architecture is a set of nodes connected via FlexRay Real-Time Transport protocol. In each node is embedded a Real-Time Operating System µCOSII and a publish/subscribe middleware.

VII. Results and comments

In this section, we propose an algorithm to calculate the response time of the DataWariters tasks.

The equation (8) gives us the needed parameters to determine the response time for both static and dynamic segments tasks:

- C_i the computing time is equivalent to the transmission delay $C_{m,s}$ and $C_{m,d}$, because the execution of a message relative to a writing task is the fact to transmit data on the bus.
- The worst blocking time Bi is defined as follows:

$$B_i = \frac{Frame\ size - 1\ bit}{Lowest\ flow\ rate\ used} \qquad (9)$$

This equation is true for the CAN case; but. in the FlexRay case:

$$B_i = 0$$

- CS^1 is the cost of switching to the task. This parameter is given by the used real-time operating system µCOSIII [12].

$$CS^1 = 0.005\,ms$$

- CS^2 is the cost of switching away from the task, this parameter is also given by the used real-time operating system µCOSIII [12].

$$CS^2 = 0.009\,ms$$

- IH is the cost of executing an interrupt service routine. This interrupt is supposed to be at the maximum priority level. The number of STATUS registers present in the system determines the time taken by the handler to execute the interrupt routine. The FlexRay driver interrupt routine takes more time in response to the status of receiving communications data. For our study we approximate this parameter as follow:

$$IH = 10 \times CT_c$$

- CT_c is the cost of a single clock interrupt for the microcontroller MB91F465X we have approximated its value:

$$CT_c = \frac{1}{10} \times T_{clk}$$

- CT_s is the cost of moving one task, which is equivalent to switching a task.

$$CT_s = CS^2$$

- T_{clk} is the clock period calculated for a given core frequency.

The response time calculation process is described by the following algorithm:

Algorithm Worst Case Response Time Computing

```
for i in 1..N loop
  n := 0
  loop
    Calculate $C_i$ for periodic tasks
    Calculate $C_i$ for sporadic tasks
    n := n + 1
  end loop
end loop
for i in 1..N loop
   n := 0
   $W_i^n = C_i$
   loop
      calculate new $w_i^{n+1}$
      if $w_i^{n+1} = w_i^n$ then $R_i = w_i^n$
         exit value found
      end if
      if $w_i^{n+1} > T_i$ then
         exit value not found
      end if
      n := n + 1
   end loop
end loop
```

For the simulation, we consider a set of FlexRay nodes the sending 36 messages on the FlexRay bus. Since each node in the system that generates static messages needs at least one static slot, the minimum number of static slots is the number of nodes (*nodesST*) sending static messages [1].

In the extended benchmark [11], there are 15 nodes sending 36 messages; among them, 30 are periodic messages that need to be scheduled on the FlexRay static segment. We will regroup these nodes into 6 for the simulations.

The period of the bus cycle (*gdCycle*) must be lower than the maximum cycle length *cdCycleMax* equal to 16 ms and has to be, also, an integer divisor of the period of the global static segment. In addition, each node has a counter *vCycleCounter* in the interval 0…63. Thus, during a period of the global static schedule there can be at most 64 bus cycles. Observing our message set, we have noticed that almost all of the message periods are multipliers of 5 ms. So we can fix the period of the bus cycle to 5 ms and adjust some message periods, especially the messages introduced by Ben Gaid, M-M in [13] and others introduced by M. Utayba in [11].

All messages with period equal to 8 ms will have a new period of 5 ms, and the messages with period equal to 12 ms will have a period of 10 ms. This will not affect our system efficiency since it will make it faster and more reactive.

There is another problem with messages having a 1000 ms period; they cannot be scheduled with a bus cycle of 5ms and 64 cycles. In fact, even if we consider the longest period of the global static schedule (64 bus cycles), we wouldn't manage to reach the 1000 ms. Thus, we have to decrease this period to 64*5=320 ms.

We have also replaced the original bus priorities designed for an event triggered bus (CAN) by a local priority able to order transmission of messages having the same Frame Identifier on different slots assigned to their source node.

Applying the previous algorithm with a bus speeds of 10 Mbit/s for one channel transmission scheme, and a core frequency of 12 Mhz. The results obtained are summarized in Table I.

TABLE I. BODY CONTROL MODULE RESULTS

Vehicle Module	Message ID	Size (Bytes)	Deadline [ms]	T [ms]	Task Priority	Worst Case Response Time R (ms)
BODY Control Module	3	1	5	5	1	0.1397
	13	1	**5**	**5**	1	0.1397
	31	4	100	100	1	0.1487
	34	3	**320**	**320**	1	0.1457
Engine Controller Module	4	2	5	5	1	0.1424
	6	2	5	5	1	0.1424
	20	2	10	10	1	0.1424
	35	1	**320**	**320**	1	0.1394
Active Suspension Unit	27	2	**10**	**10**	1	0.2229
Active Frame Steering	22	2	10	10	1	0.2229
Electronic Brake Control Module	14	4	**5**	**5**	1	0.2289
Traction Control Unit	8	1	5	5	1	0.2199
	15	4	**5**	**5**	1	0.2289
ESP/ROM	16	4	**5**	**5**	1	0.2289
	28	5	**10**	**10**	1	0.2319

Hydraulic Brake Control Unit	2	2	5	5	1	0.2499
	32	1	100	100	1	0.2469
Transmission Control Unit	5	1	5	5	1	0.2469
	33	1	100	100	1	0.2469
	36	1	**320**	**320**	1	0.2469
Throttle Control Unit	7	1	5	5	1	0.2469
Adaptive Cruise Control	29	3	10	10	1	0.2529
Front-Right Wheel Module	10	1	**5**	**5**	1	0.1389
	24	2	**10**	**10**	1	0.1419
Rear-Right Wheel Module	12	1	**5**	**5**	1	0.1389
	26	2	**10**	**10**	1	0.1419
Front-Left Wheel Module	9	1	**5**	**5**	1	0.1389
	23	2	**10**	**10**	1	0.1419
Rear-Left Wheel Module	11	1	**5**	**5**	1	0.1389
	25	2	**10**	**10**	1	0.1419

We notice, on this table of results, that the entire tasks deadline is matched. For the worst case response time using the worst case Core frequency which is 12 Mhz, we have noticed that the deadline has been met and the equation below is verified.

$$D \geq R$$

Thanks to FlexRay bus speed, we can assume that the DDS Deadline QoS Policy can always be reached.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we have proposed to use DDS on top of the real-time network FlexRay to take advantage of its high speed and to profit of the DDS QoS management in an automotive context. We have proposed a scheduling model based full FPS scheduling to first calculate the worst case response time for our vehicular system and evaluate its performance on a benchmark application, an extended SAE benchmark. After the simulations, results have shown that the applications deadline requirements have been met. One promising research direction would be the evaluation of the real-time QoS parameters offered by DDS on the same system configuration.

ACKNOWLEDGMENT

The researches presented in this paper would not have been possible without the support of many people. We wish to express our gratitude to the SYSCOM ENIT members for their help and assistance.

REFERENCES

[1] FlexRay Consortium," FlexRay Communications System-Protocol Specification", Version 2.1, Revision A, 2005.

[2] T. Guesmi, R. Rekik, S. Hasnaoui, and H. Rezig, "Design and Performance of DDS-based Middleware for Real-Time Control Systems", IJCSNC, vol. 7, No. 12, 2007, pp. 188-200.

[3] R. Bouhouch, W. Najjar, H. Jaouani, and S. Hasnaoui, "Implementation of Data Distribution Service Listeners on Top of FlexRay Driver", INFOCOMP 2011, IARIA, October 2011,Barcelona Spain, pp. 64-69.

[4] W. Najjar, R. Bouhouch, H. Jaouani, and S. Hasnaoui, "Static and Dynamic Scheduling for FlexRay Network Using the Combined Method", International Journal of Information Technology and Systems, Vol. 1, No. 1, January 2012, pp. 18-26.

[5] K. W. Tindell, A. Burns, and A. J. WELLINGS," An extendible approach for analyzing fixed priority hard real-time tasks", Real_Time Systems, Vol. 6, No. 2, 1994, pp. 133-151

[6] A. Hagiescu, U. Bordoloi, and S. Chakraborty, "Performance Analysis of FlexRay-based ECU Networks", DAC proceedings, 2007, pp. 284-289

[7] M. Gerke, "FlexRay : A state of the art vehicle bus, Embedded Systems Lecture", Chair Professor Finkbeiner Saarbrucken, Dec 2008, pp. 3-4.

[8] A. Zhao, "Reliable In-Vehicle FlexRay Network Scheduler Design", master of science thesis in Electrical Engineering, Mai 2011, Delft University of Technology,The Netherlands, pp. 17-23.

[9] T. Guangyn, B. Peng, and C. Quanshi, "Response Time Analysis of FlexRay Communication in Fuel Cell Hybrid Vehicle", Vehicle Power and Propulsion Conference VPPC'08. IEEE, 2008, pp. 1-4.

[10] A. Burns and A. Wellings, "Scheduling Real-Time Systems", Chapter 11, Real-Time Systems and Programming Languages, The university of York, Department of Computer Science, pp. 121-125.

[11] M. Utayba and N. Al-Holou, "Development of An Automotive Communication Benchmark", Canadian Journal on Electrical and Electronics Engineering, Vol. 1, No. 5, August 2010.

[12] A J. J. Labrosse. "MicroC/OS-II The Real Time Kernel". Miller Freeman, Inc, United States of America, 1999.

[13] M-M. Ben Gaid, A. Cela, S. Diallo, R. Kocik, R. Hamouche, and A. Reama, "Performance Evaluation of the Distributed Implementation of a Car Suspension System", the IFAC Workshop on Programmable Devices and Embedded Systems, February 2006, pp.776-787 .

[14] K. Tindell and A. Burns, "Guaranteeing Message Latencies on Control Area Network (CAN)", Real-Time Systems

Research Group, Department of Computer Science, University of York, England, pp. 5-6.

[15] H. Kopetz, “A solution to an automotive Control System Benchmark”, Real-Time Systems Symposium, 7-9 Dec. 1994, pp. 154-158. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.110.3545&rep =rep1&type=pdf.

Analyzing the Trade-offs Between Minimizing Makespan and Minimizing Energy Consumption in a Heterogeneous Resource Allocation Problem

Ryan Friese*, Tyler Brinks*†, Curt Oliver*, Howard Jay Siegel*†, and Anthony A. Maciejewski*
*Department of Electrical and Computer Engineering
†Department of Computer Science
Colorado State University
Fort Collins, CO, 80523
Email: Ryan.Friese@rams.colostate.edu, tbrinks@engr.colostate.edu, wcoliver@rams.colostate.edu,
HJ@colostate.edu, aam@colostate.edu

***Abstract*—The energy consumption of data centers has been increasing rapidly over the past decade. In some cases, data centers may be physically limited by the amount of power available for consumption. Both the rising cost and physical limitations of available power are increasing the need for energy efficient computing. Data centers must be able to lower their energy consumption while maintaining a high level of performance. Minimizing energy consumption while maximizing performance can be modeled as a bi-objective optimization problem. In this paper, we develop a method to create different resource allocations that illustrate the trade-offs between minimizing energy consumed and minimizing the makespan of a system. By adapting a popular multi-objective genetic algorithm we are able to construct Pareto fronts (via simulation) consisting of Pareto-efficient resource allocations. We analyze different solutions from within the fronts to further understand the relationships between energy consumption and makespan. This information can allow system managers to make intelligent scheduling decisions based on the energy and performance needs of their system.**

***Keywords- bi-objective optimization*; *energy-aware*; *makespan*; *heterogeneous computing*; *resource allocation*.**

I. Introduction

Over the past decade, the need for energy efficient computing has become increasingly important. As the performance of high performance computing (HPC) systems, such as servers and datacenters, have increased, so has the amount of energy needed to run these systems. According to the Environmental Protection Agency (EPA) [1], it was estimated that between the years 2000 and 2006 the amount of power consumed by HPC systems more than doubled. An estimated 61 billion kWh was consumed by servers and data centers in 2006, approximately equal to 1.5% of the total U.S. electricity consumption for that year. This is equivalent to the electricity consumption of 5.8 million average U.S. households, and amounts to $4.5 billion in electricity costs [1].

In addition to the rising costs of using so much energy, some data centers are now unable to increase their computing performance due to physical limitations on the availability of energy. A survey conducted in 2008 showed that 31% of the data centers surveyed identified energy availability as a key factor limiting new server deployment [2]. Another example to emphasize this point: Morgan Stanley, a global financial services firm based in New York, is physically unable to draw the energy needed to run a new data center in Manhattan [3].

To battle the rising costs of energy consumption, it is essential for HPC systems to be energy-efficient. This focus on energy-efficiency must have as little impact to performance as possible. Unfortunately, the goals of saving energy and achieving high performance often conflict with each other. To understand and illustrate the trade-offs between energy consumption and computing performance, we model this dilemma as a bi-objective optimization problem. When a problem has multiple objectives, it is often the case that there is not just one single optimal solution, but rather a set of optimal solutions. With our research, we provide a method for developing a set of “Pareto”optimal solutions that not only illustrate the trade-offs between energy consumption and performance for a specific computing system, but also allows the system manager to select a solution that fits the system needs and goals.

In this research, we study how different ways of allocating resources to tasks impact the performance and energy consumption of a computing system. We are modeling a data center consisting of a set of heterogeneous machines that must execute a batch of independent tasks. By heterogeneous, we mean that tasks may have different execution and power consumption characteristics when executed on different machines. All the tasks in a given batch are known *a priori* and are all available for scheduling at the beginning of the simulation, making this a static resource allocation problem. We define a resource allocation to be a complete scheduling of tasks to machines. We perform this research in a *static* and *offline* environment, so that, we can evaluate the resource allocations and analyze the trade-offs between the two objectives. The knowledge gained from studies such as these for a particular system can be used to set the parameters needed for designing dynamic, online, allocation

ISBN: 978-1-61208-226-4

heuristics.

To measure the performance of the system, we examine the makespan of a batch of tasks for a given resource allocation. Makespan is the total amount of time it takes for all the tasks in the batch to finish executing across all the machines. Energy is measured in the number of joules consumed by that same batch of tasks for a given resource allocation. An optimal resource allocation would be one that minimizes both makespan and energy consumed. By adapting the Nondominated Sorting Genetic Algorithm II (NSGA-II) [4] to handle scheduling problems, we are able to create resource allocations that have different makespan and energy consumption values. This set of solutions will then be one basis to analyze the energy and performance trade-offs of the system.

To summarize, in this paper, we make the following contributions:

1) Address the concern of energy efficient computing by modeling the resource allocation problem as a bi-objective optimization problem between minimizing energy consumption and maximizing performance (minimizing makespan).
2) Adapt a well-known multi-objective genetic algorithm to the domain of data center task scheduling.
3) Show that by using different resource allocations, one can greatly affect the energy consumption and performance of a heterogeneous computing system.
4) Construct a set of “Pareto”[5] optimal solutions that can be used to illustrate the trade-offs between system performance and energy consumption, as well as allowing data center managers to select appropriate resource allocations to meet the needs of the specific system.

The remainder of the paper is set up as follows. We discuss the related work in Section II. Section III will describe how we define our bi-objective optimization problem using the NSGA-II. In Section IV, we explain our system model. Our simulation setup is detailed within Section V. Section VI contains our simulation results. Finally, Section VII contains our conclusions and future work for this research.

II. Related Work

In Dongarra et al. [6] and Jeannot et al. [7], a heterogeneous task scheduling problem is modeled as a bi-objective optimization problem between makespan and reliability. This differs from our research because they are not minimizing energy consumption.

The study in Abbasi et al. [8] models a resource-constrained project scheduling problem as a bi-objective problem between makespan and robustness. Abbasi et al. solve this problem using a weighted sum simulated annealing heuristic to generate a single solution. They then adjust the weights to produce different solutions. This is different from our work in that we evaluate our two objective functions independently and generate a Pareto front composed of many different solutions in one run of our algorithm.

A Pareto-ant colony optimization approach is presented in Pasia et al. [9] to solve the bi-objective flowshop scheduling problem. Pasia et al. are minimizing makespan and total tardiness. This differs from our work because they are not considering minimizing energy nor are they using a genetic algorithm to create the solutions.

He et al. [10] develop a bi-objective model for job-shop scheduling to minimize both makespan and energy consumption. There are a couple of differences from our work. The first one is that He et al. model a homogeneous set of machines instead of a heterogeneous set of machines. Second, the algorithm used in He et al. produces a single solution while our algorithm produces a set of solutions.

The goal of minimizing makespan with solutions that are robust against errors in computation time estimates is investigated in Sugavanam et al. [11]. This differs from our work in that we do not consider uncertainties in computation time. Also, Sugavanam et al. are not concerned with energy consumption.

Resource to task matching in an energy constrained heterogeneous computing environment is studied in Kim et al. [12]. The problem is to create robust resource allocations that map tasks onto devices limited by battery capacity (energy constrained) in an ad hoc wireless grid. This differs from our paper because our machines are not energy constrained nor are they in an ad hoc wireless environment. Also the heuristics used in this study only create a single resource allocation, whereas ours creates a set of solutions.

The work in Apodaca et al. [13] studies static resource allocation for energy minimization while meeting a makespan robustness constraint. In contrast to our paper, Apodaca et al. only find a single solution, and we do not place constraints on either objective function.

An energy constrained dynamic resource allocation problem is studied in Young et al. [14]. In this work, the resource allocation must try to finish as many tasks as it can while staying within the energy budget of the system. Our work differs because we are modeling a static environment and have no constraints on how much energy our resource allocations can use.

In Pineau et al. [15], the authors are trying to minimize energy consumption while maximizing throughput. Pineau et al. are modeling a heterogeneous system that executes a single bag-of-tasks application where each task is the same size. To solve the problem, Pineau et al. place a constraint on the throughput objective, and then try to minimize energy while meeting the throughput constraint. While similar to our approach, it differs because we are optimizing for makespan, we model tasks that can differ greatly in size, and we do not constrain either of our objectives.

Mapping tasks to computing resources is also an issue in hardware/software co-design, Teich et al. [16]. This problem domain differs from ours however, because it typically considers the hardware design of a single chip. Our work assumes a given collection of heterogeneous machines.

III. Bi-Objective Optimization Using Genetic Algorithms

A. Overview

It is common for many real-world problems to contain multiple goals or objectives. Often, these objectives work against each other, as optimizing for one objective can negatively impact another objective. This is the case in our research, because it is important for HPC systems to be concerned with both lowering energy consumption as well as increasing overall system performance. In general, resource allocations using more energy will allow one to achieve greater performance, while resource allocations trying to conserve energy will cause the system to have slower performance. In Section III-B, we describe how to determine which solutions should be considered when trying to solve a bi-objective optimization problem. We then briefly discuss the genetic algorithm we have adapted to solve our specific problem in Section III-C.

B. Determining Solutions to a Bi-Objective Optimization Problem

When multiple objectives are present within a problem, it is often the case that there does not exist a single global optimal solution, but rather a *set* of optimal solutions. There is no guarantee one can find the exact solutions within this optimal set, so instead we try to find a set of solutions that are as close to the optimal set as possible. We will call this set of solutions the set of Pareto optimal solutions [5]. This set of Pareto optimal solutions can be used to construct a Pareto front that illustrates the trade-offs between the two objectives.

To understand what it means for a solution to be part of the Pareto optimal set, we illustrate the notion of solution dominance. Dominance is defined as one solution being better than another solution in at least one objective, and better than or equal to in the other objective. To help explain what it means for one solution to dominate another, please refer to Figure 1. Figure 1 shows three potential solutions. The objectives are to minimize energy (along the x-axis), and to minimize makespan (along the y-axis). Let us first examine the relationship between solutions A and B. From the figure we can see that solution B is dominated by A because A uses less energy and has a smaller makespan. Likewise, any solution residing within the upper right (green) shaded region would also be dominated by A. Next, consider solutions A and C. We cannot claim either solution dominates the other because A uses less energy than C, but C has a smaller makespan than A. Thus, for this example, both A and C are solutions in the Pareto optimal set and form the Pareto front. Finally, solution A would be dominated by any solution residing within the lower left (red) shaded area.

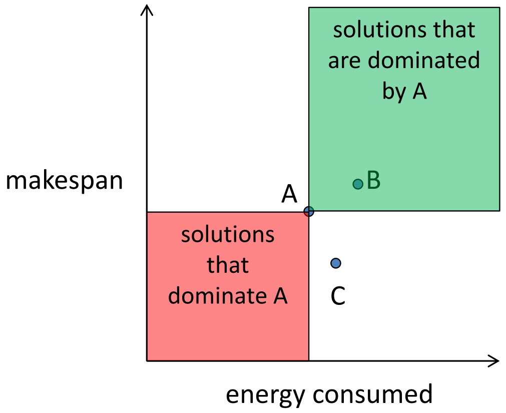


Figure 1. Illustration of solution dominance for three solutions: A, B, and C. Solution A dominates solution B because A has lower energy consumption as well as a lower makespan. Neither solution A nor C dominate each other because A uses less energy, while C has a lower makespan.

C. Nondominated Sorting Genetic Algorithm II Adapted For Resource Allocation

To solve our bi-objective optimization problem, we chose to implement a popular genetic algorithm from the literature, the Nondominated Sorting Genetic Algorithm II (NSGA II) [4]. We will briefly describe the algorithm and how we have adapted it for our use.

The NSGA II is a multi-objective genetic algorithm that uses the idea of solution dominance to create offspring populations, where for our problem domain a population is a set of possible resource allocations. For a given population, the algorithm performs the nondominated sorting algorithm that ranks the solutions within the population based on how many solutions dominate a given solution. Any solution that is not dominated by any other solution is given a rank of one and is part of the current Pareto optimal set. The basic algorithm is outlined in Algorithm 1.

To create the child population in step three, we start with a parent population of size N. From this parent population N crossover operations are performed to create a child population of also of size N. The mutation operation is then performed with a given probability on each chromosome in the child population. If a chromosome is selected for mutation, only the mutated version is kept in the population.

It is important to note the NSGA II is an elitist algorithm as it combined the offspring and parent populations in step six. Elitism means that the algorithm keeps the best chromosomes from the previous generation in consideration for the current generation.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

Algorithm 1 NSGA II algorithm

```
1:  create initial population of $N$ chromosomes
2:  while termination criterion is not met do
3:      create offspring population of size $N$
4:          perform crossover operation
5:          perform mutation operation
6:      combine offspring and parent populations
        into a single meta-population of size $2N$
7:      sort solutions in meta-population using
        nondominated sorting algorithm
8:      take all of the rank 1, rank 2, etc. solutions until
        we have at least $N$ solutions to be used in the parent
        population for the next generation
9:      if more than $N$ solutions then
10:         take a subset of solutions from the
            highest rank number used based on
            crowding distance [4]
11:     end if
12: end while
13: the final population is the Pareto front used to show the
    trade-offs between the two objectives
```

To further explain how the next parent population is created in steps eight and nine, assume we have a parent population with 100 chromosomes and a child population with 100 chromosomes for a total of 200 chromosomes. We want to create a new parent population for the next generation that only has 100 chromosomes. Let us assume, that after step seven (where we have ranked the current populations), there are 60 chromosomes of rank one, 30 chromosomes of rank two, 20 chromosomes of rank three, and 90 chromosomes that have a rank higher than three. First, we will place all the rank one chromosomes into the new population, this will leave room for 40 more chromosomes. Next, we place all the rank two chromosomes into the population, leaving room for ten more chromosomes. Since there are 20 rank three chromosomes, but only room left in the new population for ten chromosomes we must select a subset of the rank three chromosomes to place in the population. These ten solutions will be based on the crowding distance [4] and we will have our full 100 chromosome population.

To use the NSGA II, we needed to encode the algorithm so that it could be used to solve resource allocation problems. This meant we needed to create our own genes, chromosomes, crossover operator, and mutation operator. Genes are the basic data structure of the genetic algorithm, and for our problem each gene represents a task. Within each gene there is a single integer number representing the machine on which the task will execute. Chromosomes represent complete solutions, i.e., resource allocations. Each chromosome is comprised of T genes, where T is the number of tasks the system must execute. The i^{th} gene in a chromosome represents the same task in every chromosome. Each chromosome is individually evaluated with respect to makespan and energy consumption, allowing dominance relationships to be found amongst all the chromosomes within a population.

To allow chromosomes and populations to evolve from generation to generation, we implemented the following crossover and mutation operations. For crossover, two chromosomes are selected randomly from the population. Next, the indices of two genes within the chromosomes are selected randomly. We then swap the genes between these two indices from one chromosome to the other. This operation switches the machines on which the tasks will execute. This potentially allows chromosomes making good scheduling decisions to pass on the useful traits to other chromosomes. For mutation we randomly select a chromosome from the population and randomly select a gene within that chromosome. We then randomly select a machine for that task to execute on.

IV. System Model

A. Machines

Our computing system is modeled as a suite of M heterogeneous machines where each node belongs to a specific machine type μ. Machines are assumed to be dedicated, meaning only one task can be executing on the machine at a time, such as the ISTeC Cray located at Colorado State University [17]. Once a task starts executing it runs until it is finished. Machines of the same machine type are identical to one another. Machine types exhibit heterogeneous performance (i.e., machine type A may be faster than machine type B for some tasks, but slower for others) [18]. Machine types are also heterogeneous with respect to energy consumption (i.e., machine type A may use less energy than machine type B for some tasks, but more energy for others). We implement a heterogeneous behavior for both performance and energy consumption to model a computing system that contains a variety of different resources. Real world systems may be *highly* heterogeneous due to having machines of different ages, varying micro-architectures, subsets of machines that have accelerators, and the inclusion of special purpose machines. Differences in machine components such as memory modules, hard disks, and power supplies also cause systems to be heterogeneous.

B. Workload

We assume we have a static collection of T tasks. Each task t is a member of a given task type. Each task type has unique performance and energy consumption characteristics for executing on each of the machine types. To model the performance of the task types, we assume that the estimated time to compute (ETC) a task of type τ on a machine of type μ, ETC(τ,μ), is given. Entries in the ETC matrix represent the estimated amount of time a task type takes to execute on a given machine type. Research in resource allocation often assumes the availability of ETC information (e.g., [19, 20, 21, 22]). We have provided the analysis framework for system administrators to use ETC information from data collected on their specific systems. This allows for systems of varying size and heterogeneity to be analyzed. For our simulation studies, we have constructed synthetic ETC values modeling real-world systems, but these values can also be taken from various sources of historical data (e.g., [21, 20]).

Similar to the ETC values used for determining compute times, we also assume we have estimated power consumption (EPC) values that tell us the average power a task type consumes while executing on a specific machine type. The EPC values represent the power consumption of a machine as a whole, not just the CPU. Again, we have constructed synthetic EPC values for our simulations, but historical power consumption data could also be used to populate the matrix.

Finally, to obtain the estimated energy consumed (EEC) of a task of type τ on a machine μ we take the product of the execution time and the estimated power consumption, as shown below.

$$EEC[\tau,\mu] = ETC[\tau,\mu] \times EPC[\tau,\mu] \qquad (1)$$

C. Objective Functions

1) Makespan: One objective we are trying to optimize is makespan, which is the total amount of time it takes for all the tasks in the batch to finish executing across all machines. When optimizing for makespan the goal is to minimize the makespan. For a given resource allocation, calculating the makespan of the system requires that we first determine the finishing time of each machine.

To calculate the finishing time of a machine we let the set T_m represent all the tasks in *T* that were allocated to machine *m*, where $t_m \in T_m$. Let the function $\Upsilon(t_m)$ return the task type that task t_m belongs to, and let the function $\Omega(m)$ return the machine type to which machine m belongs. We then calculate the expected finishing time of machine *m* denoted as F_m, with the following equation

$$F_m = \sum_{\forall t_m \in T_m} ETC(\Upsilon(t_m), \Omega(m)). \qquad (2)$$

The makespan for a given resource allocation, denoted ρ, can be found from the machine with the maximum finishing time, and is given as

$$\rho = \max_{\forall m \in M} F_m. \qquad (3)$$

2) Energy Consumption: The other objective we will optimize for is energy consumption. For a given resource allocation, the total energy consumed is the sum of the energy consumed by each task to finish executing. Recall that the amount of energy consumed by a task is dependent upon the machine on which that task is executing. Therefore, the total energy consumed for a resource allocation, denoted E, can be found as

$$E = \sum_{\forall t_m \in T_m, \forall m \in M} EEC[\Upsilon(t_m), \Omega(m)]. \qquad (4)$$

V. Simulation Setup

A. Simulation Environment Parameters

To construct a Pareto front and illustrate the trade-offs between makespan and energy consumption, we conducted numerous simulation trials. For each trial, the number of tasks to execute was set to 1000, with 50 different task types. The number of machines used throughout the simulations was set to 50, with 10 different machine types. The number of tasks per task type and number of compute nodes per compute node type were randomly assigned, and could change from trial to trial.

The ETC values were obtained using the Coefficient of Variation (COV) method from [18], which allows us to model a heterogeneous set of machine types and task types. For our simulations, the mean execution time for the tasks was 10 seconds, and the variance amongst the tasks was 0.1, while the variance amongst the machines was 0.25. These parameters allowed us to model a heterogeneous set of compute nodes.

The EPC values were constructed in a similar manner as the ETC values. Specifically, the mean power consumption for the tasks was 200 watts, and the variation amongst tasks was 0.1, while the variance amongst the machines was 0.2.

For each trial, the genetic algorithm consisted of 100 chromosomes. In the initial population, we used 98 randomly generated chromosomes, and two chromosomes generated using two heuristics based on approaches taken from literature, as discussed below.

B. Seeding Heuristics

The goal of the seeding heuristics are to provide the genetic algorithm with initial solutions that try to optimize the objectives. These seeds can help guide the genetic algorithm towards better solutions faster than an all-random initial population. We chose to implement two greedy heuristics, min energy and min-min completion time, based on concepts found in [23, 24, 25]. The execution times of the greedy heuristics are negligible compared to the NSGA II. Utilizing these seeds in the initial population does not negatively affect the computation time of the NSGA II.

1) Min Energy: Min energy is a single stage greedy heuristic that maps tasks to machines to minimize energy consumption. The heuristic selects a task from the batch and places that task on to the machine that has the smallest energy consumption. For this heuristic, the order in which tasks are mapped to machines does not matter. This heuristic creates a solution that will have the minimum possible

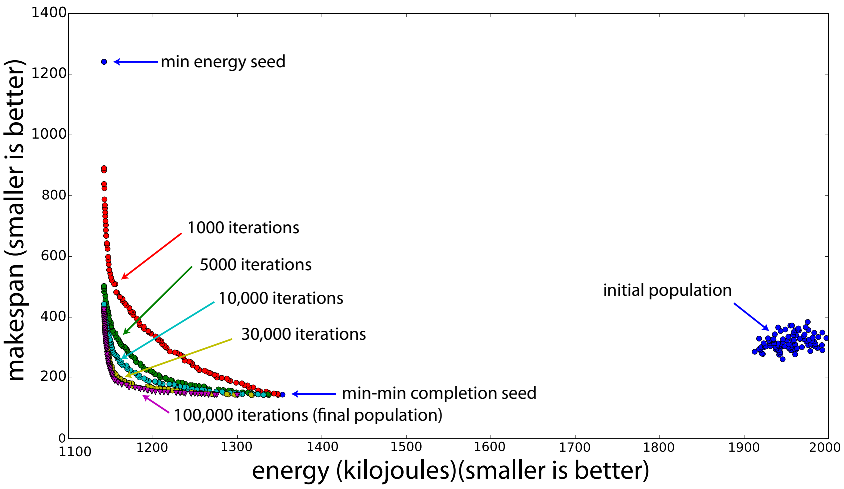


Figure 2. Pareto fronts showing the trade-offs between energy consumption and makespan. Shows the evolution of the solutions through number of iterations completed

energy consumption. For a solution to be more efficient, it must have a smaller makespan.

2) Min-Min Completion Time: Min-min completion time is a two-stage greedy heuristic that maps tasks to machines to minimize the makespan of the system. During each iteration of the heuristic, one task gets mapped to the machine that provides the minimum completion time. One iteration consists of two stages. In the first stage, every unmapped task finds the machine that minimizes completion time. In the second stage, the heuristic selects the task and machine pair from the first stage that has the smallest overall completion time and assigns that task to that machine. This continues until there are no more tasks to map. There is no guarantee that the solution created by this heuristic represents the absolute lowest makespan of the system, so better solutions can potentially improve in both makespan and energy consumption.

VI. Results

Throughout this section, we will only be discussing the results from one simulation trial. We have confirmed that the findings and trends for this trial hold for the other trials we ran. In Figure 2, we show the evolution of the solutions through the number of NSGA-II iterations completed. It is important to note for genetic algorithms, as we increase the number of iterations, the genetic algorithm will in general find new and better solutions; some solutions may remain a member of the Pareto front as we increase the iterations. Each point in Figure 2 represents a complete resource allocation. The set of points corresponding to a given number of iterations form the Pareto front. These points are obtained from the genetic algorithm running through that number of iterations. We see that as the genetic algorithm runs for more iterations, the Pareto fronts are converging towards the lower-left corner. This makes sense because we are minimizing makespan as well as energy consumption. We can also see that for this size problem there is very little improvement to the Pareto front after 30,000 iterations. The size of the problem as well as using the two seeds help the solutions converge in a relatively short number of iterations. Also, observe that both of the seeds provide good starting solutions for the genetic algorithm to evolve from relative to the rest of the initial population.

Although it is useful to see how the solutions evolve over time, the most important information to take away from Figure 2 are the trade-offs between makespan and energy consumed. Figure 3 shows a blown-up plot of the final Pareto front from Figure 2. There are a number of points we can learn from the final Pareto front shown in Figure 3. One such point is circled in red. We can see that around this point there is a definite and visible “knee” in the front. To the left of the knee, small increases in energy consumption result in large decreases in makespan. To the right of the knee we see the opposite, small decreases in makespan result in large increases in energy. Given the information provided in this Pareto front, it is then up to the system manager to select which region of the curve to operate in based on the individual system needs.

To further understand how solutions in the Pareto front differ from one another, we analyzed the individual finishing times and energy consumptions for the 50 machines at five points along the final Pareto front. The five points were the two endpoints of the front, the middle point, and the two points between the middle point and each endpoint, as shown in Figure 4 and Figure 5. The results for machine finishing

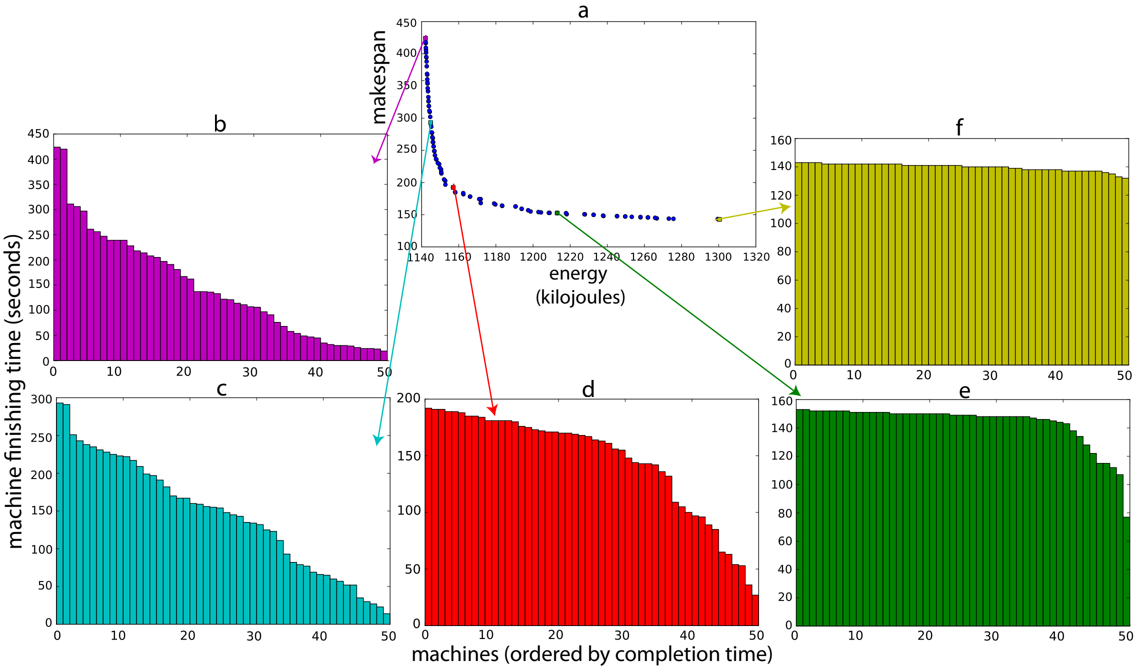


Figure 4. Finishing time of the 50 machines in descending order of finishing time for five solutions from the Pareto front. The y-axis contains different ranges of machine finishing times from plot to plot (to show each plot in greater detail). Subfigure “a” is the same as Figure 3 and has different axis labels from the other subplots. Each subplot b-f has a different ordering of the machines along the x-axis.

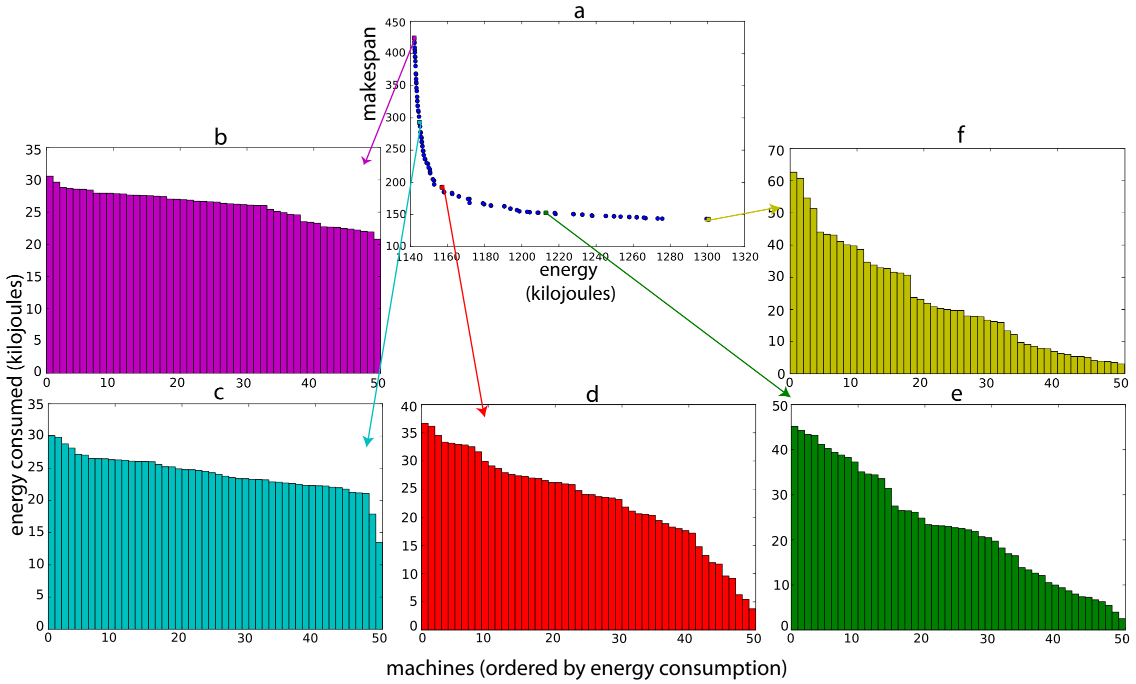


Figure 5. Energy consumption of the 50 machines in descending order of energy consumed for five solutions from the Pareto front. The y-axis contains different ranges of machine energy consumption from plot to plot (to show each plot in greater detail). Subfigure “a” is the same as Figure 3 and has different axis labels from the other subplots.Each subplot b-f has a different ordering of the machines along the x-axis.

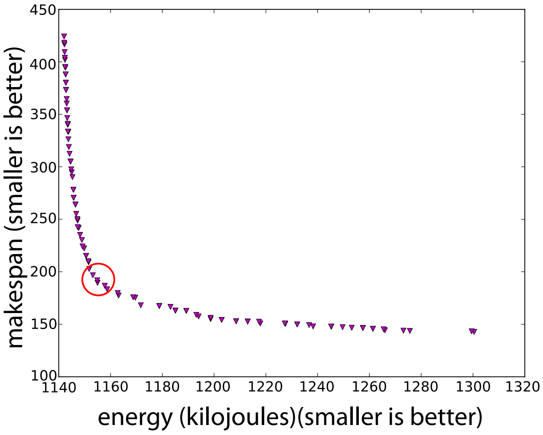


Figure 3. The final Pareto front after 100,000 iterations showing the trade-offs between energy consumed and makespan.

times are shown in Figure 4, while the results for machine energy consumption are shown in Figure 5.

First, we consider Figure 4, which focuses on the finishing times of each machine. Each of the five subplots (b-f) in the figure represents a solution from the Pareto front. On the x-axis of the plots we have the machines sorted by finishing time in descending order, this ordering is different from subplot to subplot. On the y-axis we have the actual finishing time of each machine. Note that each figure (b-f) has different values along the y-axis. In Figure 4.f, we have the solution that provides the lowest makespan. As we can see, the finishing times for all the machines are evenly balanced; this allows the makespan to be small since no one machine is doing a lot more work than the others. As we move left along the Pareto front selected points in Figure 4.a (minimizing energy) we see that the solutions become more and more unbalanced with respect to machine finishing times going from Figure 4.f to Figure 4.e to Figure 4.d, etc. This is because each task type has an affinity for a specific machine type that minimizes that task type's energy consumption.

If we now consider the plots of machine energy consumption in Figure 5 which focus on energy consumption, we see similar trends as before, but in reverse order. This time the machines are ordered in descending order based on energy consumption. Figures 5.b and 5.c are more balanced in terms of energy consumption amongst the machines. This is because this area of the Pareto front focuses on trying to minimize energy and thus makespan is compromised; as we saw in the corresponding makespan plots from Figure 4. By similar reasoning, this is why Figure 5.f is unbalanced. In this region of the Pareto front, makespan is being optimized so tasks are going to have to run on machines that use more energy to lower system makespan.

With the information provided by the Pareto fronts as well as the plots showing the completion time and energy consumption of individual machines, a system manager will be able to analyze the trade-offs between energy consumption and makespan. The system manager can then make a scheduling decision based on the needs of the computing system.

VII. Conclusion and Future Work

As high performance computing systems continue to become more powerful, the energy required to power these systems also increases. In this paper we have developed a bi-objective optimization model that can be used to illustrate the trade-offs between the makespan and energy consumption of a system. Having adapted the nondominated sorting genetic algorithm for use within our domain, we successfully ran simulations that provided us well defined Pareto fronts. We then analyzed five different solutions from the final Pareto front and discussed the differences in their makespan and energy consumption. Given this information a system administrator would be able to pick a specific resource allocation from the Pareto front that meets the energy and performance needs of the system.

There are many possible directions for future work. We would like to enhance our energy consumption model by considering machines that utilize dynamic voltage and frequency scaling techniques to save more energy. We do not currently consider communications within our environment, but the analysis framework we present here could be extended to do this. We would like to try and increase the execution rate and performance of the genetic algorithm by trying numerous parallel techniques. To more accurately model real-world systems, we would like to use probability density functions to model both task execution times and task energy consumption characteristics.

Acknowledgements

The authors thank Bhavesh Khemka and Mark Oxley for their valuable comments on this work. This research was supported by the United State National Science Foundation (NSF) under grant number CNS-0905399, and by the Colorado State University George T. Abell Endowment.

References

[1] Environmental Protection Agency, "Report to congress on server and data center energy efficency," http://www.energystar.gov/ia/partners/prod_development/downloads/EPA_Datacenter_Report_Congress_Final1.pdf, Aug. 2007.

[2] D. Filani, J. He, S. Gao, M. Rajappa, A. Kumar, P. Shah, and R. Nagappan, "Dynamic data center power management: Trends, issues, and solutions," *Intel Technology Journal*, vol. 12, no. 1, pp. 59–67, Feb. 2008.

[3] D. J. Brown and C. Reams, "Toward energy-efficient computing," *Communications of the ACM*, vol. 53, no. 3, pp. 50–58, Mar. 2010.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[5] V. Pareto, *Cours d'economie politique*. Lausanne: F. Rouge, 1896.

[6] J. J. Dongarra, E. Jeannot, E. Saule, and Z. Shi, "Bi-objective scheduling algorithms for optimizing makespan and reliability on heterogeneous systems," in *The Nineteenth Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA '07)*, 2007, pp. 280–288.

[7] E. Jeannot, E. Saule, and D. Trystram, "Bi-objective approximation scheme for makespan and reliability optimization on uniform parallel machines," in *The 14th International Euro-Par Conference on Parallel Processing (Euro-Par '08)*, 2008, vol. 5168, pp. 877–886.

[8] B. Abbasi, S. Shadrokh, and J. Arkat, "Bi-objective resource-constrained project scheduling with robustness and makespan criteria," *Applied Mathematics and Computation*, vol. 180, no. 1, pp. 146 – 152, 2006.

[9] J. Pasia, R. Hartl, and K. Doerner, "Solving a bi-objective flowshop scheduling problem by Pareto-ant colony optimization," in *Ant Colony Optimization and Swarm Intelligence*, 2006, vol. 4150, pp. 294–305.

[10] Y. He, F. Liu, H.-j. Cao, and C.-b. Li, "A bi-objective model for job-shop scheduling problem to minimize both energy consumption and makespan," *Journal of Central South University of Technology*, vol. 12, pp. 167–171, Oct. 2005.

[11] P. Sugavanam, H. J. Siegel, A. A. Maciejewski, M. Oltikar, A. Mehta, R. Pichel, A. Horiuchi, V. Shestak, M. Al-Otaibi, Y. Krishnamurthy, S. Ali, J. Zhang, M. Aydin, P. Lee, K. Guru, M. Raskey, and A. Pippin, "Robust static allocation of resources for independent tasks under makespan and dollar cost constraints," *Journal of Parallel and Distributed Computing*, vol. 67, no. 4, pp. 400–416, Apr. 2007.

[12] J.-K. Kim, H. J. Siegel, A. A. Maciejewski, and R. Eigenmann, "Dynamic resource management in energy constraind heterogeneous computing systems using voltage scaling," *IEEE Transactions on Parallel and Distrubted Systems*, vol. 19, no. 11, pp. 1445–1457, Nov. 2008.

[13] J. Apodaca, D. Young, L. Briceno, J. Smith, S. Pasricha, A. Maciejewski, H. Siegel, S. Bahirat, B. Khemka, A. Ramirez, and Y. Zou, "Stochastically robust static resource allocation for energy minimization with a makespan constraint in a heterogeneous computing environment," in *9th IEEE/ACS International Conference on Computer Systems and Applications (AICCSA '11)*, Dec. 2011, pp. 22–31.

[14] B. D. Young, J. Apodaca, L. D. Briceno, J. Smith, S. Pasricha, A. A. Maciejewski, H. J. Siegel, B. Khemka, S. Bahirat, A. Ramirez, and Y. Zou, "Deadline and energy constrained dynamic resource allocation in a heterogeneous computing environment," *Journal of Supercomputing*, accepted, to appear.

[15] J.-F. Pineau, Y. Robert, and F. Vivien, "Energy-aware scheduling of bag-of-tasks applications on master-worker platforms," *Concurrency and Computation: Practice and Experience*, vol. 23, no. 2, pp. 145–157, 2011.

[16] J. Teich, "Hardware/software codesign: The past, the present, and predicting the future," *Proceedings of the IEEE*, 2012.

[17] "ISTeC cray high performance computing (HPC) system," http://http://istec.colostate.edu/istec_cray/, Aug. 2012.

[18] S. Ali, H. Siegel, M. Maheswaran, D. Hensgen, and S. Ali, "Representing task and machine heterogeneities for heterogeneous computing systems," *Tamkang Journal of Science and Engineering*, vol. 3, no. 3, pp. 195–207, 2000.

[19] "An integrated technique for task matching and scheduling onto distributed heterogeneous computing systems," *Journal of Parallel and Distributed Computing*, vol. 62, no. 9, pp. 1338–1361, Sept. 2002.

[20] A. Khokhar, V. Prasanna, M. Shaaban, and C.-L. Wang, "Heterogeneous computing: challenges and opportunities," *IEEE Computer*, vol. 26, no. 6, pp. 18–27, June 1993.

[21] A. Ghafoor and J. Yang, "A distributed heterogeneous supercomputing management system," *IEEE Computer*, vol. 26, no. 6, pp. 78–86, June 1993.

[22] M. Kafil and I. Ahmad, "Optimal task assignment in heterogeneous distributed computing systems," *IEEE Concurrency*, vol. 6, no. 3, pp. 42–50, Jul.-Sep. 1998.

[23] T. D. Braun, H. J. Siegel, N. Beck, L. L. Blni, M. Maheswaran, A. I. Reuther, J. P. Robertson, M. D. Theys, B. Yao, D. Hensgen, and R. F. Freund, "A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems," *Journal of Parallel and Distributed Computing*, vol. 61, no. 6, pp. 810–837, June 2001.

[24] O. H. Ibarra and C. E. Kim, "Heuristic algorithms for scheduling independent tasks on nonidentical processors," *Journal of the ACM*, vol. 24, no. 2, pp. 280–289, Apr. 1977.

[25] M. Maheswaran, S. Ali, H. J. Siegel, D. Hensgen, and R. F. Freund, "Dynamic mapping of a class of independent tasks onto heterogeneous computing systems," *Journal of Parallel and Distributed Computing, Special Issue of Software Support for Distributed Computing*, vol. 59, pp. 107–131, Nov. 1999.

Design and Implementation of Context-aware Hadoop InputFormat for Large-scale Scientific Dataset

Jae-Hyuck Kwak, Jun Weon Yoon, and Soonwook Hwang
Supercomputing Center
Korea Institute of Science and Technology Information (KISTI)
Daejeon, Republic of Korea
e-mail: {jhkwak,jwyoon,hwang}@kisti.re.kr

***Abstract*—Hadoop is a open-source software framework for the distributed processing of large-scale data analysis across computer clusters using a MapReduce programming model. It is becoming more popular to scientific communities including bioinformatics, astronomy and high-energy physics due to its strength of reliable, scalable data processing. Hadoop InputFormat describes the input-specification for a MapReduce job and defines how to read data from a file into the Mapper instance. Hadoop comes with several implementations of InputFormat. However, it is basically line-oriented and not suitable for context-oriented scientific data processing. In this paper, we have designed and implemented CxtHadoopInputFormat, context-aware Hadoop InputFormat for large-scale scientific dataset. Scientific dataset consists of numbers of variable-length data compartmented by user-defined context. CxtHadoopInputFormat is aware of the context in the scientific dataset and enables Hadoop to be used for distributed processing of context-oriented scientific data.**

Keywords-Data-intensive computing; Hadoop; MapReduce; Context-aware InputFormat

I. INTRODUCTION

Data-intensive computing [1] is one of the evolving scientific area which needs large-scale data analysis. Typical application areas are including bioinformatics, astronomy and high-energy physics. These scientific communities are dealing with high volume of experimental data and need reliable and scalable data management.

Hadoop [2] is a open-source software framework for the distributed processing of large-scale data analysis across computer clusters using a MapReduce programming model. Hadoop is becoming more popular to data-intensive computing application due to its strength of reliable, scalable data processing. However, it has the limitations of data processing, depending on the characteristics of scientific dataset because its default implementation is based on line-oriented and not appropriate for context-based scientific data processing.

In this paper, we have implemented CxtHadoopInputFormat, context-aware Hadoop InputFormat for large-scale scientific dataset. Scientific dataset consists of numbers of variable-length data compartmented by user-defined context. CxtHadoopInputFormat is aware of context information within scientific dataset and enables Hadoop to be used for distributed processing of large-scale scientific dataset.

The rest of this paper is as follows. We introduce Hadoop in Section 2. Then, we examine Hadoop InputFormat in Section 3, what it is and how it works. Section 4 describes the limitation of default Hadoop InputFormat implementation and the necessity of developing context-aware Hadoop InputFormat for scientific data processing. In Section 5, we describe the implementation details of context-aware Hadoop InputFormat implementation for large-scale scientific dataset. Finally, we give conclusion and future work in Section 6.

II. HADOOP OVERVIEW

Hadoop is the Apache project to develop open-source software for reliable, scalable and distributed computing. Basically, it is a framework that allows for the distributed processing of large data sets across computer clusters using a simple programming model. It can be scaled up to thousands of machines, each offering local computation and storage and delivering a highly-available service on top of that.

Hadoop consists of HDFS (Hadoop Distributed FileSystem) and MapReduce. HDFS is a distributed file system, which is highly fault-tolerant and provides high throughput access to application data.

Figure 1 shows HDFS architecture. HDFS cluster consists of a single NameNode and a number of DataNodes.

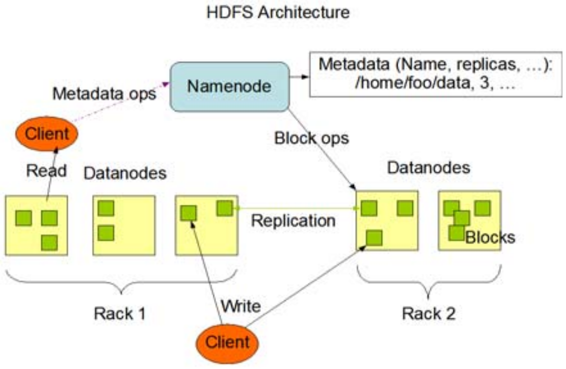


Figure 1. HDFS Architecture

A File in HDFS is split into one or more blocks and these blocks are stored and replicated in a set of DataNodes. NameNode executes file system namespace operations like opening, closing, and renaming files and directories and determines the mapping of block to DataNodes. DataNode is responsible for serving read and write request requests from the file system's clients and performs block creation, deletion, and replication under the NameNode's instruction.

MapReduce is a simple programming model for processing and generating large data sets. It consists of Map and Reduce functions, respectively.

Map(key1, value1) → list<key2, value2>
Reduce(key2, list<value2>) → list<value3>

Figure 2 shows MapReduce dataflow. A map function transforms input data row of key and value to an intermediate output key/value. A reduce function take all values for a specific key, and generate a new list of the final output.

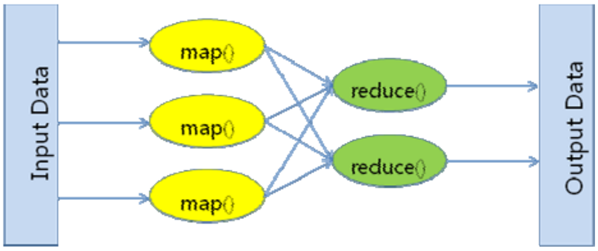


Figure 2. MapReduce Dataflow

The key advantage of the MapReduce is that every Map and Reduce is independent of all other ongoing Maps and Reduces, then the operation can be run in parallel on different keys and lists of data. If you write an application in the MapReduce form, scaling the application to run over hundreds of machines in a cluster is merely a configuration change. Furthermore, Hadoop runs the Map functions on compute nodes where the data lives, rather than copy the data over the network to the program. The output list can then be saved to HDFS and the Reduce functions run to merge the results.

Hadoop is suitable for applications which have large data sets due to HDFS block size (64MB by default) and in-memory representation of the filesystem metadata. Applications that run on HDFS need streaming access to their data sets and write-once-read-many (WORM) access model for files. As a result, Hadoop is suitable for applications which need high-throughput access of large-sized data.

III. HADOOP INPUTFORMAT IMPLEMENTATION

The InputFormat describes the input specification for a MapReduce job and defines how to read data from a file into the Mapper instances. MapReduce relies on the InputFormat of the job to:

1. Validate the input-specification of the job
2. Split-up the input files into logical InputSplits, each of which is then assigned to an individual Mapper.
3. Provide the RecordReader implementation to be used to glean input records from the logical InputSplit for processing by the Mapper.

Main goal of the InputFormat is to divide the input data into fragments that make up the inputs to individual map tasks. These fragments are called "splits" and are encapsulated in instances of the InputSplit interface. Most files are split up on the boundaries of HDFS block size, and are represented by instances of the FileInputSplit class. RecordReader ensures that the splits do not necessarily correspond neatly to line-ending boundaries and do not miss records that span InputSplit boundaries.

Hadoop comes with several implementations of InputFormat. TextInputFormat is the default InputFormat that each record is a line of input. Within TextInputFormat, the key is the byte offset within the file of the beginning of the line and the value is the contents of the line. NLineInputFormat receives a fixed number of lines of input. Like TextInputFormat, the keys are the byte offsets within the file and the values are the lines themselves.

IV. IMPLEMENTATION OF CONTEXT-AWARE HADOOP INPUTFORMAT

Hadoop InputFormat handles input data formats, how it handles the way input data is split into parts for processing by the map tasks, and how it handles the extraction of atomic data from the split data. Figure 3 describes default InputFormat implementation.

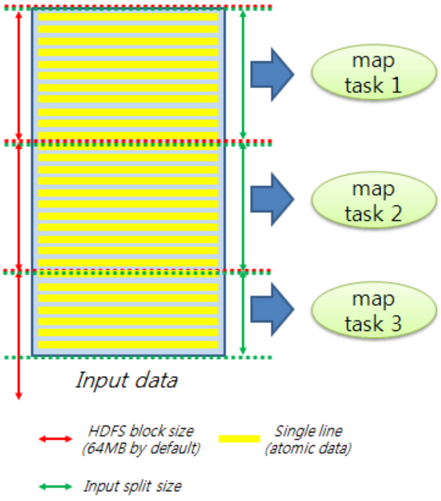


Figure 3. Default Hadoop InputFormat Implementation

In default InputFormat implementation, the atomic data is mostly a line of text (bold yellow line) in a file separated by carriage returns. Hadoop normally processes a very large data file containing a long sequence of atomic data that each can be processed independently. The file is split automatically, normally along HDFS block boundaries (dotted red line) and each split data (dotted green line) is passed to the separate map task for processing.

However, the situation in the scientific data processing could be different in two aspects at least. First, atomic data could be not only a line, but also a multi-line block, the rows of a DB table or a file in the folder. Second, the split data should be made on the boundary of atomic data. It could be not only end of line, but also end of multi-line block, end of row records or end of a file. These aspects are depending on the characteristics of scientific data processing application.

Default InputFormat implementation has some limitations to accommodate this situation and could result in a incorrect data processing. Figure 4 describes a broken context in default InputFormat implementation. In this example, multi-line block (yellow box) should be processed atomically by the map task. However, default InputFormat implementation does not know about this and try to split the data in the middle of atomic data, which is on the boundary of HDFS block. As a result, the data context in the file could be broken.

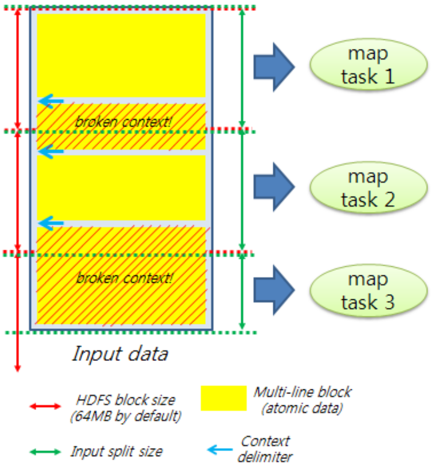


Figure 4. Broken context in default Hadoop InputFormat Implementation

To overcome this situation from scientific data processing on Hadoop, we have implemented context-aware Hadoop InputFormat. Section 5 describes the implementation details about this.

V. Implementation Of Context-aware Hadoop InputFormat

Scientific dataset, especially from large-scale experiment-oriented science such as high-energy physics, astronomy and bioinformatics mostly consists of unstructured data. For example, high-energy physics uses distributed Monte-Carlo simulation and outputs data files what you got from the experiment. Astronomy deals with observation data files which is extracted from the observation instruments.

Hadoop is suitable for large-sized data processing. Scientific dataset could be merged depending on scientific data processing framework. Typical data structure comes with numbers of variable-length data compartmented by user-defined context, depending on the data structure from the field of science application. As mentioned in Section 2, though Hadoop can be used for reliable, scalable and distributed processing of large-scale scientific dataset, default InputFormat implementations are line-oriented, not context-oriented. MapReduce framework reads data from computed input splits and assigned to the Mapper. However, input split is calculated by the formula of file size and HDFS block size, not taking into account the data context in the file. This is likely that file can be split at the wrong position and then cause the incorrect result.

We have implemented CxtHadoopInputFormat, context-aware Hadoop InputFormat. Figure 5 shows CxtHadoopInputFormat implementation.

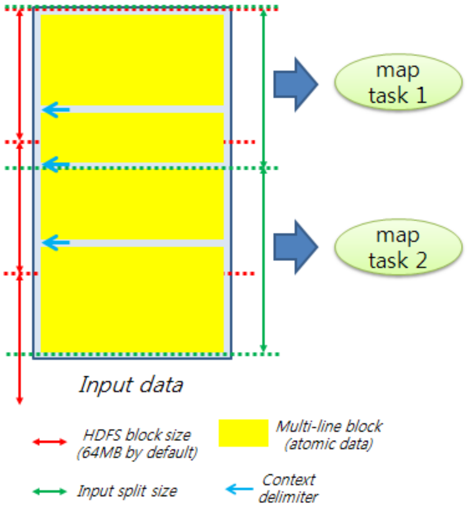

map task 1
map task 2
Input data
HDFS block size (64MB by default)
Multi-line block (atomic data)
Input split size
Context delimiter

Figure 5. Context-aware Hadoop InputFormat Implementation

We assume that scientific dataset has the so-called “context delimiter” that compartmentalized into input records on per-context basis. Our implementation is aware of context delimiter and ensures that input data are split up on the boundaries of the context delimiter, not on the boundaries of HDFS block size.

Figure 6 shows CxtHadoopRecordReader class which implements RecordReader interface. We reuses LineRecordReader class which is the RecordReader implementation used by TextInputFormat. We wrapped the LineRecordReader with our own implementation which converts the value to the expected types.

The next method is called repeatedly to populate the key and value objects. It reads input split in lines and add them to value object until line is about to start by context delimiter. The next method returns value object which contains input records that have the same data context.

```
public class CxtHadoopRecordReader implements
RecordReader<LongWritable, Text> {
    …
    public synchronized Boolean next(LongWritable
key, Text value) throws IOException {
            // read input split in lines
            // add line to value object until line is about
to start by context delimiter
    }
}
```

Figure 6. CxtHadoopRecordReader class

Figure 7 shows CxtHadoopInputFormat class that extends FileInputFormat class. We need to define a factory method for RecordReader implementations to return new instance of CxtHadoopRecordReader class.

```
public class CxtHadoopInputFormat extends
FileInputFormat<LongWritable, Text> implements
JobConfigurable {
    …
    public RecordReader<LongWritable, Text>
getRecordReader(InputSplit genericSplit, JobConf
job, Reporter reporter) throws IOException {
            reporter.setStatus(genericSplit.toString());
            return new CxtHadoopRecordReader(job,
(FileSplit) genericSplit);
    }
    …
    Public InputSplit[] getSplits(JobConf job, int
numSplits) throws IOException {
    // identify hostname, offset and size of data
    // read file in lines
    // generate new input split if line is started by
context delimiter and larger than user-defined
splitSize

    }
}
```

Figure 7. CxtHadoopInputFormat class

The getSplits method gets the desired number of map tasks as the numSplits argument. This number is treated as a hint and a different number of input splits can be made. In our implementation, The getSplits method reads input file from HDFS and calculates input splits on the boundaries of context delimiter if the size of input split is larger than user-defined splitSize.

VI. Conclusion and Future Work

Hadoop is one of the emerging technologies for large-scale data analysis. However, it has some limitations to deal with context-oriented scientific dataset. In this paper, we have implemented context-aware Hadoop InputFormat for processing context-oriented scientific dataset using the Hadoop. CxtHadoopInputFormat is aware of the context in the scientific dataset and enables Hadoop to be used for distributed processing of context-oriented scientific data by spliting up the input data on the boundaries of the context in the scientific dataset correctly.

We have a plan to apply CxtHadoopInputFormat to the representative scientific data processing. We are working with astronomy scientists who want to process data files from SuperWASP project, which is the UK's leading extra-solar planet detection program. They have quite many data files extracted from the observation cameras and are willing to processs them on Hadoop framework to know how it can help large-scale scientific data processing from the field of astronomy. CxtHadoopInputFormat will be helpful to deal with SuperWASP dataset on Hadoop framework.

References

[1] Ian Gorton, Paul Greenfield, Alex Szalay, and Roy Williams, Data-Intensive Computing in the 21st Century, Computer, vol. 41, Apr. 2008, pp. 30-32, doi:10.1109/MC.2008.122.

[2] Apache Hadoop Project, http://hadoop.apache.org

[3] Yahoo! Hadoop Tutorial, http://developer.yahoo.com/hadoop/tutorial/

[4] Tom White, Hadoop: The Definitive Guide (2nd Ed), Oreilly, 2011.

[5] Chuck Lam, Hadoop in action, Manning, 2010.

[6] Jason Venner, Pro Hadoop, Apress, 2009.

[7] Hadoop 0.20.2 API, http://hadoop.apache.org/common/docs/r0.20.2/api/

[8] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung, "The Google file system," Proceedings of the nineteenth ACM symposium on Operating systems principles (SOSP'03), Oct. 19-22, 2003.

[9] Jeffrey Dean and Sanjay Ghemawat, "MapReduce: simplified data processing on large clusters," Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation (OSDI'04), pp. 10-10, Dec. 06-08, 2004.

[10] Hadoop Performance Tuning, Impetus White Paper, http://www.impetus.com

[11] Milind Bhandarkar, Suhas Gogate, and Viraj Bhat, Hadoop Performance Tuning A case study, http://cloud.citris-uc.org/system/files/private/BerkeleyPerformanceTuning.pdf.

Boundaries of Supercomputing

NP Revisited

Lutz Schubert and Stefan Wesner
HLRS
University of Stuttgart
Stuttgart, Germany
{schubert; wesner}@hlrs.de

***Abstract*—Supercomputers can reach an unprecedented degree of scale and miniaturization reaches the quantum level of manufacturing. Non-regarding this progress, however, computing capabilities are and will remain insufficient to meet the demands of many compute intensive scenarios. The major obstacle thereby consists in the non-deterministic polynomial (NP) nature of these problems. Recent research and development seems to have almost forgotten about this intrinsic problem. With this paper we want to remind of the relevance of NP for supercomputing by exploring its impact on future computing development and discuss potential approaches to relieving (not solving) this issue.**

Keywords- high performance computing, NP, scalability, non-determinism, parallel computing

I. Introduction

The number of computational units that are available to a user increased constantly over the recent years: super-computing clusters integrate thousands of processors with high speed interconnects that allow the user to execute large scaling applications. Multicore processors bring parallelism to the common desktop PC and even though scale still ranges in the area of 4 to 8 cores, manufacturers already plan on processors integrating 100s of compute units, so that today's cluster scale will be available for desktop machines in 10 to 15 years' time. In addition, the resources available over the internet and thus principally available for grid, cloud and P2P computing have reached several millions by now [1].

Even though the effective global computational power is high, there still remains a large set of problems that cannot be solved – this includes accurate (long-term) weather forecasting, simulation of the human being, astrophysics etc. All approaches so far provide approximations rather than accurate results – mostly this is due to the size of the respective system, which in the case of weather forecasting and astrophysics is "open", meaning that a potentially infinite number of parameters impact on the computation - classically, this is referred to as dealing with "LaPlace's demon" [2]. It is obvious that "open world" problems are unsolvable due to the physical limitations of the resources – however, most of these problems can be reduced to a subspace in which parameters have only minimal impact (e.g. the gravitational forces across large distances) and thus can be subsumed to a simpler factor or even neglected.

As we will show, the actual main computational problem however consists in the non-determinism of the respective systems, i.e. their "chaotic" nature. Non-regarding the wording, this does not imply that the computation is not causal, but that there is no functional representation of the results for any time t - instead g(t+n) can only be generated by (n) stepwise iterations from g(t). Computation of g typically involves multiple iterations for approximation, which means that the complexity for calculation quickly increases beyond the capacities of existing infrastructures and, in fact, will always exceed these restrictions (chapter II).

All current development concentrates on increasing the number of computational resources by exploiting parallelism – be that on the level of the instructions or on the level of the full processing unit, or on increasing the execution speed by employing specialized accelerators. In all cases, the capabilities effectively increase polynomial whilst the requirement growth remains exponential. In chapter III we will elaborate why such development is insufficient.

This restriction is due to the fact that modern computing still builds on Turing's model, which is strictly sequential in nature. In order to cope with the NP class of problems, a new computing model is required which can deal with the non-deterministic nature of these tasks. In chapter IV, we will discuss what such a model could look like.

We conclude the paper with a discussion on the obstacles towards realizing such a computing model.

II. NP Applications / Non-Determinism

Simulating real world behavior is essential for both academia and industry: not only to understand the mechanics of the system examined, but in particular to be able to modify or replicate it. Thus enabling for example the evaluation of a design prior to its production and to estimate (and contain) impact on the environment, such as oil leakage, air poisoning etc. This equally affects all disciplines, ranging from engineering over natural sciences to social studies.

These disciplines typically investigate different levels of the system, from subatomic (quantum physics) over living creatures (biology, sociology etc.) to galaxies and beyond (astrophysics). And even though the range seems well defined for most disciplines, such as medicine, ranging from individual cells to living beings, there is nonetheless an important reciprocation between most of these levels. This interdependency sometimes leads to the emergence of new, merged disciplines, such as biochemistry, and in other cases one of the major concerns consists in eliminating all influence from other systems, such as in quantum physics. It becomes more and more apparent that effectively all levels have to be considered for accurate predictions and simulation. For example the virtual physiological human (VPH) community aims at simulating the whole human body on all levels (i.e. from cell systems down to molecules) in order to predict e.g. the spread of medicine in the body.

The interest in such research is due to the cross-impact of other domains unto calculations of the respective system. It is a specific aspect of natural systems that they are essentially chaotic, meaning that miniscule changes in parameters lead to completely different results. In other words, minor errors in the data can lead to completely wrong results. Since natural systems are also mostly “open”, there is an infinite number of impact factors. For example, in order to measure an exact kilogram, already the gravity shift due the planetary constellation plays a crucial role. The impact of the combined two factors – openness and chaotic – is also well known as the so-called “butterfly effect” [3].

There is no direct determinism in the underlying functionality, that allows calculation of f(t) for any t directly. This is simply due to this large degree of interactions between all parameters, leaving a potentially infinite number of equations to be solved in each iteration step. Whilst the scope can be reduced according to the number of particles considered in the subspace and according to the strength of coupling, it still leaves the system essentially unsolvable without approximation and optimization.

A. The Impact Of NP: An Example Application

Let us examine this in a simple example of particle collision in an isolated subspace, where each particle can be simulated as (ideal) snooker balls: even though each trajectory can be represented as a vector, collisions need to be checked with all other particles (leaving mechanisms for reducing the search space aside). In the most straight forward approach, we would therefore advance the position of each particle per time step by a safe distance thereby checking with all other particles whether a collision would occur in the respective time step, and reflect it accordingly.

Since at any time a collision may occur (or even several at once), the outcome at any time t depends on the constellation at t-1. In other words $f_n(t+1) = g(f_n(t))$, whereas $f_n(0)$ is the initial constellation and g(x) is the combined collision test and movement over n particles. This means that there are n! equations to be solved in g(x) for one time step.

If each test could be performed by an individual computer in order to achieve maximum performance, adding only one particle more would require n*n! more resources. This can be simply shown: if the complexity for calculating $f_n(t)$ is n!, then the complexity of f_n+1(t) is (n+1)!. Therefore:

$$(n+1)! = n!*(n+1). \qquad (1)$$

To be more concrete: for 1.000.000 (10^6) particles, $8{,}26*10^{5.565.708}$ resources would be required. Just adding one particle to this would lead to additional(!) $8{,}26*10^{5.565.708+6}$ resources. In one cubic meter of air alone there are more than 10^{25} molecules and hence particles to be calculated. There are multiple methods to reduce the number of calculations, such as neighborhood restrictions etc. The main point of this example is not so much to show the complexity of the calculation itself, but the enormous growth of requirements with only slight increments in the problem or data space.

The usual approach to dealing with this amount of equations consists in approximation and result estimation. Chaotic non-deterministic systems can however lead to enormous result deviation if just a single value is changed minimally. In the example of the particle system, we can see how errors can sum up over time, assuming that just a single particle shows a vector deviation (speed or angle) of ε between simulation and real world. We can calculate the average time t_{col} for a collision between two particles to be

$$t_{col} = V / (N\pi D^2 v) \qquad (2)$$

with V being the volume of the subspace, N the amount of particles within V, D the diameter of the particles and v the velocity. For the sake of simplicity, we assume that all particles have a diameter of 1 angstrom (nitrogen has 1.5 angstrom, oxygen 3.6). With a density of 10^{25} molecules/m^3 and the average velocity per particle of 500 m/s (air has roughly 463 m/s at 20° C), this leads to roughly $157*10^6$ collisions per particle and second (for air, this is roughly $5*10^9$).

Ideal elastic collisions preserve the energy of both particles – i.e. given initial vectors $\overline{v}_{1o}$ and $\overline{v}_{2o}$ of two particles, the new vectors $\overline{v}_{1n}$ and $\overline{v}_{2n}$ sum up to the same combined vector. Without going into full detail, one can show that an initial error ε of just one particle, i.e. $\overline{v}_{1real} = \varepsilon*\overline{v}_{1simulated}$ is maintained across all collisions and even transplanted onto all colliding particles [4]. Due to the exponential nature of the collisions, the total (maximum) error after one second is $\varepsilon*2^{157*10*6} \approx \varepsilon*5*10^{2835}$ - for reasons of simplification, we ignore the cancellation of two errors, so that the total error will be slightly lower.

Notably, this does not hold equally true for all problem fields (see section V). Since the calculations themselves are just approximations the effective error is essentially higher. Accordingly, one of the major efforts consists in keeping ε as small as possible. However, limitation of resources and exponential growth of complexity is compensated on cost of precision, thus leading to higher, rather than lower ε and thus to less precise results.

B. Dealing With NP Applications

There are multiple ways to address problems with exponential complexity – typically these consist in restricting the problem space, subsuming multiple equations under one, reducing the precision, using approximation calculations etc. Given the potential error introduced through these methods, the major interest is obviously to employ means that are more precise, thus reducing the risk of an increasing error.

Much effort is vested into finding a representation of the according task in the polynomial problem space, in other words to reduce the specific element of NP to a respective representation in P which would reduce the resource need and complexity by an exponential factor. HPC programmers spend much effort into finding such a reduction, yet that effort is exponential in itself. Even though there is a set of problems which can clearly not be reduced to P, how much of the NP space is identical to P is unsolved as yet [5]. Ideally, the set of NP problems equals the set of P, in which case all major mathematical problems could be calculated in polynomial time.

In addition to “ordinary” NP problems, there is a set of problems which cannot be reduced to P, generally referred to as “NP-hard”. Any problem belonging to this space will exponentially grow in complexity with the degree of desired granularity, respectively data size. In these cases, desired accuracy must be weighed against computational effort. Higher accuracy and larger data set are nonetheless urgent demands from industrial and academic research.

III. THE DEFICIENCIES OF CURRENT DEVELOPMENT

The classical approach to increasing the performance of a processing unit consists in increasing its execution clock rate, e.g. by higher settling rates or extended instruction level parallelism allowing for execution of more operations at once [6]. However, these approaches face multiple problems and effectively do not really improve performance any more [7]. Much optimization is nowadays handled by the compiler, rather than the hardware, even though specialized processing units that offer application specific optimization capabilities are growing in interest (see below).

Current manufacturers extend parallelism on a higher level by exploiting principles that have long been employed in high performance computing: parallel processing units. As opposed to ILP, parallelization on this level implies effectively complete replication of the whole processor, including the logical unit, cache and I/O. Modern multicore processors are thus exactly that: multiple full units within a confined space, though the architecture of interconnects etc. has slightly changed in order to maximize performance.

A. The Limitations of Scale (or Why Multicores Are Not The New Messiah)

Multicores thereby face the same issues as cluster computing: applications simply do not scale to the amount of available resources. In other words, we already have access to more compute units than most programs can effectively use. This is due to two major constraining factors: Amdahl’s law and messaging overhead.

Amdahl’s law generally states that the speedup of parallel program execution is limited by its sequential aspects, or in other words: there are functions and code segments in any program that simply cannot be parallelized. This statement was later turned into a general formula for speedup:

$$\text{speedup} = 1 / (r_s + r_p/n) \qquad (3)$$

whereas r_s denotes the sequential and r_p the parallel portion of a program and n the number of parallel processes, i.e. compute units. Plotting this function clearly reveals how the speedup gained by parallelization saturates with a specific number of compute units (cf. Figure 1.)

It is notable that only applications with a very high degree of scalability (> 90%) can actually make use of the number of processing units offered in large scale clusters and even there, the actual gain is comparatively low. This calculation however does not even consider the impact of messaging overhead or the sequential properties of the individual processes themselves, let alone the ratio between messaging and workload of the processes.

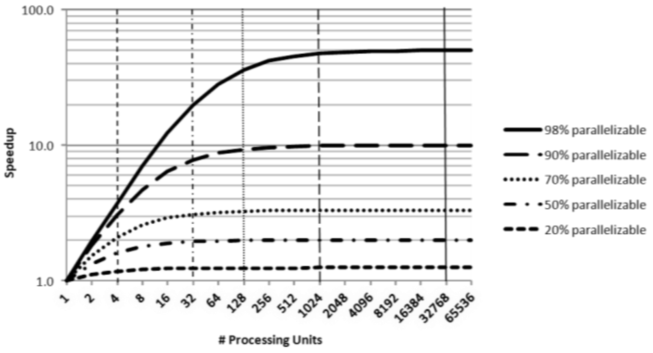


Figure 1. Speedup of an application according to Amdahl’s law

Messaging Overhead thereby is the major problem for all parallel programs – this includes data exchange between processes as much as access to (remote) resources, such as memory or hard drive. Whilst access to memory is specifically defined by the limitations of cache per processing unit, the impact of data exchange between processes is particularly dependent on the distribution of tasks and / or data across processing units:

Let us assume that a given task consists of n iterations (per value) on a dataset with m values. In a straightforward algorithm this means that n*m iterations have to be processed. Assuming that a single processor would take t_{exec} seconds to execute this task, p_{total} processors would take t_{exec}/p_{total} seconds without overhead for distribution and synchronization. In the ideal case, we have n*m resources available, each thus only processing one iteration for one datum. Leaving aside the fact that data needs to be passed between iterations, this distribution is only sensible if the time for gathering the results t_{msg} is higher than the time for execution of one iteration $t_{exec}/(n*m)$. Otherwise, $p_{ideal}=[(n*m)*t_{msg}]/t_{exec}$ defines the number of iterations per processing unit that should at least be executed in order to not reduce performance through messaging – just for result gathering, i.e. for embarrassingly parallel tasks.

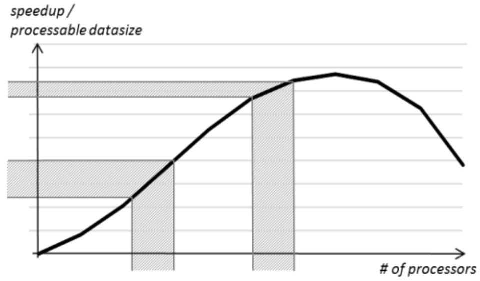


Figure 2. Speedup of program parallelization in relationship to the overhead produced by messaging for synchronization purposes

In real cases, the degree of messaging is obviously defined by the dependency between and across iterations, i.e. which data is required for a single iteration and which is carried across. Typically, a kind of synchronization step is required at least once per iteration. Therefore if the execution of a single iteration becomes too small, messaging produces more load than process execution itself.

Without specifying a concrete unit, as this depends on the respective use case, we can therefore say that performance increases with parallelization up to the point where the messaging overhead exceeds the optimal ratio of workload to messaging of an individual processor. This leads to a effective speedup figure as depicted in Figure 2.

In fact few applications scale well over a few dozen processing units and the best scaling applications are denoted by very little communication. This is particularly true for embarrassingly parallel jobs, such as rendering.

B. Not Enough Resources?

Performance of parallel applications and hence usefulness of large scale supercomputer is naturally limited, basing on the type of calculation to be performed. But it is exactly this type of highly dependent calculations that need to be executed with higher accuracy and over larger datasets in order to satisfy industrial and academic needs. By increasing the workload per processor, by improving the interconnect and by reducing the sequential portion of the program, this saturation point can be pushed to higher scalability numbers - however, an even more common approach consists in combining different levels of process interdependency by linking simulations of different scale [8]. This is in principle identical to segmenting the problem space into sub spaces, thus improving data correctness on a large scale, i.e. across the individual simulations' boundaries, but not within the given segment.

Even if manufacturers could reduce the interconnect problem and the sequential workload, problems with exponential complexity growth would still exceed the number of available resources. As noted, current manufacturers all aim at increasing the number of resources rather than increasing the performance of the individual processing unit – however, the effective gain of this approach decreases with the number of resources, as the amount of data that can be processed in a given timeframe is in direct relationship to the performance of the system. One can interpret Figure 2. also as the process-able size of the dataset over the amount of cores: it can be clearly seen that an increase in the amount of cores in small scale processing systems leads to a stronger increase than in larger scaled ones (gray boxes in the figure).

As opposed to this, increase in clock rate leads to a uniform increment in data-size that can be processed by factor n. More concretely, an increment of the clock rate by factor c also increases the amount of process-able data by c. However, due to the power wall issue, manufacturers must decrease the clock rate when integrating more compute units into a single processor [9] – the main problem for manufacturers is therefore to find the best relationship between amount of cores and clock rate of the individual units. As this relationship is strongly application dependent, there is no clear solution as yet.

Leaving aside the effect of messaging overhead, jitter, limitations of scale etc. and assuming an ideal scalability performance, i.e. where the combined performance is defined over the sum of all processing units' clock-rates:

$$p_{comb} = \Sigma_n p_n \qquad (4)$$

(with n being the number of processing units and p_n the respective performance / clockrate), we can easily show that the effective (combined) performance in current systems does not grow according to Moore's law anymore. Instead the growth has effectively decreased from exponential to linear. Mapping this to the complexity to size ratio (Figure 3.), it is obvious that as we advance the performance of computing systems basically linear (following classical mechanisms), the complexity these systems can handle grows linear too. Implicitly, the size of the according problem space grows only logarithmically.

To summarize:

- the number of necessary resources grows exponentially to the complexity in NP problems
- the performance increase through current large scale systems is naturally limited
- most applications do not meet the scaling capabilities of the underlying hardware

IV. NP PROCESSING

As the interest in more accurate processing of larger data sets increases, so does the pressure on computer manufacturers and application developers to deal with large scale. However, as can be clearly seen from the discussion above, the growth of resources needed exceeds the capabilities exponentially over time – in other words, whilst manufacturers and developers try to push the degree of scalability further up the scale, the need for scale and efficient usage thereof grows faster than manufacturers and developers can achieve. As all the major improvement steps have been taken, the impact of all the minor adjustments that can still be taken in order to increase scalability constantly decreases and becomes more and more use case dependent.

One reason for this deficiency is caused by the limitations of current processors in terms of dealing with non-deterministic problems. More specifically, the strict "sequentiality" and determinism of the Turing machine prevents current computer models to deal with NP problems and the implicit communication. Given the effort to replace a current, well-established computing model with a potentially non-interoperable alternative model, the major question to pose is: what would be the benefit of having machine that can deal (better) with NP?

A. The Impact of Reducing NP to P

As noted above, the hard task for HPC developers consists in finding a representation of the NP problem in the polynomial time space, or at least an approximation. One principle thereby consists in subsuming multiple equations into more global, general equations that, even though they disrespect e.g. interactions between particles in the subspace, still deliver results accurate enough for the purposes of the task. The actual error may be quite substantial in such an environment, where not only the individual deviations sum up, but also the additional error for aggregation and approximation of multiple equations contribute to the overall deviation. In other words the result becomes inaccurate by the factor of potential deviation as calculated above.

If, however, an accurate representation of the respective problem in P space can be found, the error (and hence accuracy) of the result is maintained, i.e. not increased further by the according subsumation and approximation. Since many problems in NP actually belong to the P space, this is frequently possible, though very difficult to achieve. There is furthermore no proof whether all NP problems can thus be represented as P tasks.

Nonetheless, the gain achieved by this complexity reduction is obvious. Figure 3. depicts the complexity reduction and hence the decrease in time to complete the task. It can be noted that for small n (and small c), the complexity of NP problems is actually lower than that of polynomial tasks – this however is quickly surpassed (note the logarithmic scale) with growing n.

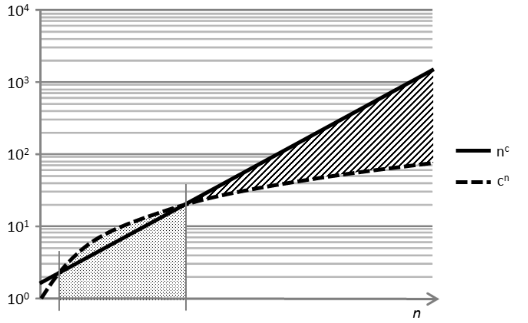


Figure 3. The complexity of n^c versus c^n. The striped area denotes the overhead of NP over P. In the left area, P exceeds NP in complexity

In our particle collision example, the complexity of the original function is n!. Reducing this to a set of equations of complexity n^c would reduce the complexity by factor

$$n! / n^c = (n-1)! / n^{c-1} \tag{5}$$

Whereby we must assume that c is comparatively high, so that an improvement is notable only for large datasets (cf. Figure 3.). However, as noted, it is not always possible, let alone easy to find a P representation for an NP problem, so that ideally, the task is approached the other way round:

B. Classical Approaches to NP Computing

Essentially, if the processing unit itself can process non-deterministic tasks, an NP task on top of it will essentially be executed in order of complexity P. Obviously, this is easier said than done, as otherwise NP processors would have long since emerged on the market. Nonetheless it needs to be stressed again, that the according industrial interest has simply not arisen so far – instead primary focus rested on advancing existing computing types, i.e. Turing machines.

Back in the seventies and eighties, some attempts have been made to instigate “unconventional” processors and in particular to examine the capabilities for NP processing via other means, such as biochemical computing etc. The lacking success is thereby less due to the lack of quality of the results, but rather again for the lacking interest from industrial side: switching from the successful Turing model to an architecture that has neither proven successful nor can be easily applied to common problems would imply too many manufacturing costs and risks.

Many of the according approaches based on the increased scalability of the system rather than the respective capabilities to deal with non-determinism. Reduction in size in comparison to electronic PCs often provokes the misconception that the according capability to deal with NP problems is higher. This builds on the same misunderstanding as the assumption that multicore processors and large scale systems can solve the resource need of the NP space – as has been shown, this is not the case though.

On a similar basis, it is often assumed that quantum computing (QC) would essentially enable non-deterministic computing, and thus solve the NP issue. However, QCs like normal desktop PCs are purely deterministic and sequential in their processing. Even though quantum processes are non-deterministic in nature, the according effects are not exploited in QCs. Miniaturisation and interconnectivity reaches a peak in QC, thus allowing for enormous scale and in theory, quantum processers could perform calculations over full real time numbers as opposed to pure bitwise operations on PCs – however, this faces the same obstacles as analog computing did during the 60ies and 70ies, and is unlikely to be successful due to the same issues [10]. It must be expected though that quantum computers (if ever realized) represents one of the upmost boundaries of scalability or rather of miniaturization.

C. Alternative Paths to Computing

The main reason for this failure to cope with NP consists in our restricted way of thinking in terms of computation, which is still essentially Turing in nature. Processes in nature are therefore examined for how they can be converted into Turing machines, not how they functionally behave. In other words, the processes are interpreted deterministically, without actually exploiting their non-deterministic nature.

Alternative paths to computing must therefore focus on exactly this rather than forcing determinism onto these processes. Instead of exploiting particle collision for message transaction, it can instead be considered as a segment in a chain of non-deterministic events that can be expressed as particle collisions. In the simplest case, a well-defined sub space of particles can simulate the overall behavior of aerodynamics in a larger space etc. Only few approaches try to address the computational nature, in the sense of the underlying processing logic:

A spin-off of MIT for example investigates into processors that replace the underlying binary logic with a probabilistic logic [11]. This does not address non-determinism in the actual processing, yet it allows for more efficient computation of all probabilistic problems, as involved in most NP tasks.

Some natural systems are not only non-deterministic, but are capable of dealing with it. These involve swarms, neural networks and other self-organising systems which are capable of solving problems, such as finding shortest routes towards a food source etc., which belong to NP and lead to high complexity when simulated. Such systems effectively employ mechanisms of dealing with imprecision and fast adaptation in order to approximate a locally optimal solution. A certain degree of scale allows improving optimality further through redundancy, where by it must be noted that the scale is essentially polynomial to the complexity.

Also, the principles of molecular computing are capable of dealing with bounded NP problems in a polynomial number of steps and a limited size of scale [12].

In such systems the problem (and hence the goal) is either implicitly encoded, such as finding the food source for a swarm system, or needs to be painstakingly trained, such as for neural networks. In both cases it generally limits the system to the specific problem domain. The effort of "coding" the problem is hence in most cases NP itself which considerably restricts its applicability.

V. Conclusions: Next Computing Models?

Complexity is a growing problem for computing, that is often ignored or considered to be a pure problem of scale, i.e. that can be overcome by future large scale computing systems. However, this is building on the – generally wrong – assumption that problem complexity scales linear (or polynomial) with the problem space and that performance grows linear and unbounded with the amount of computational resources. Whilst the NP problem space and its implications are actually well known, the actual consequences of it are often ignored or simply forgotten.

The seeming unboundedness of performance through scale however only arises from the fact that many supercomputing problems still move within the area of a positive scale to performance ratio (left gray area in Figure 2.) in the last years, but reaching its peak (right gray area). Personal computers on the other hand just only have reached the beginning of the scale to performance curve, so that still considerable improvements can be achieved through increasing the amount of computational resources [13].

Investing in scale rather than in complete new computing systems is also economically more viable and less disruptive in the short run. As we reach the peak of performance, such a disruptive paradigm switch will become imminent. Current approaches to improving the performance over scale, in particular by reducing the impact of messaging or exploiting more concurrency / asynchronicity, and even quantum computing will only help delaying this problem, i.e. stretching the scale to performance ratio. The main problem can however not be overcome this way, as messaging will always create delay in execution with a certain point of scale ("speed of light is not fast enough" [14]) and the resource need of NP problems grows exponentially and unbounded. Nonetheless, first attempts in that direction need to be seriously undertaken within the near future in order to compensate for the delay in research and development, until more long-term results have been achieved.

That some systems principally can deal better with the NP problem space has already been shown through attempts already initiated back in the 1970ies and 1980ies which base in particular on self-adaptation under uncertain conditions. But also stochastic mechanisms, combinatorial optimization, elastic scale, bounded non-determinism, dynamic segmentation etc. all have introduced principles that significantly contribute to the capability of dealing better with the complexity of such problems and reducing the impact of NP. This however is far from maturity as yet.

Acknowledgment

This work was partially supported by the S(o)OS project under FP7-ICT-2009.8.1, grant agreement no. 248465. See www.soos-project.eu for more information

References

[1] Netcraft: May 2010 Web Server Survey. (2010). http://news.netcraft.com/archives/category/web-server-survey/ [accessed: 2012-06-06]

[2] Laplace, M.D.P.S.: A Philosophical Essay on Probabilities. Forgotten Books (2009)

[3] Lorenz, E.: Atmospheric predictability as revealed by naturally occurring analogues. In: Journal of the Atmospheric Sciences 26: 636–646 (1969).

[4] Leckie, W., Greenspan, M.A.: Pool Physics Simulation by Event Prediction 2: Collisions. In: ICGA Journal 29(1): 24-31 (2006)

[5] Richard, L.: On the Structure of Polynomial Time Reducibility. In: Journal of the ACM (JACM) 22 (1): 155–171. (1975)

[6] Hennessy, J.L., Patterson, D.A.: Computer Architecture: A Quantitative Approach. Morgan Kaufmann (2006).

[7] Manferdelli, J.: The Many-Core Inflection Point for Mass Market Computer Systems. In: CTWatch Quarterly 3(1) (2007). http://www.ctwatch.org/quarterly/articles/2007/02/the-many-core-inflection-point-for-mass-market-computer-systems/ [accessed: 2012-06-28]

[8] Hox, J.J.: Multilevel analysis: techniques and applications. Routledge (2002)

[9] Jones, B.L.: Do Newer Processors Equate to Slower Applications? DevX online publication (2010). Available at: http://www.devx.com/enterprise/Article/34588/1954 [accessed: 2012-06-12]

[10] Aaronson, S.: The Limits of Quantum Computers. In: Scientific American 3 (2008)

[11] Feldman, M.: Startup Aims to Transform Computing with Probability Processing. HPC Wire (2010). Available at http://www.hpcwire.com/hpcwire/2010-08-16/startup_aims_to_transform_computing_with_probability_processing.html [accessed: 2012-04-18]

[12] Beigel, R., Fu, B.: Molecular Computing, Bounded Nondeterminism and Efficient Recursion

[13] Wesner, S., Schubert, L. Kuper, J., Baaij, C.: Convergence of HPC and PC: Really? In: Proceedings of the UK e-Science All Hands Meeting 2010 (2010).

[14] Alessandrini, V.: DEISA Perspectives - Towards cooperative extreme computing in Europe. Fourth EGEE Conference (2005).

ISBN: 978-1-61208-226-4

A Framework for Migrating Traditional Web Applications into Multi-Tenant SaaS

Eyad Saleh, Nuhad Shaabani, and Christoph Meinel
Hasso-Plattner-Institut
University of Potsdam
Potsdam, Germany
{*eyad.saleh, nuhad.shaabani, christoph.meinel*}*@hpi.uni-potsdam.de*

***Abstract*—Software-as-a-Service (SaaS) is emerging as a new model of delivering a software, where users utilize software over the internet as a hosted service rather than an installable product. Multi-tenancy is a core concept in SaaS. It is the principle of running a single instance of the software on a server to serve multiple companies (tenants). Re-engineering traditional web applications from scratch into multi-tenancy requires tremendous efforts in terms of cost, manpower, and time. Thus, we provide a framework to migrate traditional web applications into multi-tenant SaaS. The framework provides a detailed overview of the proposed multi-tenant architecture that helps software architects and developers to migrate their applications into multi-tenancy.**

***Keywords*-Multi-tenancy; Migration; Software-as-a-Service; SaaS.**

I. Introduction: Multi-tenancy Evolution

History has shown that advances in technology and computing changes the way software are designed, developed, and delivered to the end users. These advances yield to the invention of personal computers (PCs) and graphical user interfaces (GUIs), which in turn adopted the client/server architecture over the old big, super, and expensive mainframes. Currently, Fibers and fast internet connections, Service-Oriented Architectures (SOAs), and the high-cost of managing and maintaining on-premises dedicated applications raised the flag for a new movement in the software industry, and the result was the introducing of a new delivery model called Software-as-a-Service (SaaS) [1].

Cloud computing has several definitions, one of those is 'delivering computation to end-users over the Internet'. This computation could be software, hardware, or even information. Users use this computation in a pay-as-you-go model, this means that they will pay for their usage of this computation. For instance, renting a server for two hours or one day, or using a certain financial application for two weeks, without the need to provision a complete data center or buy a full license of the software.

SaaS provides major advantages to both service providers as well as consumers. Service providers can provision a single set of hardware to host their applications and manage hundreds of clients (tenants). They can easily install and maintain their software. As for consumers, they can use the application anywhere and any time, they are relieved from maintaining and upgrading the software (on-premises scenario), and benefit from cost reduction by following the pay-as-you-go model [2].

Multi-tenancy is a requirement for a SaaS vendor to be successful (Marc Benioff, CEO, Salesforce) [3]. Multi-tenancy is the core of SaaS; it is the ability to use a single-instance of the application hosted by a provider to serve multiple clients (tenants). Multi-tenancy is different from multi-instance architecture (Figure 1) where separated instances of the same software are hosted on different servers to serve different tenants.

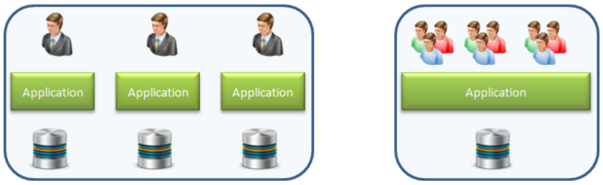


Figure 1. Multi-instance vs. Multi-tenant Architecture.

Software applications had been built for decades into non-SaaS mode, re-engineering or re-designing such applications from scratch requires tremendous efforts in terms of cost, manpower, and time. Therefore, researchers have been proposing several approaches to migrate such applications into SaaS mode.

While migrating the non-SaaS applications into SaaS mode, certain issues need to be considered, such as database architecture, data partitioning, UI customizations, data-model extension by tenants, scalability issues, and workflow management. This paper introduces a framework that supports the migration of traditional web applications into SaaS mode, and discusses certain solutions to the concerns above, as well as introducing new features that could utilize the multi-tenancy mode.

The main contribution of the paper is the framework itself and its components, especially, the business logic configuration and workflow customization, as well as the techniques described to implement the framework, which can be applied in different use cases.

The rest of this paper is organized as the following: Section II discusses the related work. Section III outlines the framework and its main components. In Section IV, we detailed the configuration and customization layer. Section V outlines the use case. Section VI discusses the different database architectural designs. Finally, Section VII concludes the paper and outlines the future work.

II. Related Work

There is much research that has been carried on SaaS and multi-tenancy; however, to the best of our knowledge, there are only a few who proposed a complete framework for migrating traditional web applications into SaaS mode. Moreover, none of the related work touches critical components such as business logic configuration or workflow customization.

A. *SaaS-ization*

Cai et al. [6] propose an end-to-end methodology for SaaS-ization based on identifying isolation points between tenants, such as UI, constant fields, and configuration files. Once these points are identified, the development team can modify these points to be configurable per tenant using a toolkit which was built for the purpose of their work. This approach addresses only the look-and-feel and basic configuration options between tenants, however, business logic and data-model are not covered.

A new approach based on migrating traditional web applications into SaaS automatically and without changing the source code has been proposed by Song et al. [7]. They adopt several technologies to accomplish this goal, mainly (1) page template to fulfill configurability, (2) memory in thread to maintain tenant-info, and (3) JDBC proxy to adopt the modified database. Additionally, they propose a SaaSify Flow Language (SFL) which models and implements the flow of the migration process. Practically, migrating an application from non-SaaS mode into SaaS without having or changing the source code is very hard to achieve. For instance, how the developers are going to handle the session data, how the database changes would be reflected on the Data Access Layer (DAL), even the simple UI components, how it could be managed.

B. *Migration into SaaS*

Bezemer et al. [8] report on a use case of converting an industrial, single-tenant application (CRM) for a company called Exact [9] into a multi-tenant one. They propose a pattern to migrate the application taking into account hardware sharing, high degree of configurability, and shared database instance. They propose three components that need to be added to accomplish the migration process: (1) Authentication module to map end-users credentials to tenants, (2) Configuration module to handle tenant-settings, and (3) database module to adapt insert, modify, and query tenant-oriented data. It is not possible in this approach to have a different business logic or workflow for each tenant.

C. *Customization and Configuration*

Since multi-tenancy is based on sharing the same application by more than one tenant, and the business requirements can vary among those tenants, there is a need to customize the application to appear as tenant-specific, in terms of the look-and-feel, workflow, business logic, etc.

Nitu [10] focuses on the configuration of SaaS applications, basically on user interfaces and access control. They store the configurations in XML files that are injected into applications at runtime. Based on a simple case study they introduce about a university grading system in two Indian universities, they suggest that a tenant should not expect all aspects of the software to be customized. If the tenant needs a complete customized software, then SaaS is not the right choice. This approach is very simplified, and does not cover other critical configuration items, such as workflow and data-model.

Müller et al. [11] identify different categories of customization, such as desktop integration and UI customization. Additionally, they identify two cornerstones for a customizable, multi-tenant aware infrastructure, namely, dynamic instance composition and abstraction from the persistency layer. However, they focus mainly on back-end customization by injecting custom business logic at runtime.

D. *Database Architecture*

Database design is considered as one of the most critical issues in multi-tenant SaaS, because multiple tenants are mapped into one physical database. Therefore, non-traditional concerns are involved, such as data-model extension, workloads, and database scalability.

Several approaches for designing a database for multi-tenant applications from different point of views are discussed by Chong et al. [12], such as completely isolated databases for each tenant versus shared databases with different schemas, or shared databases and shared schemas. Choosing specific approach depends on several factors, such as cost, security level, tenant requirements, scalability options and SLA.

W. Tsai et al. [13] propose a new database partitioning schema to maximize SaaS customization called two-layer schema. Their approach combines read-optimized column store and update-oriented writeable operations.

A new technique for mapping logical schema into physical schema is introduced by Aulbach et al. [14]. They categorize the tables into two types; conventional and chunk tables. The most and heavily utilized parts of the database are placed in conventional tables while the remaining parts are vertically partitioned into chunks. These chunks are folded into different multi-tenant physical tables and joined as needed.

All the aforementioned research proposals strongly contribute to our research; however, in this paper, we tried to introduce a complete framework to migrate traditional web applications into multi-tenancy and introduces new components such as, business logic configuration and workflow customization.

III. The Framework

In this paper, we propose a framework for migrating traditional web applications into SaaS mode as shown in Figure 2.

The process flow of the migrated application will be as follows:

- A user belonging to a certain tenant logs into the system by entering his username and password.
- A dedicated authentication module is used to map this user to the tenant he belongs to, to create a token for the tenant including the Tenant-ID as well as other relevant information (such as locale settings), and finally to pass it to the customization layer.
- In the customization and configuration layer, the UI components, such as logos and colors, the business logic, and workflow configuration data for this tenant are restored, and passed to the application server.
- The DB configuration data will be passed to the DB server for query transformation.
- The application server receives the above specified data from the upper layer and pass it to the run-time customization engine, which integrates all components and lunches the application instance.
- A log service is used to record the application actions and store them in text files.
- A dedicated monitoring service is used to monitor the performance and status of the application, and detects any faults or bad resource usage.

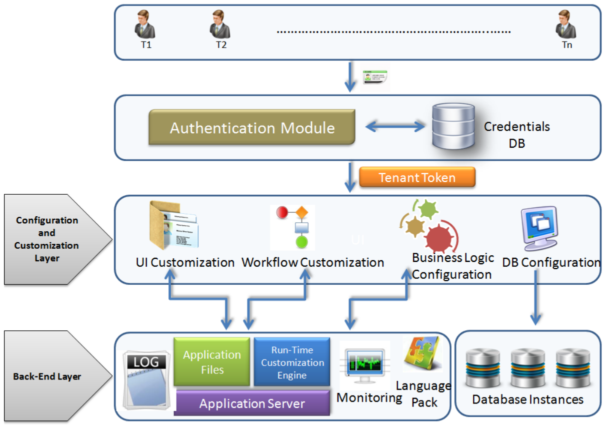


Figure 2. The Architecture of the Proposed Framework

A. The Back-End Layer

The main components of this layer are as follows:

1) The Performance Monitoring Service: Monitoring means getting feedback from the usage of the current software, which leads to enhance and improve the current version of the software.

This service monitors the performance of the software, for example, which queries respond slowly, what are the most heavily-used components of the application, which tenant is overusing the resources, etc.

This collective data will enable the vendor to enhance (upgrade) the software and better isolate tenants to improve the performance.

2) The Log Files: Log files are important to several applications, and more importantly to the multi-tenant ones. They can be used for many reasons, such as, monitor the performance of the application, figure out processing bottlenecks, discover software bugs in the early stages of the release and fix them immediately.

3) The Language Pack: A multi-tenant application is used by several tenants, and they might be from different cultures or having specific language requirements. Therefore, a language pack is additional component the tenant may use to personalize the language settings he needs.

This component is responsible for managing language files, and provide the settings that correspond to the tenant preferences to the run-time customization engine. Several languages could be defined in the language pack, such as English, Arabic, German, Chinese, etc.

IV. The Configuration and Customization Layer

A. User Interface Customization

UI customization means changing the look-and-feel of the application to be tenant-specific. This includes general layout, logos, buttons, colors, and locale settings, such as date and time. To utilize this customization, we propose the usage of Microsoft's ASP.NET master page concept [17].

ASP.NET master page allows the developer to create a consistent look for all pages (group of pages) in the application; one or more master pages could be created for each tenant and used in the application. The master page provides a shared layout and functionality for the pages of the application, when users request any page, ASP.NET engine merges the master page that contains the layout with the requested content-page, and send the merged page to the user as shown in Figure 3.

The application developers would be able to define a master page for each tenant by applying the master page technique, which contains the required layout, color, buttons, and other design components. Moreover, several master pages could be defined for each tenant. Therefore, tenants will have the chance to get benefit of using dynamic look-and-feel. It is worth to mention that applying the ASP.NET concept does not require Microsoft's technologies or platform, our approach aims to apply a similar concept regardless of the technology or the platform.

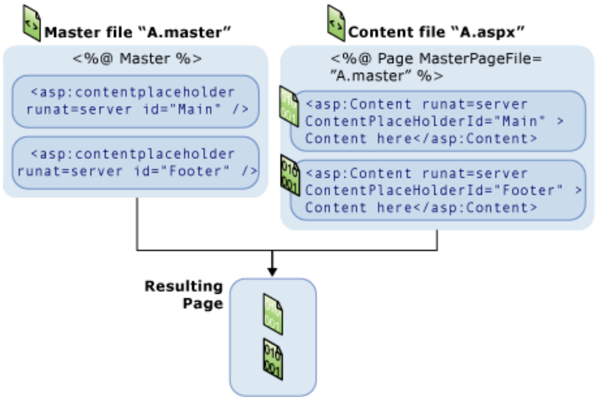

Master file "A.master"
<%@ Master %>
<asp:contentplaceholder runat=server id="Main" />
<asp:contentplaceholder runat=server id="Footer" />
Content file "A.aspx"
<%@ Page MasterPageFile= "A.master" %>
<asp:Content runat=server ContentPlaceHolderId="Main" > Content here</asp:Content>
<asp:Content runat=server ContentPlaceHolderId="Footer" > Content here</asp:Content>
Resulting Page

Figure 3. ASP.NET Master Page [17]

B. Workflow Customization

The workflow of the application might vary from one tenant to another, for instance, a recruitment agency (Tenant A) might wait until they receive a request for a specific vacancy (from a company looking for employees), then start looking for applicants, while another agency (Tenant B) would collect applications, meet applicants, and then short-list them according to their potential, and have them ready for any vacancies from companies looking for employees (Figure 4). Therefore, we assume that customizing the workflow of the multi-tenant software is important. In order to achieve this, two steps are required. First, identify the components of the software that need to be customized, second, change the design of these components to be loosely coupled, thus they can be easily replaced by other versions and integrated with other components, and therefore, each tenant can have his own version of the same component.

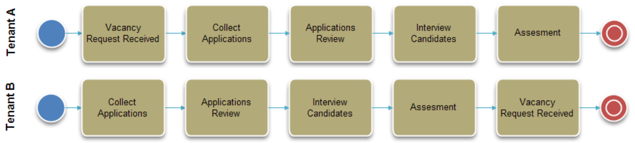


Figure 4. Different Workflow for two recruitment agencies

Changing the design of the entire application into loosely coupled components is a difficult task, on the other hand, customizing the complete workflow of the application may not be necessary since the majority of the application components are normally common among all tenants.

The second step is crucial to make workflow customization successful. The benefit of changing the components selected in step one into loosely coupled is twofold; first, it will maximize the utilization of workflow customization, by allowing the tenant to architect the workflow according to their needs; second, these services/components will become interoperable, thus they can integrate with each other as well as with other services or components from other applications, and even with third party components.

C. Business Logic Configuration

In software engineering, a multi-tier architecture enables developers to divide the application into different tiers to maximize the application re-usability and flexibility. One of the most common implementations of multi-tier is the three-tier architecture, which divides the application into three-tiers, namely, the presentation layer (PL), the business logic layer (BLL), and the data access layer (DAL) (Figure 5).

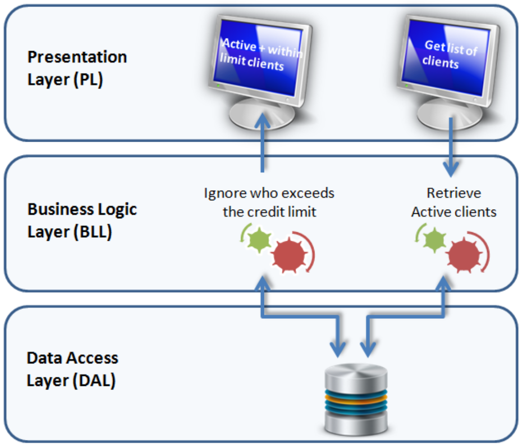

Presentation Layer (PL)
Active + within limit clients
Get list of clients
Business Logic Layer (BLL)
Ignore who exceeds the credit limit
Retrieve Active clients
Data Access Layer (DAL)

Figure 5. Multi-tier Architecture

Business rules are part of the business logic layer (BLL), and these rules varies from one organization to another. For instance, in a travel agency, if a reseller or a client exceeds his credit limit, all his upcoming purchases will be rejected, while another agency, may apply a different rule which state that if the reseller exceeds his credit limit for three weeks without any payment, he will be blacklisted.

In order to achieve and maximize multi-tenancy, we propose that these business rules need to be tenant-specific, this means that the tenant should have the ability to design, apply, and re-configure his own rules at the run-time. Therefore, a tool that offers this feature is needed as a part of the proposed framework.

D. Database Configuration

Database design is considered as one of the most critical issues in multi-tenant SaaS because multiple tenants are mapped into one physical database. Therefore, a robust database architecture must be modelled and implemented.

Consolidating multiple tenants into one database requires changes to the design of the tables and the queries as well, thus, a query transformation is required. For instance, in a traditional hospital management system, a simple query to fetch a patient record would be “select * from patient where

SSN=1234”, while in a multi-tenant system, this will not work, since the “patient” table would have information for many tenants (i.e., hospitals). Therefore, the query should be changed to something similar to “select * from patient where tenant_id=12 and SSN=1234”; in this case, the patient record that belongs to the tenant 12 (i.e., hospital 12) will be retrieved. Based on that, the rules for query transformation should be stored in the database configuration files and restored by the transformation engine when required. Further details are discussed in Section VI.

V. The Use Case: GTDS

Gießen Tumor Documentation System (GTDS) [18] is a software developed mainly for hospitals treating cancer. The project was funded by the federal ministry of health in Germany. GTDS is a powerful and comprehensive system for the documentation of cancer data. GTDS is widely accepted in all over Germany, and it is used in more than 60 hospitals and clinics. From technical point of view, GTDS was initially implemented with Oracle Forms, and its relational data model contains about 400 tables [19]. The base schema for GTDS that we used in the experiments is shown in Figure 6. There are some efforts to redesign the software with modern technologies to produce a standard version that could be used in all hospitals around Germany. Therefore, we are designing and implementing a new modern multi-layer architecture for GTDS. This new architecture will enable us to speed-up the process of migrating GTDS into multi-tenant one.

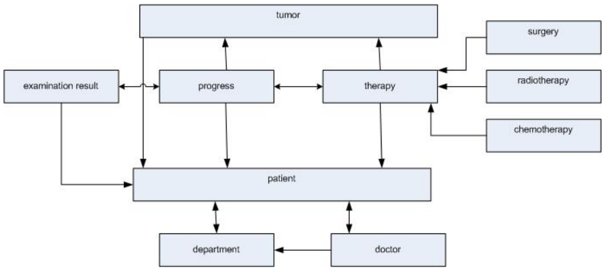


Figure 6. GTDS Basic Schema Architecture

VI. Database Architecture

There are several database architectures that could be used for multi-tenancy, such as completely isolated database for each tenant, shared database with different schemas, or shared database and shared schemas.

The isolated database design makes it easy to extend the data model to meet the requirements of individual tenants. Additionally, it works well in terms of backup and restore, tenant migration between VMs or hosts, and requires only minor changes to the software or database layer. However, it does not reflect the real benefits of multi-tenancy, at least on the database level, such as consolidating several tenants into one physical database. Moreover, it is costly since we need a separate database instance for each tenant, and we will be limited with the number of databases the server can support.

Another approach is the shared database with separate schemas, several tenants will share the same database instance, while everyone is having his own schema. This approach is relatively easy to implement, tenants can extend the data model, and a moderate degree of logical isolation is gained. However, the main limitation of this approach is maintainability, recovering the data for a single tenant in case of failure if a complex task. The database administrator cannot restore the entire database (e.g., from a backup) since this will overwrite the data for other tenants. Therefore, additional efforts need to be taken.

The third approach is shared database with shared schema, where all tenants share the same database with common schema. Obviously, this approach overcomes most of the shortcomings of the aforementioned approaches. However, few questions arise, such as, how to accomplish data isolation between tenants, how to extend the data model since the extension is different from one tenant to another, etc.

The most simple technique to accomplish data isolation is adding a tenant-id column to all tables, then change the queries, functions, and triggers to add the tenant-id filter. However, other advanced techniques can also be used, such as extension table layout, pivot tables, and chunk folding. For more details about these techniques, please refer to [14].

VII. Conclusion and Future Work

Migrating traditional web applications into multi-tenant SaaS poses several research challenges in terms of software customization, database architecture, and isolation between tenants. In this paper, we presented a complete framework to facilitate the migration process. We explored the configuration and customization of the application from several layers, such as UI, business logic, workflow, and database design. Our future work will focus on implementing the framework on GTDS. A set of tools will be developed to facilitate the business logic configuration and workflow customization. Further, security in terms of isolation between tenants will be investigated. Furthermore, how to protect the privacy of tenants will be studied. Finally, several validation experiments of the framework will be conducted on different use cases.

Acknowledgment

The authors would like to thank Mohammed AbuJarour from SAP for his valuable feedback, insights and comments.

References

[1] T. McKinnon: “The Force.com multitenant architecture: understanding the design of Salesforce.com’s internet application development platform”, White Paper, USA, 2008. [Online]. Available:

http://www.developerforce.com/media/ForcedotcomBookLibrary/Force.com_Multitenancy_WP_101508.pdf [retrieved: 08, 2012]

[2] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, and M. Zaharia: "Above the clouds: A Berkeley view of cloud computing". Technical report, University of California, Berkeley, USA, 2009.

[3] D. Woods: Salesforce.com Secret Sauce, Forbes, January 2009. [Online]. Available: http://www.forbes.com/2009/01/12/cio-salesforce-multitenancy-tech-cio-cx_dw_0113salesforce.html [retrieved: 08, 2012]

[4] X. H. Li, T. C. Liu, Y. Li, and Y. Chen: "SPIN: Service Performance Isolation Infrastructure in multi-tenancy environment". Proc. 6th Int. Conf. on Service-Oriented Computing, Sydney, Australia, 2008, pp. 649-663.

[5] C. J. Guo, W. Sun, Y. Huang, Z. H. Wang, and B. Gao: "A framework for native multi-tenancy application development and management". Proc. 9th IEEE Int. Conf. on E-Commerce Technology and The 4th IEEE Int. Conf. on Enterprise Computing, E-Commerce and E-Services, Tokyo, Japan, 2007, pp. 551-558.

[6] H. Cai, K. Zhang, M. J. Zhou, W. G., J. J. Cai, and X. Mao: "An end-to-end methodology and toolkit for fine granularity SaaS-ization". Proc. IEEE Int. Conf. on cloud computing, Bangalore, India, 2009, pp. 101-108.

[7] J. Song, F. Han, Z. Yan, G. Liu, and Z. Zhu: "A SaaSify tool for converting traditional web-based applications to SaaS application". Proc. IEEE 4th Int. Conf. on Cloud Computing, Washington, DC, USA, 2011, pp. 396-403.

[8] C. Bezemer, A. Zaidman, B. Platzbeecker, T. Hurkmans, and A. Hart: "Enabling multi-tenancy: An industrial experience report". Proc. 26th IEEE Int. Conf. on Software Maintenance, Timi oara, Romania, 2010, pp. 1-8.

[9] Exact Website. [Online]. Available: http://www.exact.com [retrieved: 08, 2012]

[10] Nitu: "Configurability in SaaS (software as a service) applications". Proc. 2nd India Software Engineering Conference, Pune, India, 2009, pp. 19-26.

[11] J. Müller, J. Krüger, S. Enderlein, M. Helmich, and A. Zeier: "Customizing enterprise software as a service applications: Back-end extension in a multi-tenancy environment". Proc. 11th Int. Conf. on Enterprise Information Systems, Milan, Italy, 2009, pp. 66-77.

[12] F. Chong, C. Gianpaolo, and R. Wolter: "Multi-tenant data architecture", Microsoft Corporation, http://www.msdn2.microsoft.com, 2006.

[13] W. Tsai, Q. Shao, Y. Huang, and X. Bai: "Towards a scalable and robust multi-tenancy SaaS". Proc. Second Asia-Pacific Symp. on Internetware, Suzhou, China, 2010.

[14] S. Aulbach, T. Grust, D. Jacobs, A. Kemper, and J. Rittinger: "Multi-tenant databases for sSoftware as a service: schema-mapping techniques". Proc. 2008 ACM SIGMOD Int. Conf. on Management of data, Vancouver, Canada, 2008, pp. 1195-1206.

[15] D. Lin and A. Squicciarini: "Data protection models for service provisioning in the cloud". Proc. 15th ACM symp. on access control models and technologies, Pittsburgh, Pennsylvania, USA, 2010, pp. 183-192.

[16] Y. Shen, W. Cui, Q. Li, and Y. Shi: "Hybrid fragmentation to preserve data privacy for SaaS". Proc. 2011 Eighth Web Information Systems and Applications Conf., Chongqing, China, 2011, pp. 3-6.

[17] ASP.NET Master Page on Microsoft.com. [Online]. Available: http://msdn.microsoft.com/en-us/library/wtxbf3hh.aspx [retrieved: 08, 2012]

[18] GTDS Website. [Online]. Available: http://www.med.uni-giessen.de/akkk/gtds/ [retrieved: 08, 2012]

[19] U. Altmann, FR. Katz, and J. Dudeck: "A reference model for clinical tumour documentation". Institute of Medical Informatics, University of Gießen, Germany, 2006.

Bandwidth-Efficient Parallel Visualization for Mobile Devices

Andreas Helfrich-Schkarbanenko, Vincent Heuveline, Roman Reiner, Sebastian Ritterbusch
Engineering Mathematics and Computing Lab (EMCL)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
{*andreas.helfrich-schkarbanenko, vincent.heuveline, roman.reiner, sebastian.ritterbusch*}*@kit.edu*

***Abstract*—For visual analysis of large numerical simulations on mobile devices, we introduce a remote parallelizable visualization method 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 derive optimal impostor placement from a derived metric. Using stochastic usage models, we prove the optimal bandwidth consumption order for choosing corresponding viewport impostor sets, leading to bandwidth-efficient remote visualization concepts for high performance computing simulation results.**

Keywords*-Remote Visualization; Mobile Visualization; Optimal Impostor Placement.*

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, many desirable applications are impeded by limitations of the current hardware [1].

Image-based rendering techniques [2] 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 [3] 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 [4] 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 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.

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 in Section V, which is leading us to the conclusions in Section VI.

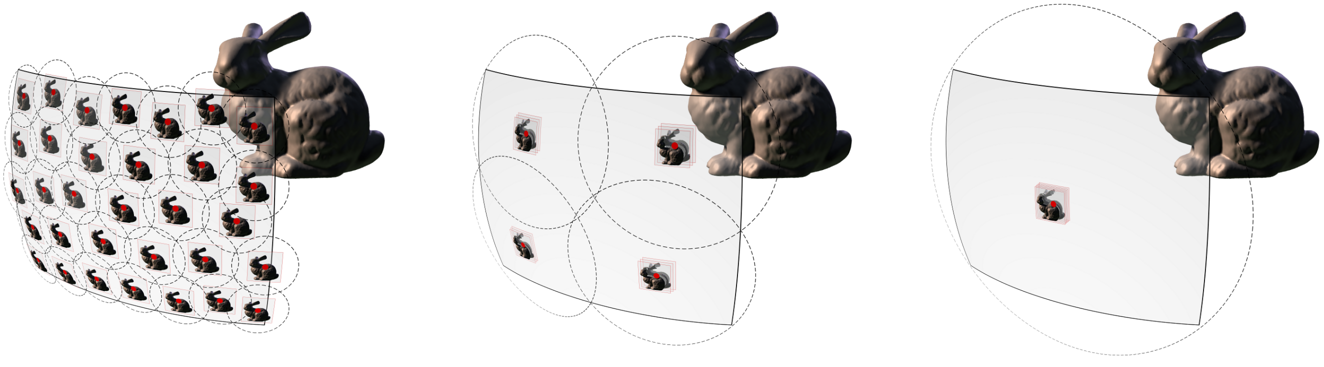
II. Related work

A variety of image-based rendering techniques are reviewed in [4] and [2]. 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 [3], [5], [6]), 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 [3]. Another interesting use of planar impostors is [7], which treats the rendering of volume data on mobile phones.

Several approaches using geometrically more complex impostors can be found in [6], [8] and [9]. In [4], 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



(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 .

costly, the result allows examination from different viewing directions.

A very current example is Street Slide [10]. 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 [4], [5], without further specifying how to choose which viewports to load. A more in-depth analysis on the subject of pre-fetching is given in [11] and [12]. 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 [13] and [14], 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 [2]. 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, modelling 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 [15], 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

$$\pi_d : \left\{ \begin{array}{ccc} \mathbb{P}^3 \backslash \{(0,0,0,1)^\top\} & \longrightarrow & \mathbb{P}^3 \\ x & \longmapsto & P_d x \end{array} \right.$$

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 $(\frac{d}{v_3} v_1, \frac{d}{v_3} v_2, d)^\top$ 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.

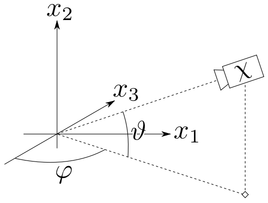


Figure 2. The angles φ and ϑ of a viewport χ

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).

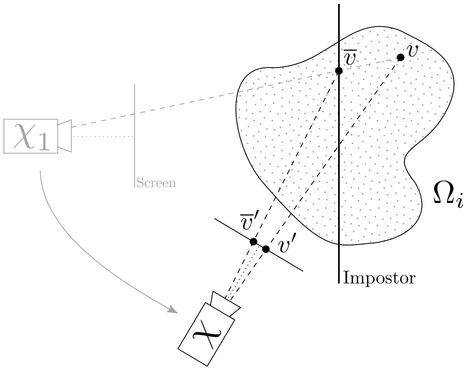


Figure 3. Rendering process for changed viewport

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 point in Ω. As the distribution 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*

$$\begin{aligned} \mathrm{D}(\chi, \chi_1) \quad &:= \quad \sup_{v \in \Omega} \left\| v'(\chi) - \overline{v}'(\chi, \chi_1) \right\|_2 \\ &= \quad \max_{0 \le i \le m} \left\{ \sup_{v \in \Omega_i} \left\| v'(\chi) - \overline{v}'(\chi, \chi_1) \right\|_2 \right\}. \end{aligned}$$

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

$$\mathrm{D}(\chi, \chi_k) = \min_{1 \leq j \leq n} \mathrm{D}(\chi, \chi_j).$$

For $1 \leq k \leq n$ let Ξ_k denote that subset of Λ, on which $\mathrm{D}(\chi, \chi_k)$ is the smallest of all domain errors:

$$\Xi_k := \left\{ \chi \in \Lambda \,\middle|\, \mathrm{D}(\chi, \chi_k) = \min_{1 \leq j \leq n} \mathrm{D}(\chi, \chi_j) \right\}. \tag{1}$$

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 $n \geq 1$. We define the *interaction error* $\mathrm{I} : \Lambda^n \to \mathbb{R}$, where

$$\begin{aligned} \mathrm{I}(\chi_1, \ldots, \chi_n) \quad &:= \quad \int_\Lambda \min_{1 \leq j \leq n} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi) \qquad (2) \\ &= \quad \sum_{j=1}^{n} \int_{\Xi_j} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi). \end{aligned}$$

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

$$\mathrm{I}(\chi_1) \geq \mathrm{I}(\chi_1, \chi_2) \geq \cdots \geq \mathrm{I}(\chi_1, \ldots, \chi_n).$$

Proof: For $1 \leq k \leq n$, it is

$$\begin{aligned} \mathrm{I}(\chi_1, \ldots, \chi_k) \quad &= \quad \int_\Lambda \min_{1 \leq j \leq k} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi) \\ &\leq \quad \int_\Lambda \min_{1 \leq j \leq k-1} \mathrm{D}(\chi, \chi_j) \, \mathrm{d}P(\chi) \\ &= \quad \mathrm{I}(\chi_1, \ldots, \chi_{k-1}). \end{aligned}$$

■

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

$$\Omega := \left\{ (x_1, x_2, x_3, 1)^\top \in \mathbb{P}^3 \,\middle|\, 0 < a_0 < x_3 < a_{m+1} \leq \infty \right\},$$

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 [16, Theorem 3.2]. ■

With this impostor placement, we have the following asymptotic behaviour 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 [16, 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

$$C_1 n^{-1/5} \leq \mathrm{I}(\chi_1, \ldots, \chi_n) \leq C_2 n^{-1/5},$$

for constants $C_{1/2} = C_{1/2}(\Lambda, m)$. Proving these bounds will be the endeavor of the next section.

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 $a_1, \ldots a_5$ and $\bar{a}_1, \ldots, \bar{a}_5$ such that

$$\|A_1(x - y)\| \leq f(x, y) \leq \|A_2(x - y)\| \tag{3}$$

where $A_1 := \mathrm{diag}(a_1, \ldots, a_5)$ and $A_2 := \mathrm{diag}(\bar{a}_1, \ldots, \bar{a}_5)$.

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

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

$$\int_G \|x\|\,\mathrm{d}x \;\geq\; \int_B \|x\|\,\mathrm{d}x.$$

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,

$$\int_{G\backslash B} \|x\|\,\mathrm{d}x \geq \int_{G\backslash B} R\,\mathrm{d}x = R\,\mathrm{vol}\,(G\backslash B)$$

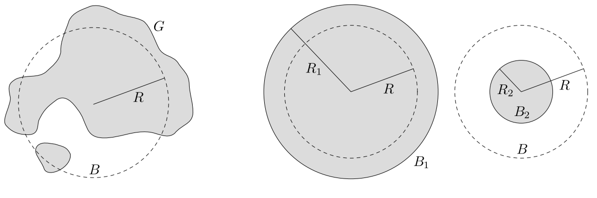
and, conversely,

$$\int_{B\backslash G} \|x\|\,\mathrm{d}x \leq \int_{B\backslash G} R\,\mathrm{d}x = R\,\mathrm{vol}\,(B\backslash G)\,.$$

This implies

$$\begin{aligned}\int_G \|x\|\,\mathrm{d}x &= \int_B \|x\|\,\mathrm{d}x - \int_{B\backslash G} \|x\|\,\mathrm{d}x + \int_{G\backslash B} \|x\|\,\mathrm{d}x \\ &\geq \int_B \|x\|\,\mathrm{d}x - R\underbrace{\big(\mathrm{vol}\,(B\backslash G) - \mathrm{vol}\,(G\backslash B)\big)}_{=0}.\end{aligned}$$

■



(a) Lemma 4. (b) Lemma 5.

Figure 4. Accompanying illustrations for the lemmata.

Lemma 5. Let B and $B_1, \ldots, B_n$ be d-dimensional balls (with respect to a norm $\|\cdot\|$), such that the volume of B is the arithmetic mean of the volumes of $B_1, \ldots, B_n$. Then

$$\sum_{k=1}^{n} \int_{B_k} \|x\|\,\mathrm{d}x \;\geq\; n \int_B \|x\|\,\mathrm{d}x.$$

Proof: We first regard the case $n = 2$. Without loss of generality, let $R_1 \geq R \geq R_2$.

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

$$\begin{aligned}\int_B \|x\|\,\mathrm{d}x &\leq \int_G \|x\|\,\mathrm{d}x \\ &= \int_{B_1} \|x\|\,\mathrm{d}x - \int_B \|x\|\,\mathrm{d}x + \int_{B_2} \|x\|\,\mathrm{d}x.\end{aligned}$$

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

$$\mathrm{I}(\chi_1, \ldots, \chi_n) \;\geq\; C_1 n^{-1/5},$$

with the constant

$$C_1 := \frac{5}{6}\left(\frac{15}{8\pi^2}\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{1/5},$$

where $A_1 := \mathrm{diag}(a_1, \ldots, a_5)$ 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

$$\mathrm{I}(\chi_1, \ldots, \chi_n) = \mathrm{vol}\,(\Lambda)^{-1} \sum_{k=1}^{n} \int_{\Xi_k} \mathrm{D}(\chi, \chi_k)\,\mathrm{d}\chi. \tag{4}$$

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

$$\sum_{k=1}^{n} \int_{\Phi_k(\Xi_k)} \|x\|\,\mathrm{d}x \geq \sum_{k=1}^{n} \int_{B_k} \|x\|\,\mathrm{d}x \geq n \int_B \|x\|\,\mathrm{d}x,$$

ISBN: 978-1-61208-226-4

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

$$\mathrm{I}(\chi_1,\dots,\chi_n) \geq \left(\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{-1} n \int_B \|x\|\,\mathrm{d}x \tag{7}$$

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

$$\mathrm{I}(\chi_1,\dots,\chi_n) \;\geq\; \frac{5}{6}\left(\frac{15}{8n\pi^2}\det(A_1)\mathrm{vol}\,(\Lambda)\right)^{1/5}.$$

■

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 $\chi_1,\dots\chi_n$ for which the interaction error satisfies

$$\mathrm{I}(\chi_1,\dots,\chi_n) \;\leq\; C_2 n^{-1/5},$$

with the constant

$$C_2 := \frac{\pi^2}{36}\frac{(\max\{\bar{a}_1,\dots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)},$$

where $A_2 := \mathrm{diag}(\bar{a}_1,\dots,\bar{a}_5)$ with constants $\bar{a}_i > 0$ as in Proposition 1.

Proof: To begin with, we will proof 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

$$\begin{aligned} \mathrm{I}(\chi_1,\dots,\chi_n) &\leq \mathrm{vol}\,(\Lambda)^{-1}\sum_{k=1}^{n}\int_{\Xi_k}\mathrm{D}(\chi,\chi_k)\,\mathrm{d}\chi \\ &\leq \left(\det(A_2)\mathrm{vol}\,(\Lambda)\right)^{-1}\sum_{k=1}^{n}\int_{\Psi_k(\Xi_k)}\|x\|\,\mathrm{d}x \\ &\leq \left(\det(A_2)\mathrm{vol}\,(\Lambda)\right)^{-1} n\int_B \|x\|\,\mathrm{d}x, \end{aligned} \tag{8}$$

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

$$R = \frac{1}{2}\mathrm{diam}(\Psi_k(\Xi_k)) \geq \max\{\bar{a}_1,\dots,\bar{a}_5\}\frac{\mathrm{diam}(\tilde{\Lambda})}{2n^{1/5}}.$$

With this and Lemma 6 we finally obtain from (8)

$$\mathrm{I}(\chi_1,\dots,\chi_n) \;\leq\; \frac{\pi^2}{72}\frac{(\max\{\bar{a}_1,\dots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)}n^{-1/5}.$$

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)

$$\begin{aligned} \mathrm{I}(\chi_1,\dots,\chi_n) &\leq I(\chi_1,\dots,\chi_{\tilde{n}}) \\ &\leq \frac{\pi^2}{72}\frac{(\max\{\bar{a}_1,\dots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)}\tilde{n}^{-1/5} \\ &\leq \frac{\pi^2}{36}\frac{(\max\{\bar{a}_1,\dots,\bar{a}_5\}\mathrm{diam}(\tilde{\Lambda}))^6}{\det(A_2)\mathrm{vol}\,(\Lambda)}n^{-1/5}. \end{aligned}$$

■

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 $\mathrm{I} = \mathcal{O}(m^{-1}n^{-1/5})$.

VI. Conclusion

In this paper, we developed a mathematical model which allows to measure, analyze and optimize the display error of image-based approximation techniques. The error asymptotics derived for our method based on parallelized rendering shows a clear advantage over traditional remote visualization concepts like Virtual Network Computing (VNC) which, under ideal conditions, represent the scene by one image $m = 1$. In contrast to this, $m = 10$ impostors with $n = 1$ viewport cover the same volume of permissible viewports as $m = 1$ impostors for $n = 10000$ optimally chosen viewport sets. Considering the bandwidth $\mathcal{O}(mn)$ needed for transmission of impostors compared with the error contribution $\mathcal{O}(m^{-1}n^{-1/5})$, the method offers significant decrease of bandwidth consumption, and low latency rendering for the user.

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 [17].

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] 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.

[2] H.-Y. Shum and S. B. Kang, "A review of image-based rendering techniques," in *IEEE/SPIE Visual Communications and Image Processing*, 2000.

[3] S. Jeschke and M. Wimmer, "An error metric for layered environment map impostors," Tech. Rep. TR-186-2-02-04, 2002.

[4] S. Jeschke, M. Wimmer, and W. Purgathofer, "Star: Image-based representations for accelerated rendering of complex scenes," *Proc. of Eurographics*, 2005.

[5] S. Jeschke, M. Wimmer, and H. Schuman, "Layered environment-map impostors for arbitrary scenes," *Graphics Interface*, May 2002.

[6] 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.

[7] 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.

[8] 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. 395ff.

[9] P. Debevec, Y. Yu, and G. Boshokov, "Efficient view-dependent image-based rendering with projective texture-mapping," Berkeley, CA, USA, Tech. Rep., 1998.

[10] J. Kopf, B. Chen, R. Szeliski, and M. Cohen, "Street slide: browsing street level imagery," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–8, 2010.

[11] 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.

[12] 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.

[13] 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.

[14] 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, p. 55ff.

[15] A. Beutelspacher and U. Rosenbaum, *Projective Geometry: From Foundations to Applications*. Cambridge University Press, February 1998.

[16] R. Reiner, "Numerical methods for optimal impostor prefetching in scientific visualization," Diploma Thesis, Karlsruhe Institute of Technology, 2011.

[17] 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.

High-performance Network Accommodation and Intra-slice Switching Using a Type of Virtualization Node

Yasusi Kanada
Central Research Laboratory
Hitachi, Ltd.
Yokohama, Japan
Yasusi.Kanada.yq@hitachi.com

Kei Shiraishi
Telecommunications & Network
Systems Division, Hitachi, Ltd.
Kawasaki, Japan
shiraishi_kei@itg.hitachi.co.jp

Akihiro Nakao
Graduate School of Interdisciplinary
Information Studies, The University of Tokyo
Tokyo, Japan
nakao@iii.u-tokyo.ac.jp

***Abstract*—The architecture for programmable network-virtualization platforms, i.e., the VNode architecture, has been developed in a project called the Virtualization Node Project. This paper introduces a type of physical node called Network ACcommodation Equipment (NACE) to the VNode architecture. NACE has dual roles in this architecture. The first role is as a *network-slice gateway* between an external network (Ethernet/VLAN) and a slice (virtual network). NACE can accommodate a data center or another testbed in a slice with high-performance (up to 10 Gbps) data-format conversion. The second role is as a special type of virtualization node that implements *intra-slice virtual switch* by using Ethernet hardware, which can replace software-based switching using a VM or a network processor. These roles are modeled as a node sliver (virtual node) with a gateway function and a node sliver with a switching function (i.e., a switch node-sliver), and these node slivers are specified by using XML. These functions were evaluated by using two testbeds, and the evaluation results confirm that both functions work correctly and perform well in terms of delay and packet loss.**

Keywords—network virtualization; virtual network; network accommodation; network-slice gateway; virtual switch; intra-slice switching

I. Introduction

In Japan, several projects targeting new-generation networks (NwGN) have been conducted [Aoy 09] [AKA 10]. These projects aim to develop new network protocols and architectures (i.e., the “clean slate” approach [Fel 07]) and to develop various applications that are difficult to run on IPs but work well on NwGNs. In one of these projects, “Virtualization Node Project” (VNP), Nakao, et al. [Nak 10] has developed a virtualization-platform architecture [ITU 12] called VNode architecture.

There have been two issues concerning the original implementation of VNode architecture [Nak 12a][Nak 12b]. The first issue is that the original implementation does not have a high-performance gateway between a slice and an external network. This architecture has a type of gateway called an access gateway (AGW) to accommodate end users’ computers in slices. However, in an AGW, because the data rate between a user and an AGW is assumed to be 1 Gbps or less, and the security of this link should be guaranteed by IPSec, the performance of the AGW is not sufficient for accommodating a data center or high-performance testbed in a slice. In addition, because an AGW is optimized to accommodate user terminals, it is not the best means for accommodating a network with many hosts.

The second issue is that this implementation does not fully utilize the performance of the hardware component of a VNode, especially the high-performance Ethernet switching function. A VNode implements node slivers (virtual nodes) using Linux virtual machines (VMs) or network processors (NPs). The node slivers can be freely programmed, so any protocol (either IP/Ethernet based or non-IP/non-Ethernet based) can be implemented. However, the performance of the node slivers is not optimum. Although a VNode contains a high-performance L3 switch with 10-Gbps Ethernet interfaces and 190-Gbps (or more) switching capacity, it has been hard to achieve multi-gigabit intra-slice switching speed by using VMs and/or NPs.

To address these issues, a type of physical node called Network ACcommodation Equipment (NACE or NC) has been developed. NACE has dual roles. The first role is as a high-performance gateway between a slice and an external network. By means of this gateway function, NACE can accommodate a data center or another testbed in a slice with high-performance (up to 10 Gbps) data-format conversion. By using NACE, external networks can also provide services to slices, and slices can provide services to external networks. The second role is as a special type of virtualization node that implements an intra-slice switching function using Ethernet hardware. This function makes it possible to switch not only Ethernet packets but also packets of arbitrary format with arbitrary type and size of addresses that can be mapped to Ethernet MAC addresses.

The rest of this paper is organized as follows. Section II summarizes the virtualization platform and the slice model developed in VNP. Section III outlines NACE. Sections IV and V describe two roles of NACE, i.e., network-slice gateway and intra-slice switch, respectively. Section VI describes potential usage of NACE, including a data-center or testbed gateway, an intra-slice Ethernet switch, and an intra-domain slice interaction. Section VII evaluates NACE by using two slices on test beds, and Section VIII concludes this paper.

II. Virtualization Platform and Slice Model

This section explains network virtualization, the structure of virtualization platform (i.e., physical network), and the structure of virtual network.

A. Network Virtualization

When many users and systems share a limited amount of resources on computers or networks, virtualization technology creates an illusion that each user or system owns resources of their own. Virtualization technology was initially developed as virtualization of computer memory

ISBN: 978-1-61208-226-4

and multiplexed (time-sharing) use of computational resources such as CPU time. However, recently, a whole computer can be virtualized as a VM.

Concerning networks, wide-area networks (WANs) are virtualized by using virtual private networks (VPNs). When VPNs are used, a physical network can be shared by multiple organizations, and these organizations can securely and conveniently use VPNs in the same way as virtual leased lines. Nowadays, networks in data centers are virtualized by using VLANs, while servers are virtualized by using VMs.

Many programmable virtualization-network research projects have been carried out, and many models, including PlanetLab [Tur 07], Virtual Network Infrastructure (VINI) [Bav 06], Global Environment for Network Innovations (GENI) [GEN 09], and Genesis [Kou 01], have been proposed. Slices are created by network virtualization using a *virtualization platform* that operates the slices.

In VNP, Nakao et al. [Nak 10][Nak 12b] developed network-virtualization technology that makes it possible to build programmable virtual-network environments in which slices are isolated logically, securely, and in terms of performance (QoS) from one another [Kan 12b]. In these environments, new-generation network protocols can be developed without disrupting other slices.

B. Structure of Virtualization Platform

A virtualization-platform domain is managed by a domain controller (DC) and has two types of nodes (**Figure 1**).

- *VNode* (virtualization node) is a physical network node that forwards packets on the platform. Each packet in the platform contains a virtual packet in a slice (as the payload).
- *Gateway*, of which access gateway (AGW) is one type, is a network node that forwards packets from the platform to user terminals (PCs) or another network, or vice versa.

A domain may contain conventional routers or switches that do not have virtualization functions. VNodes are connected by tunnels using a protocol such as Generic Routing Encapsulation (GRE) [Far 00]. A virtual network with free topology, which is not constrained by the topology of the physical network and does not depend on the specific functions of the nodes in between, can therefore be created. A VNode can operate as a router or a switch for platform packets, so it can be deployed in conventional networks.

Each VNode consists of the following three components.

- *Programmer* processes packets on the slices. Slice developers can inject programs into programmers.

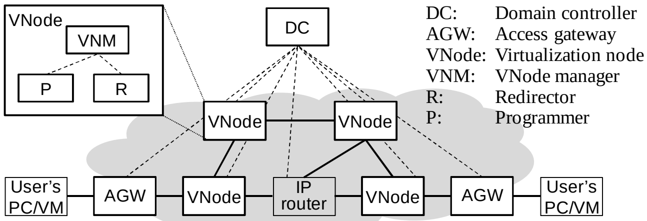


Figure 1. Physical structure of virtualization platform

- *Redirector* forwards (redirects) packets from another VNode to a programmer or from a programmer to another VNode.
- *VNode manager* (VNM) is a software component that manages the VNode according to instructions from the DC.

C. Structure of Virtual Network

In the virtual-network model developed by VNP, a virtual network (or a collection of resources in a virtual network) is called a *slice*, which consists of the following two types of components (**Figure 2**) [Nak 10][Nak 12b].

- *Node sliver* (a virtual-node resource) represents computational resources that exist in a VNode (in a programmer). It is used for node control or protocol processing of arbitrary-format packets, and it is generated by slicing physical computational resources.
- *Link sliver* (a virtual-link resource) represents resources of a (layer-2) virtual link that connects two node slivers. In the VNode architecture, any IP or non-IP protocols can be used on link slivers. A link sliver is mapped on a physical link between two VNodes or a VNode and a gateway, and it is generated by slicing physical network resources such as bandwidth.

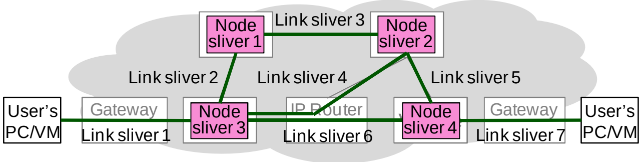


Figure 2. Example of slice design

The DC of a domain receives a slice design given by an XML-based *slice definition*. It then distributes the slice definition to each VNM, which sends necessary sliver definitions to the programmer and the redirector: the programmer receives information required for node-sliver configuration, and the redirector receives the information required for configuring link slivers.

III. Outline of NACE

This section outlines NACE; namely, describes roles, requirements, and the structure of NACE.

A. Roles of NACE

NACE has two roles. The first role is as a gateway between a slice and an external network. NACE can accommodate an Ethernet-based network or a bundle of VLANs in a slice. Thanks to network-slice gateway function, NACE can accommodate a data center or another testbed in a slice with high-performance (up to 10 Gbps) data-format conversion. This function has been implemented as an extended function of a node sliver.

The second role is as a special type of VNode that implements an intra-slice virtual switching function for packets of arbitrary format by using Ethernet hardware. This function is intended to make it possible to switch packets of arbitrary format. However, in the current version of NACE, only the Ethernet protocol can be used in slices. This function has been implemented as a special type of node sliver called a switch node-sliver.

These two roles (described in detail in Sections IV and V) have been given to the same equipment because both require similar hardware and software components, including data converter between VLANs and GRE/IPs (GRE link sliver) and management software.

B. Requirements

As described above, the two roles require gateway and switching functions, which share two common requirements. One requirement is that methods for modeling and specifying these functions must be developed. As parts of the VNode architecture, these functions must be modeled as a combination of node slivers and link slivers in the slice definition and specified by using XML. NACE must implement these functions using VLAN functions of an Ethernet switch. However, this implementation is out of scope of this paper.

The other requirement is that high-performance data-conversion functions, which enable 10-Gbps wire-rate accommodation and switching, must be developed. The data conversion is required because both network-slice gateways and intra-slice virtual switches handle two different data formats. The former must convert packets from the virtualization-platform format (GRE/IP) to the external format (VLAN), and vice versa. The latter must convert packets that can be switched by the hardware from the platform format to a VLAN format, and vice versa. In addition, because both functions are modeled using the same XML-based language and managed in the same way by the management system, the required software functions are also similar. Therefore, although the two roles are very different, the same components can be used for both roles.

C. Structure of NACE

The above requirements are satisfied by NACE. The current version of NACE is a remodeled version of the VNode (**Figure 3**). Similar to a normal VNode described in Section II, a NACE consists of three components. Two of them are almost the same as those in a normal VNode, i.e., a *redirector* and a *VNM*. However, the other component, a programmer, is replaced by a *pseudo programmer manager* (PPM). The redirector in the NACE consists of the three components: Ethernet switch, node manager, and service module cards.

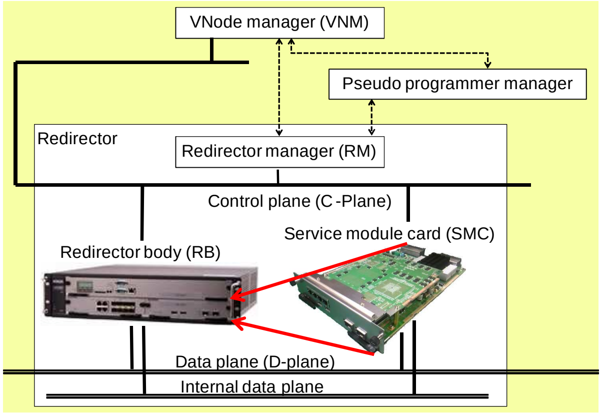


Figure 3. Structure of NACE

- *Redirector body* (RB) is a layer-3 (L3) switch, i.e., an IP/Ethernet node; it has both VLAN and IP-routing functions. A carrier-class high-end switch is used to build a VNode. To make the NACE physically smaller and portable, a 2.5U-size L3 switch is used for the RB, which is much smaller than the switch used in a normal VNode.
- *Redirector manager* (RM) is a software component in a Linux-based server. RM has link-sliver management functions; that is, it creates, modifies, and deletes link slivers. It also has a range of equipment-management functions for the redirector. The RM has an XML-RPC [XML 04]-based control API, and the VNode manages the redirector using this API.
- *Service module card* (SMC) is an add-on card installed in the RB. An SMC is programmable because it contains a 10-Gbps-class network processor. It is used for the bidirectional conversion. Similarly to a normal VNode [Kan 12c], NACE requires packet-format conversion between slice-internal formats and external formats. The internal format is GRE-based, and the external format is VLAN-based.

The redirector is connected to the following networks. *Data plane* (D-plane) is a 10-Gbps network for sending and receiving data packets between VNodes. IP/Ethernet is used in the current implementation as described above. *Control plane* (C-plane) is a network (VLAN) between a VNM and a redirector in a VNode and between VNMs. The XML-RPC-based API between VNMs (and between VNodes) and a redirector uses the C-plane. *Internal data plane* is a closed network (VLAN) in a VNode.

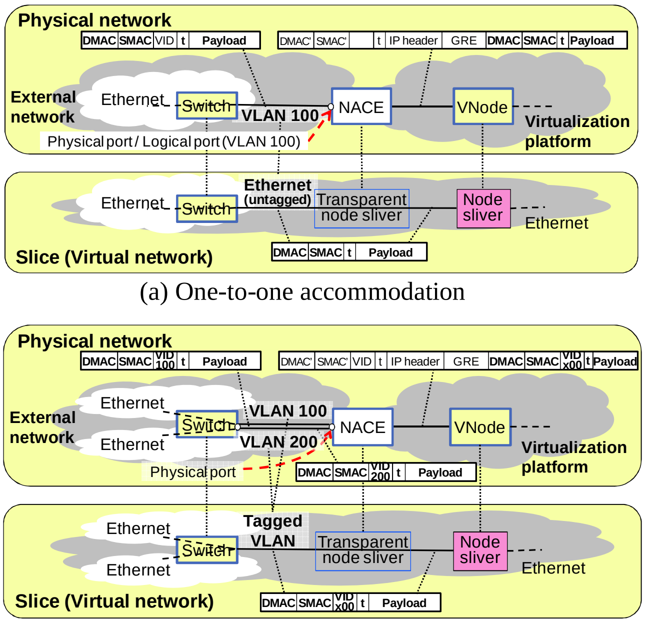
Finally, the PPM is briefly explained. NACE should be programmable, but the programmability of NACE should be different from that of a normal VNode because NACE works as a network gateway or a special type of VNode. To reduce the size of NACE, programmer hardware, which is not required for the gateway and switching functions, was not added to this NACE implementation. Instead, a pseudo (software only) programmer for communicating with the VNM and RM was implemented in the redirector.

IV. NACE AS GATEWAY

NACE can function as a gateway that connects slices with external networks such as the Internet, an Ethernet-based network, or any other type of network. Slices on a virtualization network can be used through the Internet and access gateways. If NACE does not exist, slices are basically closed; that is, they cannot exchange packets with other networks such as the Internet. However, NACE enables external networks to provide services to slices and enables slices to provide services to external networks.

In the future, NACE will be able to accommodate slices and external networks with non-IP/non-Ethernet protocols. Although the VNode architecture supports non-IP/non-Ethernet protocols, the external networks are almost always IP and/or Ethernet networks. Therefore, network accommodation with high-speed data-format conversion between these protocols should be implemented. However, currently, NACE only supports Ethernet (and IP/Ethernet) networks.

The gateway function is modeled in the following way. A network-slice gateway can accommodate multiple external



(a) One-to-one accommodation

(b) Many-to-one accommodation

Figure 4. NACE accommodation types and packet formats

networks (i.e., only VLANs in the current version). A gateway should provide two types of network accommodation (**Figure 4**).

- *One-to-one accommodation* connects an external network (i.e., a VLAN or an Ethernet network) to a slice (Figure 4(a)). Node slivers on the slice cannot access non-virtualized information (i.e., a VLAN tag) on a packet that arrives at the external network, but they read an untagged packet. Subtypes of one-to-one accommodation are explained below.
- *Many-to-one accommodation* connects multiple networks (i.e., VLANs with different VLAN IDs) to a slice (Figure 4(b)). Node slivers on the slice can read and handle the network identifier (i.e., the VLAN IDs) on a packet as is. The physical and virtual packet formats used in the current NACE implementation are also shown.

One-to-one accommodation can be split into two subtypes.

- *Physical accommodation* connects an external network through a specified physical port of a network interface. The port number is specified in the slice definition (as part of a special-purpose node-sliver for network accommodation). In the current NACE version, the packet format of the external network may be tagged VLAN or untagged Ethernet. If a tagged format is used, the VLAN ID must be specified in the slice definition (in the node sliver). If no VLAN ID is specified, the untagged format is used.
- *Logical accommodation* connects an external network through a specified logical network name (i.e., a VLAN ID). In accordance with the configuration of the platform, the network can be connected to any physical port that is available, and packets may be either tagged or untagged. The VLAN ID is specified in the slice definition.

To accommodate an external network in a slice, the slice developer must include a node sliver in the XML-based slice definition. A node sliver usually contains a link to a programmable component such as a VM image. However, in this version of NACE, no node sliver can contain a programmable component because the NACE just forwards packets from the slice to the external network, or vice versa, as is, unless packet-header conversion is required for virtualization.

A node sliver with a gateway function is specified using an interface specification in the node-sliver definition. An example of interface definition is shown below.

```
<interfaces>
   <interface name="ExtIF" type="VLAN">
      <params><param key="VLANID" value="100" />
         <param key="port" value="1/2" />
      </params></interface></interfaces>
```

In the interface tag in this definition, the name property **"ExtIF"** specifies the identifier of the accommodation, and the type property **"VLAN"** specifies the type of external network. The above interface definition specifies a one-to-one accommodation, and **"1/2"** is used for the port number, and **"100"** is used for the VLAN ID. That is, only packets with VLAN ID 100 that arrive at port 1/2 are forwarded to the slice, VLAN ID 100 is added to packets that arrive at the node sliver, and the packets are forwarded to the port 1/2. The accommodation type is one-to-one because a specific port number is specified, but it is many-to-one if the port number is specified as **"all"**. The accommodation subtype is physical because a port number is specified.

These parameters are validated by the redirector and can be validated by the DC. However, some malicious access or erroneous accommodation is still possible. This problem should be solved in the future.

A node sliver that contains accommodation parameters is virtually managed by the PPM in NACE; the DC sends the node-sliver definition to the PPM through the XML-RPC interface. Because the accommodation function is actually implemented by the redirector, the parameters are passed to the RM, which configures the physical switch, i.e., the RB.

V. NACE AS VNODE FOR INTRA-SLICE SWITCHING

NACE can function as a special type of virtualization node that implements an intra-slice virtual switch by using Ethernet hardware. This function is intended to enable switching of not only Ethernet packets but also packets of arbitrary format with arbitrary type and size of addresses that can be mapped to Ethernet MAC addresses.

A slice may implement the Ethernet protocol or any other protocol suited for switching. If a packet contains the source and destination addresses of any format (with 48-bit (or less) effective address space), it can be switched by the same switching method as Ethernet switches. In addition, if the source and destination addresses can be computed from the packet content, content-based switching can be implemented by using the same hardware. The address converter implemented by using a network processor can be used for a wide range of conversions. However, in the current NACE version, only the Ethernet protocol can be used in slices.

The switching function is modeled as a special type of

ISBN: 978-1-61208-226-4

node sliver called a *switch node-sliver* (**Figure 5**). In contrast to normal node slivers, this node sliver is not programmable, but several switch parameters can potentially be specified. The physical-packet formats used in these slivers in this version of NACE are also shown in this figure. The virtual-packet format always follows the Ethernet protocol.

In Ethernet networks, two types of forwarding functions are available: broadcasting (implemented by repeaters) and switching. We believe that a link sliver should be used for modeling an intra-slice broadcast function because broadcasting is non-intelligent. However, in contrast, a node sliver should be used for modeling an intra-slice switching function because switching is intelligent. Therefore, a node sliver specialized for the intra-slice switching function, namely, a switch node-sliver, is introduced into the VNode architecture.

A switch node-sliver is connected to other node slivers in normal VNodes by link slivers. It is natural to use VLAN-based link slivers to connect an Ethernet-based virtual switch to other virtual node functions; however, the virtualization platform currently only has GRE link slivers, which are thus used for connecting a switch node-sliver to other node slivers. These link slivers, which are specified as normal GRE link slivers, are implemented by using a specialized conversion program in SMCs. Therefore, although these link slivers are specified by using a normal XML definition, the switch node-slivers and these GRE link slivers are handled as a *macro* called a *switch sliver*, which is a combination of node and link slivers and functions as an Ethernet network (Figure 5).

A switch node-sliver can be specified using a switch instance definition in the node-sliver definition. An example of switch node-sliver definition is shown below.

```
<nodeSliver name="InSliceSW">
   <vports><vport name="p1" /><vport name="p2" />
      <vport name="p3" />
   </vports>
   <instance type="switch" /></nodeSliver>
```

It has three ports, p1, p2, and p3, for connecting to link slivers. Because no data conversion is used (currently not available), no other parameters are specified.

A switch node-sliver is virtually managed by the PPM in NACE; that is, the DC sends the node-sliver definition to the PPM through the XML-RPC interface. Because the switching function is actually implemented by the redirector, the parameters in the definition are passed to the RM through a control interface between the RM and the PPM, and the RM configures the physical interface of the RB.

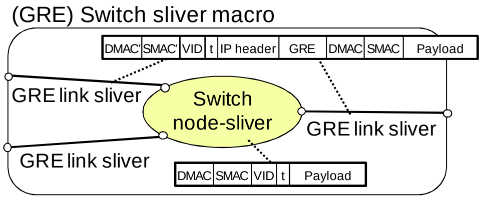


Figure 5. Switch node-sliver, switch sliver macro, and packet formats in the platform

VI. Applications of NACE

This section describes three applications of NACE; namely, gateway for data center, network with intra-slice switches, and slice interaction.

A. Gateway for Data Center

NACE can be used as a gateway for a data center (**Figure 6**). In typical cases, a VLAN used in a data center is connected to a NACE by using one-to-one accommodation. However, two or more VLANs can be accommodated in a slice by using many-to-one accommodation. The NACE can be placed either in the data center via a 10-Gbit Ethernet link or near another VNode of the platform. The NACE can be connected to a virtualization platform through an IP network, and the data center can be accommodated by using a GRE link sliver. The NACE is managed by the DC. Users of the slice can also use the servers in the data center. The maximum bandwidth between the slice and the data center is 10 Gbps. The platform does not manage the addresses of servers and PCs. If IP is used, the addresses are distributed in the usual manner by Address Resolution Protocol (ARP).

Instead of a data center, an external testbed can be connected to the platform and accommodated in a slice by using the same method. Any protocol can be used between a slice and the external testbed, and hardware-based OpenFlow [McK 08] can be used between them. The latter type of deployment is called “OpenFlow In A Slice” (OFIAS) [Du 12][Nak 11].

If protocol converters are inserted at the NACE and the AGW in Figure 6, respectively, a non-IP and/or non-Ethernet protocol can be used in the slice. For example, IP-Ether-Chimera (IPEC) [Kan 12a] can be used here. Each VNode in the platform contains an IPEC software switch as a node sliver. Usual IP/Ethernet is used both in a data center and a user terminal. If both NACE and AGW have a protocol-conversion function from IP/Ethernet to IPEC or vice versa, the user can use servers in the data center through the terminal. However, because NACE and AGW currently do not support this protocol conversion, VNodes connected to them must have the protocol-conversion function instead.

B. Network with Intra-slice Switches

Slices with one or more intra-slice Ethernet switches can be implemented. Two or more switch node-slivers are connected by using link slivers to form a spanning tree. Each slice may contain a different tree structure because a virtual network topology is not restricted by the physical topology. Although the platform packet-format used in

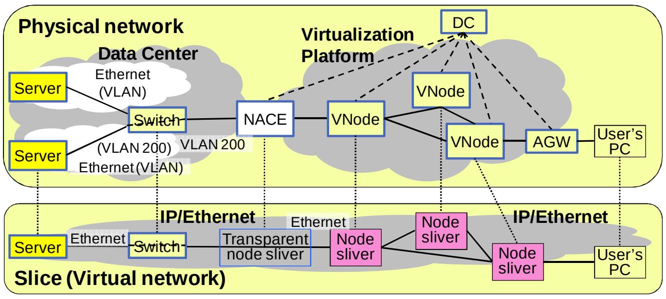


Figure 6. Network with a VLAN-slice gateway

NACE is different from that used in an Ethernet switch, it only implements the Ethernet switching function, so it is not very useful unless it is combined with other node slivers that convert the address format in VNodes. However, a future version of NACE will implement non-Ethernet switching functions customized to each slice.

C. Slice Interaction

A well-controlled interaction between slices, called *slice interaction* or *slice exchange*, may be useful in situations such as an extranet that connect multiple enterprise intranets. Two or more slices can be connected by using NACE. Instead of connecting external networks to the slices, two untagged Ethernet ports can be connected by using a wire for slice interaction. Although the current version of NACE cannot implement filters to control communication between the slices, filters will be implemented by using node slivers in a future version of NACE.

VII. Evaluation

NACE was evaluated by using two slices on testbeds.

A. IPEC-Ethernet Protocol-conversion Gateway

A slice and PC server environments were set up as shown in **Figure 7** as a testbed for measuring the performance of NACE. Unlike Ethernet, IPEC allows redundant paths (loops) in the network [Kan 12a]. The user can use features of IPEC while using Ethernet packets on this network. This slice contains bidirectional IPEC-Ethernet protocol conversion function because, as described above, this version of NACE does not have protocol conversion function. This network consists of three node slivers on three VNodes, and three PCs through two NACEs and a gateway.

Performance between the PC servers and the PC client was measured by using **`iperf`** command. For these measurements, 2-Mbps UDP traffic was used, and the measurement results show that the packet loss rate is less than 0.1%. The performance is better than that in the LAN environment. Round-trip time, 2.8 ms on average, was measured by using a **`ping`** command.

B. VNode with Intra-slice Switch

A slice with a distributed key-value storage application was set up (**Figure 8**). In this slice, each node sliver has a database server. When a new key-value pair is stored in a server, it sends a packet that contains a key to the slice by using an advertise message. The intra-slice switch learns the key. When a client sends a query message that contains the key as the destination MAC address, the server that contains the key-value pair receives the message and sends an answer message that contains the value. A switch can usually learn 10k to 100k keys (addresses). If many more key-value pairs are stored, broadcast storms may occur. This application must also work on a non-virtual Ethernet-based network, but it may cause trouble in a network in real use because it is an unexpected usage of Ethernet. A virtualization platform with NACE is a good environment to test such applications.

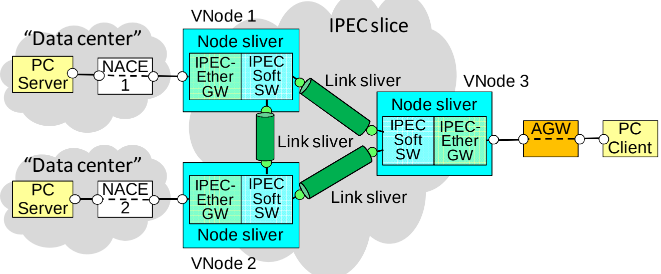


Figure 7. Experimental network with IPEC-Ethernet protocol conversion

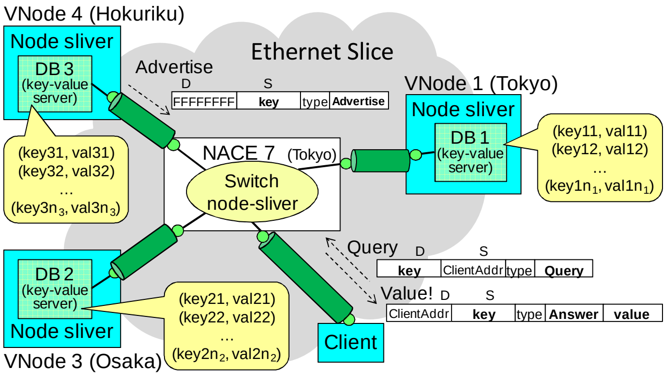


Figure 8. Experimental network for distributed key-value storage using intra-slice switch

In the evaluation, a client that randomly generates queries and three servers (each server for one third of the keys) were used. The virtual switch successfully learned and retrieved 16,384 keys, but it failed to learn 32,768 keys. The servers wait for 8 ms to learn a key, so it takes more than two minutes to send 16,384 advertise messages.

VIII. Conclusion

This paper has introduced a type of physical node, called NACE, which has two roles, i.e., a *network-slice gateway* and a VNode with an *intra-slice virtual switch*, to the VNode architecture. These roles are modeled as a node sliver with a gateway function and a node sliver with a switching function (i.e., a switch node-sliver), and these node slivers are specified by using XML. NACE contains an SMC with a network processor and enables 10-Gbps wire-rate accommodation. These functions were evaluated on two testbeds, and the evaluation results confirm that both functions work correctly and perform well in terms of delay and packet loss.

The most-important focus of future work is to add a high-performance protocol-conversion function to NACE. This function will enable accommodation of external network in a non-IP/non-Ethernet slice and intra-slice switching of packets of any format. Future work also includes automatic configuration of accommodation parameters, such as VLAN ID or physical-port number, and federation of virtualization networks (including those with our virtualization platform), which are connected by NACE. The federation should enable creation and management of slices across domains.

ACKNOWLEDGMENTS

We thank Atsushi Takahara and Noriyuki Takahashi from NTT for discussing the design of NACE and for implementing the management function. We also thank Akihiro Motoki from NEC, Ken'ichi Abiru from Fujitsu Laboratories, Makoto Kitani from ALAXALA Networks, Yasushi Kasugai from Hitachi, and other members of VNP for discussing the design of NACE.

Part of the research results described in this paper is an outcome of the Advanced Network Virtualization Platform (Project A), funded by the National Institute of Information and Communications Technology (NICT), and experiments using a testbed called JGN-X deployed by NICT.

REFERENCES

[AKA 10] AKARI Architecture Design Project, "New Generation Network Architecture — AKARI Conceptual Design (ver2.0)", http://akari-project.nict.go.jp/eng/concept-design/-AKARI_fulltext_e_preliminary_ver2.pdf, May 2010.

[Aoy 09] Aoyama, T., "A New Generation Network: Beyond the Internet and NGN", *IEEE Communications Magazine*, Vol. 47, Vol. 5, pp. 82–87, May 2009.

[Bav 06] Bavier, A., Feamster, N., Huang, M., Peterson, L., and Rexford, J., "In VINI Veritas: Realistic and Controlled Network Experimentation", *2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications* (*SIGCOMM'06*), pp. 3–14, 2006.

[Du 12] Ping Du, Maoke Chen, and Nakao, A., "OFIAS: A Platform for Exploring In-Network Processing", *TridentCom 2011*, *LNICST 90*, pp. 142–151, Springer, 2012.

[Far 00] Farinacci, D., Li, T., Hanks, S., Meyer, D., and Traina, P., "Generic Routing Encapsulation (GRE)", RFC 2784, IETF, March 2000.

[Fel 07] Feldmann, A., "Internet Clean-Slate Design: What and Why?", *ACM SIGCOMM Computer Communication Review*, Vol. 37, No. 3, pp. 59–74, July 2007.

[GEN 09] The GENI Project, "Lifecycle of a GENI Experiment", GENI-SE-SY-TS-UC-LC-01.2, April 2009, http://groups.geni.-net/geni/attachment/wiki/ExperimentLifecycleDocument/-ExperimentLifeCycle-v01.2.pdf?format=raw .

[ITU 12] ITU-T, "Framework of Network Virtualization for Future Networks", Recommendation, Y.3011, January 2012.

[Kan 12a] Kanada, Y. and Nakao, A., "Development of A Scalable Non-IP/Non-Ethernet Protocol With Learning-based Forwarding Method", *World Telecommunication Congress 2012* (*WTC 2012*), March 2012.

[Kan 12b] Kanada, Y., Shiraishi, K., and Nakao, A., "Network-resource Isolation for Virtualization Nodes", *17th IEEE Symposium on Computers and Communications* (*ISCC 2012*), July 2012.

[Kan 12c] Kanada, Y., Shiraishi, K., and Nakao, A., "Network-Virtualization Nodes that Support Mutually Independent Development and Evolution of Components", *13th IEEE International Conference on Communication System* (*ICCS 2012*), October 2012.

[Kou 01] Kounavis, M., Campbell, A., Chou, S., Modoux, F., Vicente, J., and Zhuang, H., "The Genesis Kernel: A Programming System for Spawning Network Architectures", *IEEE J. on Selected Areas in Commun.*, Vol. 19, No. 3, pp. 511–526, 2001.

[McK 08] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J., "OpenFlow: Enabling Innovation in Campus Networks", *ACM SIGCOMM Computer Communication Review*, pp. 69–74, Vol. 38, No. 2, April 2008.

[Nak 10] Nakao, A., "Virtual Node Project — Virtualization Technology for Building New-Generation Networks", *NICT News*, No. 393, pp. 1–6, Jun 2010.

[Nak 11] Nakao, A., Ping Du, and Maoke Chen, "OpenFlow In A Slice (OFIAS)", http://www.corelab.jp/images/poster2.pdf

[Nak 12a] Nakao, A., et al., "Advanced Network Virtualization: Definition, Benefits, Applications, and Technical Challenges, January 2012", NVSG White Paper v.1.0, https://nvlab.nakao-lab.org/nv-study-group-white-paper.v1.0.pdf

[Nak 12b] Nakao, A., "VNode: A Deeply Programmable Network Testbed Through Network Virtualization", *3rd IEICE Technical Committee on Network Virtualization*, March 2012, http://www.ieice.org/~nv/05-nv20120302-nakao.pdf

[Tur 07] Turner, J., Crowley, P., Dehart, J., Freestone, A., Heller, B., Kuhms, F., Kumar, S., Lockwood, J., Lu, J., Wilson, M., Wiseman, C., and Zar, D., "Supercharging PlanetLab — High Performance, Multi-Application, Overlay Network Platform", *ACM SIGCOMM Computer Communication Review*, Vol. 37, No. 4, pp. 85–96, October 2007.

[XML 04] XML-RPC Home Page, http://www.xmlrpc.com/.

ISBN: 978-1-61208-226-4

A First Look at AMI Traffic Patterns and Traffic Surge for Future Large Scale Smart Grid Deployments

Yaling Nie, Yanchen Ma
Hitachi (China) Research & Development Corporation Beijing, China
ylnie@hitachi.cn; ycma@hitachi.cn

***Abstract*—IoT (Internet of Things) applications are deployed in China with strong market requirements. To research the key technologies for future large scale IoT deployment, especially the impacts of IoT traffic to traditional IP network and Data Center, in this paper, we analyzed IoT traffic patterns and verified them through evaluation. IoT traffic patterns with layered system architecture and parameters affecting traffic are discussed. The evaluation results show that traffic surge generated by synchronous IoT application traffic cause network congestion for network and outage of Data Center resources. The transport protocols and data encapsulation format affect the application performance. These potential problems should be further research on.**

Keywords-AMI; large scale; traffic pattern; traffic surge.

I. Introduction

Several types of IoT services are provided to industry and human life, such as smart metering, point of sale, fleet management, telemedicine, environment monitoring and control, home automation, and so on.

Compared to traditional ways of using IP network of human-centric web applications, IoT services are different. The services have less randomness. The sessions are triggered by predefined time or events; the packet series inside a session is also predefined by the program. Usually, the number of IoT devices is very large. Parallel data transmission puts extreme amount of load on individual nodes of networks. The number of IoT devices is changeable. The data might be transmitted synchronously or with a random time schedule.

From the session level point of view, there are 3 cases [1]: The first case is periodical sessions. This is the case in environments monitoring service; the device reports the monitoring data every hundreds of seconds. In the second case, the session initiation times are random. This can be found in a point of sale service; a session is initiated once when people come to trigger a new transaction, not at predefined time. In the third case, the session is usually initiated periodically but may also be triggered by random events. This is the case in telemedicine services, when the physical characteristics data are sent every few minutes and an alarm is sent to the server once a monitored indicator is over or under threshold.

In the near future, massive IoT devices will be distributed everywhere with large scale deployment. Different types of sensors are used for different kinds of applications. IoT service will converge with human centric web applications. It will make a great impact on network and Data Center. New IoT service features generate new IoT traffic features. New IoT traffic patterns are also emerging.

In this paper, we determine three important IoT traffic patterns and analyze their potential problems. The following sections of this paper are organized as follows. In Section II, we analyze the IoT traffic patterns. Section III describes the study of IoT traffic surge, and perspective analysis for use case deployment. Section IV exposes the experimental process and the results. Finally, Section V gives the conclusion.

II. Traffic Patterns Analysis

A. IoT Service Model

Figure 1 shows the typical architecture of IoT system and the corresponding traffic. Data from sensor/meter is gathered through IP network, and transported to servers in data center. The accumulated traffic in data center can reach GB/TB level for computing, storage and networking process.

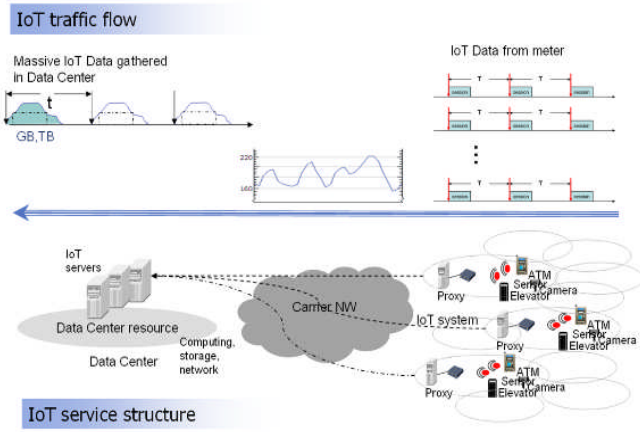


Figure 1. Architecture of IoT system and traffic.

B. *IoT Numerical Model*

We have several assumptions in the dedicated IoT service: sensors' gateways perform data aggregation and processing of sensor data before forwarding it to remote users. Data delivery models, event-driven, query-driven and periodic are assumed to be used by the gateways to transfer data.

We found the IoT traffic gathered in the Data Center server can be formulated as shown in Figure 2. The assumption is on base of IoT application parameters [2][3]. F is the frequency of meter data in sensor/meter. P is the packet sizes of the meter data. F'(t) is the time schedule of concentrator. S is the deployment scale level: like building deployment, site deployment or city level deployment.

$$f_{(F,\, P,\, T,\, S)} = \int^{S} \left(F^{*}\ P \,/\, f'(T) \right)$$

F: frequency of meter data, spans periods of 15s to 3 hours

P: packet sizes, ranging from 50 bytes to few MB

$f'(T)$: time schedule of concentrator transmit

S: scale, Building/site, City

Figure 2. IoT Traffic model.

The IoT traffic has a hybrid traffic patterns. The following three models are selected for further research:

- Traffic pattern 1: Periodical traffic. Frequency: Real-time/Periodical: 15min, 30min, 45min. And Packet length: 50-300B/meter, MB/sensor, Use case: control and automation, transportation, environmental monitoring for emergency services and healthcare.
- Traffic pattern 2: Sessional traffic surge/burst, event-driven, query-driven, Frequency: 1h~1d~1M/Packet length: KB~MB, Use case: emergency report, booting
- Traffic pattern 3: Periodical traffic surge. In large scale deployment, the traffic load rises to the system resource capacity (threshold), or even over the system resource capacity (threshold). The hybrid traffic of Traffic pattern 1 and Traffic pattern 2 can generate periodical traffic surge, Use case: interactive, conversational, streaming.

C. *IoT Traffic patterns with Layered Structure*

Hybrid IoT traffic gathered in the Data Center, with the key parameters of: F, P, T, S. The total system has a layered structure. Layer 1 is the sensor/meter network layer, where real-time traffic is generated. Most of the traffic from layer 1 will go through GW/concentrator layer for a further time schedule. Some IoT traffic will go directly to layer 3. Layer 3 is the IP network, connecting large scale geography area. The hybrid IoT traffic finally reaches servers in Data Center: layer 4.

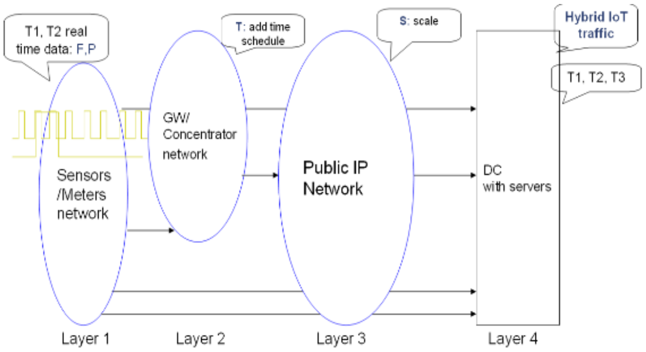


Figure 3. Layered structure.

Data collection frequency F and system scale S are parameters affecting total data volume. In a critical scenario, if the data collection frequency F is high, and the system scale S increases to a large value, the total data volume will be large.

The collection period (T1) and the collection time (T2) are parameters affecting the traffic shape. As shown in Figure 4 below, the predefined IoT traffic is periodically reporting data with T1 slot. The real transmission of the data in T1 slot is in T2 period. With different ratio of T1 and T2, the IoT traffic will be deferent.

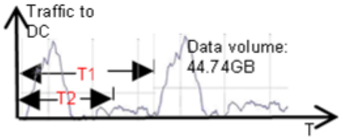


Figure 4. Collection period T1 and collection time T2.

III. Traffic Surges Analysis

Periodical load surge from IoT application or special events (e.g., emergency alerting) will conflict with Data Center resource capacity, even inducing break down of Data Center systems.

A. *Use Case Study*

Mobile data traffic surge is expected to be 40 Exabyte by 2014 [4]. Mobile connections are expanding globally, along with other mobile connections, due to the growing hardware and software components for smart meters, business and consumer surveillance, inventory management, and fleet management, all of which are designed for operational excellence. Machine-to-Machine Traffic is expected to increase 40-Fold between 2010 and 2015.

Mobile carriers such as NTT DoCoMo, Verizon met mobile traffic burst, conflict with system resource. ISP like TAOBAO met traffic surge also. Its SecKill service, which is a kind of time-limited sales promotions, one web has requirements of 1billion in 10mins. The first system break down happened in 2009.

Chinese carriers/vendors are considering traffic/flow control. In carrier network, the real time large traffic is out of control. A possible solution might be an intelligent pipe:

broadband rapid respond for bandwidth requirements, resource allocation, optimization, and so on.

B. *Beijng City Perspective AMI deployments*

City level deployment of Advanced Metering Infrastructure (AMI) services is very popular in China. Table 1 shows the deployments in three cities.

TABLE I. AMI DEPLOYMENT BY 2011

City	L2 devices (GW/Concentrator)	User Number
TaCheng, XINJIANG	36000smart meter, 185 collector	19,200
LuCheng, Wenzhou, ZHEJIANG	14,485 cellular collector	190,405
HuZhou, ZHEJIANG	487.1 thousand cellular/microwave collector	HV 12690， LV 1187,700

Refer to the AMI deployment of HUZHOU in 2011 [5], with the population result of 2010; we can get the forecast AMI numbers of BeiJing for city level deployment shown in Table 2.

TABLE II. PREDICTIVE NUMBER FOR BEIJING DEPLOYMENT

	BeiJing	HuZhou
Population	19.612 million	19,200
People/House density	2.45	2.65
House Users	8,000,000 users	1200,000 users
L2 devices	3,600,000 collector	487,100 collector

Table 3 is the comparison of China railway public ticket ordering system 'www.12306.cn' [6] and AMI service. The ticketing system met heavy traffic surges especially at spring festival ticket release time. At 8, 10, 12, 15 everyday the traffic surges happened. As the deployment reaches a large scale, and the interactive requirements increase, AMI applications have a high possibility to have traffic surges periodically as well.

TABLE III. TRAFFIC SURGE POSSIBILITY

	12306	AMI
Parallel traffic	1GB	< 1GB
Server load	1GB	MB level
User number (peak)	5 million, KB/user	5 million, KB/user
mode	interactive	Light interactive

The common solution is to increase the system capacity. But, there are some solutions considering IoT traffic surge [7][8][9][10]. Like [11], in M2M communication: terminal self-test and determine communication gap to avoid traffic congestion. Or, some vendor has a hardware solution of DC network IF with huge size buffer.

IV. EVALUATION RESULTS

We use AMI as the example for evaluation of IoT traffic patterns, its impacts and problems. The AMI system will be deployed in large scale [12]. Thus, the application will generate massive AMI data [13]. This massive AMI data will be transported through IP network and processed in the data center.

A. *Platform Implementation*

We select AMI Beijing city level deployment as the application scenario as shown in Figure 5. Sensors' data is collected every 15min 60Byte/meter, periodical data delivery (transmission finished within) in network. Traffic Generator send IoT traffic: historical AMI data repeating, predicted AMI data, AMI data through mathematics model. Multiple Traffic Generators simulated massive IoT traffic to IoT server in Data Center virtualization platform. Evaluation parameters: traffic model to Data Center, capacity, jitter, traffic/time model for different AMI scenarios.

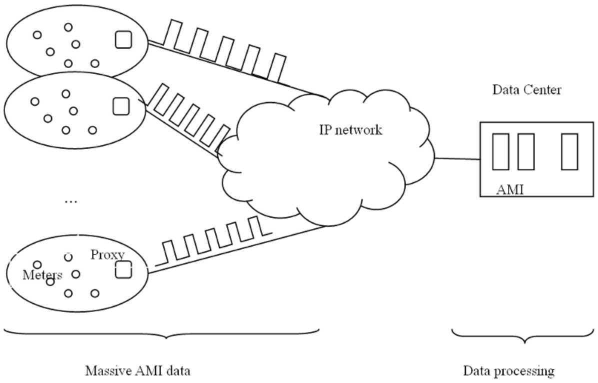


Figure 5. Logical System Structure.

In Figure 6, the Traffic Generator (TG) is used to generate massive AMI data. Multiple TGs are used to simulate distributed AMI systems. Data from TG will be transmitted to IP network, through local or public network. The AMI server and database are in the data center, with virtualization platform.

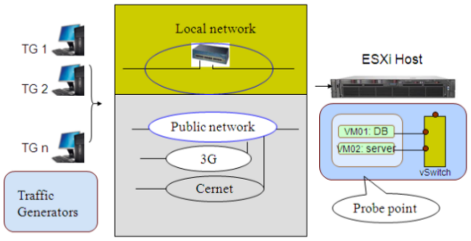


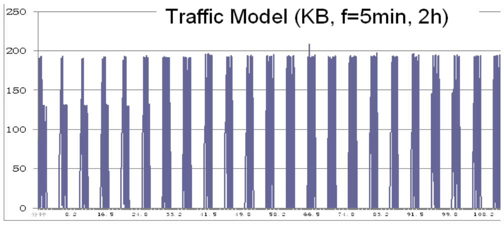
Figure 6. Evaluation System structure.

B. *Problem Analysis*

a) Traffic Paterenss via Evaluation

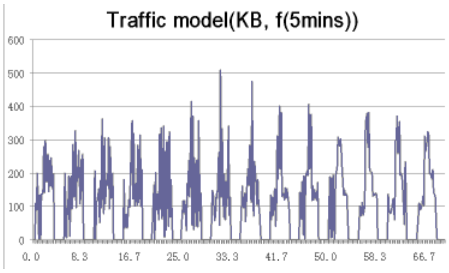
Traffic patterns are monitored through evaluation.

Data from sensors, through the public IP network, is gathered in the AMI server. The traffic is shown in Figure 7 below. It is periodical traffic. In Figure 7(a), it is around 6000 sensor's metadata, through 24 concentrators, sending data to servers in data center. The data frequency is set to 5 minutes, for it is easier to get the test result than 15 minutes used in the commercial system.



.

(a)



(b)

Figure 7. Layered structure.

In Figure 7(b), we increased the number of the sensors to 18000, through 72 concentrators. The total traffic arriving at the server in data center is shown in Figure 7(b). Compared to the result in Figure 7(a), the traffic in one period is affected with jitters. However, the periodical feature is still not changed. And the traffic in every period has similar traffic model features from statistics points.

b) Traffic Impacts to Data Center

Corresponding to the traffic patterns in Figure 7(a) and 7(b), the CPU and memory utilization in data center servers is pushed to have the same periodical load model feature.

Potential problems:

(1) Low efficiency: in the time slots between the transmission periods, IoT dedicated resources like computing, storage and network are in low efficiency status.

(2) Resource Bottleneck: For large scale massive data generated traffic surge, during the session, the allocated resource will be bottleneck to the load.

c) Traffic Impacts to network

Network resource occupation competition:

In the concentrator, there are long packets series transmitted in a short period; it occupied the network I/O buffer. In the server, packets from concentrators have competitions for the network resources. Especially in wireless network, the wireless buffer and channel resource are limited. Figure 8 shows a delay burst in the wireless channel while the wireless channel is rather congested.

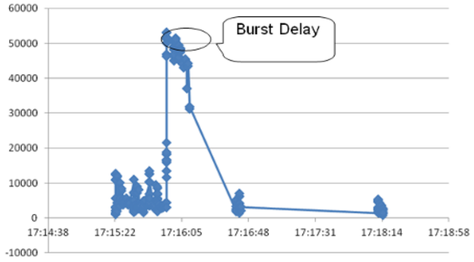


Figure 8. Delay in wireless channel competetion.

Potential problem:

(1) Long packets series, with synchronized feature for real-time applications, will generate network congestion. Then the application quality will be affected.

d) Traffic Impacts to applications QoS

The traffic model has an impact on application QoS parameters.

As Figure 9 shows, the TG sends 6000 sensor's data. These sensor's data are encapsulated in XML format and transported over TCP/IP. In one sending period, one stream, around 4KB, are divided into packet series to the TCP receiver in the server, and the server will buffer these packets, until the last packet comes. Then these 4KB data will decapsulated together. So the delay of the data in the first packet is increased with the longest buffering time: the delay of the data in the last packet is the shortest one. If the number of the packets increases together with sensor number, this factor will affect the application QoS more deeply.

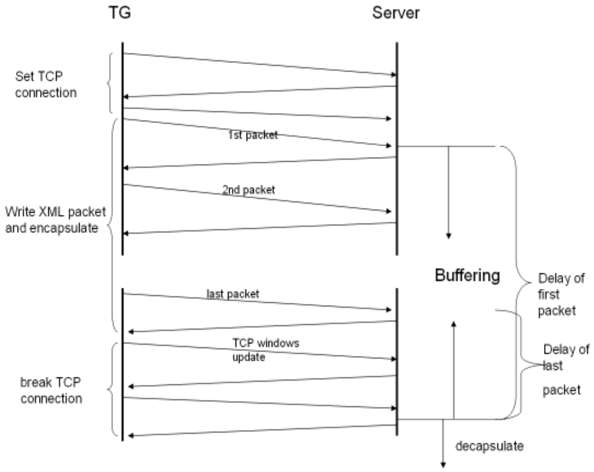
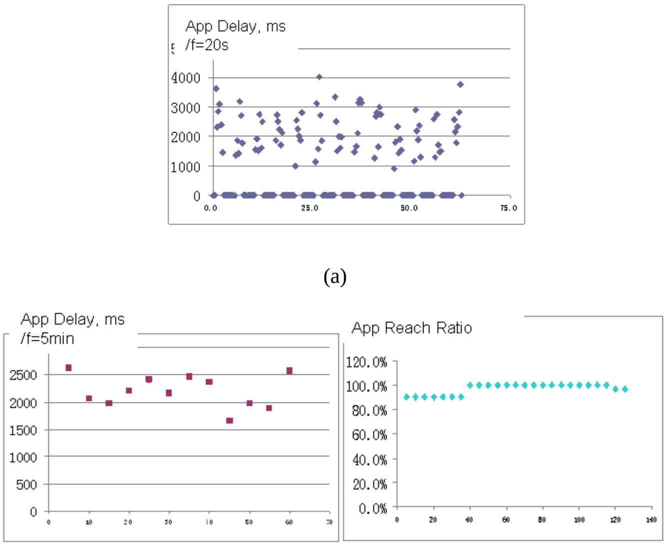


Figure 9. TG + Server schedule.

Figure 10 is result of the application QoS. They are sensor meta data delay average in 20 seconds (a), sensor meta data delay in 5 minutes (b), and sensor meta data reach ratio(c). In

Copyright (c) IARIA, 2012. ISBN: 978-1-61208-226-4

Figure 10, 6000 sensor's data cross fixed public IP network, has average delay in seconds level.



(a)

•App Delay: One AMI record delay, meter send-> server store to Database

•App Reach Ratio: AMI record reach ratio, meter send -> server store to Database

(b) (c)

Figure 10. Application QoS Parameters.

Potential problems:

(1) How to decrease the delay in the application layer and assure the QoS of IoT services for this traffic model?

(2) Transportation protocols and data encapsulation format (packet length, format) are important parameters of the traffic model. For TCP transportation, with long packets series, uniform packets encapsulation, the application QoS will be decreased.

In the test bed, TCP was the protocol used for assuring packet transmission. However, depending on the network status, TCP windows update mechanism will result in delay and even packet loss. On the other hand, if we use UDP protocol, the delay of above problem can be solved to some extent. However, in channel or network congestion, there will definitely be packet loss of the sensor data.

We use XML format for the meta data in the test bed. This format is easier for encapsulation and decapsulation. However, the efficiency for transmission and server processing is not enough, especially for massive data from sensors. There also should be an efficient and standard meta data format.

V. CONCLUSION

Large scale IoT deployments generate a new class of traffic. It is important to know the large scale IoT deployments impact on Data Center and the network. Potential problems should be further studied. These include: traffic surge generated by synchronous IoT application, in which traffic may cause network congestion and outage of DC resources. In this report, we discussed the IoT traffic modeling and evaluation in order to clarify the impact of future large scale IoT server on network and data center. Through the IoT traffic modeling analysis and the evaluation work, we found the potential impacts and problems of IoT massive data to datacenter, network and application QoS.

Traffic surge generated by synchronous IoT application traffic may cause network congestion for cellular network and outage of DC resources. The transport protocols and data encapsulation format will affect the application processing performance greatly and should be carefully selected. It is important to find solutions to these issues.

REFERENCES

[1] Huawei, "Traffic model for M2M services," 3GPP TSG-RAN WG2 Meeting #69 R2-101184, 2010

[2] Jasmina Krnic and Srdjan Krco, "Impact of WSN Applications' Generated Traffic on WCDMAAccess Networks," 19th International Symposium on Personal Indoor and Mobile Radio Communications (2008) IEEE

[3] Rongduo Liu, Wei Wu, Hao Zhu, and Dacheng Yang, "M2M-Oriented QoS Categorization in Cellular Network," 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 2011 IEEE

[4] Cisco White Paper, "Cisco Visual Networking Index: Global Mobile DataTraffic Forecast Update," 2010–2015, VNI Mobile 2011

[5] http://www.huzhou.gov.cn/art/2011/12/19/art_24_84435.html , accessed August 2012

[6] www.12306.cn , accessed August 2012

[7] CHEN Yun-sheng, FU Tun, "Application of wireless broadband access technology in power distribution and utilization network," Telecommunications for Electric Power System, Vol 31 No. 212, Jun. 10 , 2010, pp. 10-13

[8] Jae Yoo Lee and Soo Dong Kim, "Software Approaches to Assuring High Scalability in Cloud Computing," 7th IEEE International Conference on E-Business Engineering, 2010, pp. 300-306

[9] Jasmina Krnic and Srdjan Krco, "Impact of WSN Applications' Generated Traffic on WCDMA Access Networks," PIMRC 2008: 1-5

[10] M.T.S. Jonckheere, R. N´u˜nez-Queija, and B.J. Prabhu, "Performance Analysis of Traffic Surges in Multi-class Communication Networks," Proceedings of International Teletraffic Congress 2010

[11] News:http://itpro.nikkeibp.co.jp/article/NEWS/20110527/360767/ , accessed August 2012

[12] M. Zubair Shafiq, Lusheng Ji, Alex X., Liu Jeffrey, and Pang Jia Wang, "A First Look at Cellular Machine-to-Machine Traffic –Large Scale Measurement and Characterization," Joint ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS) and IFIP International Symposium on Computer Performance, Modeling, Measurements and Evaluation (Performance), London, UK, June, 2012.

[13] Sandra C´espedes, Alvaro A. C´ardenas, and Tadashige Iwao, "Comparison of Data Forwarding Mechanisms for AMI Networks," Innovative Smart Grid Technologies (ISGT), 2012 IEEE PES, Publication Year: 2012 , Page(s): 1 - 8

ISBN: 978-1-61208-226-4

A Platform for the Integrated Management of IT Infrastructure Metrics

Christian Straube, Wolfgang Hommel, Dieter Kranzlmüller
Leibniz Supercomputing Centre, MNM-Team, Boltzmannstrasse 1, 85748 Garching
[*straube,hommel,kranzlmueller*]*@lrz.de*

***Abstract*—In this work-in-progress paper, we argue that most measurement and metrics as they are used in today's IT management do not provide a sufficient foundation for qualified upper level management decisions. By exemplary applying the state-of-the-art energy efficient metrics to SuperMUC, an energy-efficient three PetaFlop/s high performance computing system that has been put into service in June 2012 at the Leibniz Supercomputing Centre, we show that there are four major gaps between the information that can be measured on a technical level and the information that is needed for management decision making. We then present our vision of a *management cockpit* that centralizes measurement and metrics management in an organization-wide manner. It aggregates, processes and transforms metrics data into indicators for management decisions. We present the research questions, our solution approaches, and preliminary results regarding the design and implementation of this management cockpit.**

***Keywords*-IT management; measurement; metrics; governance; decision support.**

I. Introduction

For many cloud computing, high performance computing (HPC), and other large data centers, raising energy consumption costs are one of the prime motives for an in-depth examination of energy-efficient technology. Obviously, energy efficiency must be considered before investing into new hardware because, for example, CPU frequency scaling is an important functionality that helps to adjust energy consumption proportional to the current workload. However, energy efficiency is not a static property that is only relevant during purchasing decisions. On the one hand, energy-saving capabilities must constantly be monitored and controlled to ensure that they are working as expected and to keep their parameters optimized for the current workload. On the other hand, the energy efficiency of, for example, air-conditioning or hot liquid cooling systems depends on ever-changing environmental characteristics such as the current outdoor temperature.

We argue that in order to better support management decisions – such as how much money to invest into specific Information Technology (IT) infrastructure improvements – a more holistic approach to measurements and metrics is urgently required. As we show in Section II, many energy efficiency metrics have been created in the past few years. However, using SuperMUC – our three PetaFlop/s HPC system [1] that entered the top 10 of the Top 500 Supercomputer Sites list in June 2012 – as an example, without loss of generality, we demonstrate in Section III that there are four major gaps that need to be closed before metrics can directly contribute to a holistic view of IT infrastructures: At the moment, 1) the information provided by metrics is not sufficient for decisions on higher abstraction levels, 2) dependencies between metrics are not sufficiently considered, 3) organization-specific requirements cannot be incorporated adequately, and 4) there is a lack of improvement recommendations that can be deduced from the metrics values.

Our work-in-progress envisions a *management cockpit*, which centralizes measurement and metrics management organization-wide. We present its design in Section IV, using energy efficiency metrics for SuperMUC as an example of how low-level metrics can be aggregated to lay the foundation for upper-level management decisions. While works for simple metrics compositions with an immediate practical benefit already, we discuss several research questions that need to be addressed in Section V. We then outline our approach, present our preliminary results, and give an outlook to our next steps.

II. Metrics management in related work

The related work to our management cockpit vision can be grouped into the three following categories.

Metric definition – There are a lot of metrics dealing with different aspects of energy-efficiency, for instance the measurement of the energy consumption of computing servers and clusters [2], [3], or the energy consumption in optical IP networks [4]. Further examples are TEEER [5], EPI [6], ECR, and ECRW [7], [8], [9]. All of them are defined by providing calculation and interpretation rules, partially in a very comprehensive way, but nevertheless they all focus on technical aspects of a single entity on a very low level. Hence, they do not facilitate a holistic view on the energy efficiency situation of SuperMUC, which, being a large HPC system, aggregates many different hardware components in a complex architecture.

Structuring and comparison – There is literature and ongoing research in several topics about metrics taxonomy [10], [11], classification [12], and comparison [13]. These approaches structure and compare the aforementioned metrics, but they neither aggregate different metrics to derive new statements, nor do they consider dependencies and correlations between metrics. Instead, they focus on a single specific class of metrics, like equipment-level metrics, and

ISBN: 978-1-61208-226-4

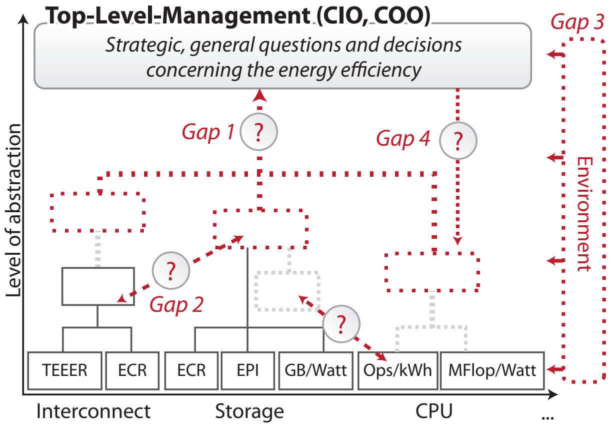


Figure 1. The four gaps in today's situation that hamper a holistic view.

confine themselves to comparison. Therefore, they do not allow a holistic view either.

Analyzing combined metrics – Besides the sole definition of metrics (group one) and the classification of those metrics (group two) there is a third group of related work that deals with the task of combining and aggregating [14] several metrics to deduce new statements about the energy efficiency of an IT infrastructure as a whole. There is no work yet that focuses on supporting upper-level management decisions. In the following, we propose our vision of a management cockpit to address this issue.

III. Problem Statement

The problem description is given with the help of the following exemplary management question: "Which components should be invested into during the next SuperMUC upgrade in order to save as much money by energy cost reduction as possible during the next *n* years?". By trying to answer this question, and given the previous outlined contributions of related work, we have identified four gaps, which are depicted in Figure 1. In the following explanations of the gaps, we first give a short example concerning SuperMUC and then a generalization of the problem, respectively.

Gap 1 – The Information Gap In order to answer the aforementioned management question, we first have to decide which components have the poorest energy efficiency in SuperMUC, as their potential for further improvement during the next system extension is the highest. Beside a few generally applicable metrics, most metrics can be applied only in one area, for instance an HPC/CPU metric like *MFlops/Watt* cannot be applied to storage, interconnect, or software components. Hence, there are several different metrics that have to be considered for SuperMUC as a complete system.

Generalization: For a holistic view, the (purely) technical information has only limited expressiveness and must be enriched by context and comparison information; additionally, all those information have to be aggregated to provide comprehensive information to support decision making at high level. Therefore, conversions, e. g., into currencies or hours of work, may be required.

Gap 2 – The Dependency Gap In the SuperMUC scenario, changing the CPU type to achieve a higher energy efficiency would have (strong) side effects on other components of SuperMUC. For instance, using CPUs with a smaller L2 cache size might improve the CPU energy efficiency, but at the same time, SuperMUC's system interconnect between the CPUs and the non node-local memory will have higher workloads and therefore, its energy efficiency is decreased. This may lead to a decreased overall energy efficiency.

Generalization: There are a lot of dependencies between metrics that have to be considered in order to include all relevant information. In most cases it is not adequate to improve one or only a few metrics, i. e., partial optimizations do not yield optimum results. Instead, all (involved) metrics should be improved [16], correlations have to be shown, and conclusions have to be drawn from these correlations [17].

Gap 3 – The Environment Gap Changes regarding the CPUs obviously do not only affect the storage, but also the cooling facilities. In order to assess the changed amount of energy, which the cooling facilities need for the additional CPUs, we have to compare numbers to other supercomputing centers. But we can get only useful statements if we consider the specific environment, including the location SuperMUC is deployed in: Bavaria in Germany is a relatively warm region, whereby Iceland is quite cold, so the demand for cooling would be lower there.

Generalization: The same measurements and metrics may not have the same expressiveness and purpose in different scenarios. Each organization will have to make specific adoptions to existing metrics, create complementary metrics for its specific environment, and specify, for example, how results must be interpreted properly.

Gap 4 – The Activity Gap In the end it may turn out that despite all reciprocity, using different CPUs than previously is the best way to optimize energy costs. Now we need activity recommendations that describe what to do while considering implications to other metrics and the caused costs. For instance, changing cache size has a medium energy saving effect and is quite cheap, but has a high effect on the interconnect components, whereby changing the clock speed has a high energy saving effect, is quite cheap as well, and has a low effect on the interconnect and storage components.

Generalization: In order to perform the best adoptions on the analyzed IT infrastructure, activity recommendations have to be generated out of the holistic view in a (semi-) automated way. There are a lot of challenges, such as the calculation of costs associated with each activity recommendation, and selecting the best one for a given scenario based on a consideration of the mutual reactions of metrics when changing the infrastructure, e. g., based on the application

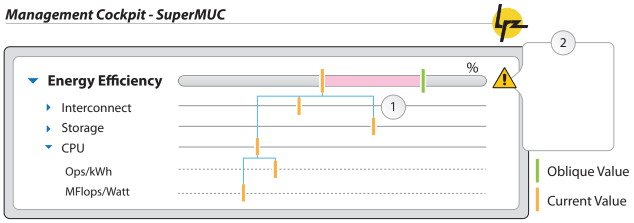


Figure 2. Our vision of a management cockpit that provides high-level, management-relevant information.

of mathematical optimization algorithms.

Closing these gaps is already a non-trivial task for the area of energy efficiency, which we use as an example here. The key challenge of our research is, however, to find a management solution that works with an arbitrary number of metrics categories and their interdependence in parallel, as we outline in the next section.

IV. Our vision of a management cockpit

To close the described four gaps, we envision a management cockpit that is built on top of an underlying comprehensive and integrated processing layer, which manages all applied metrics, and a presentation layer; a mock-up of its GUI is depicted in Figure 2. It provides an holistic view concerning the energy efficiency of SuperMUC to the top-level-management and makes all aspects feasible that were described in Section III.

Besides its main objective to "monitor, visualize and explore measurement data from different perspectives and at various levels of detail" [17] in order to characterize, evaluate, predict, and improve IT infrastructures, there are some underlying sub-objectives that shape the management cockpit, which we discuss next.

A. Considering metric dependencies

As described in Section III, metrics are currently used isolated of each other, and hence reciprocity cannot be respected. Our management cockpit shall manage all metrics that are used by an organization in a holistic way and thus make integrated statements about the SuperMUC example feasible. Besides the big advantage of respecting the reciprocity of metrics, considering metric dependencies facilitates the uncovering of strategic goal conflicts [18] and the consideration of trade-offs, for instance between energy efficiency and performance. Those trade-offs are already analyzed in some lower-level-approaches (e. g., [12]), but not yet at upper-level management.

B. Integrating measurement and metrics data

To support the holistic view and root cause analysis facilities, our solution shall integrate ("selection, embedding, and handling of the underlying data sources" [19]) and use as many data sources as required and reasonable. This leads to the problem of using and consolidating several data structures and to identify valid data contexts [17]. To be able to use the management cockpit from the first day and to avoid the "cold-start–problem" [20], existing and actual data, metrics, and measurements have to be embedded [17].

C. Using data trees

To achieve provenance for the statements provided to the management decisions, every statement must provide its data source tree to facilitate root cause analysis: starting at the top level, any aggregated metrics value can be broken down into smaller pieces and it can be explained how this high-level current value materializes. Figure 2 depicts an exemplary data tree for SuperMUC: the overall energy efficiency value is aggregated by interconnect-, storage-, and CPU-specific values, whereby the CPU value is composed of Operations/kWh and MFlops/Watt measurements.

D. Warnings and activity recommendations

As described in the aforementioned objectives, our solution shall support decision making and action planning. Therefore, there are activity recommendations that are proposed by the management cockpit, marked by (2) in Figure 2. Those recommendations depend on the delta of oblique and current values. One of the research questions we have to investigate is the modeling and creation of those recommendations; we describe this in the next section.

V. Research questions and preliminary results

To achieve our vision of the management cockpit, we have identified the following research questions, which we will analyze and answer in our future work.

A. Adequate nomenclature and classification

There are a lot of different terms that are used in the context of assessing, characterizing, and valuating the energy efficiency as well as other characteristics of an IT infrastructure, for instance *metric*, *measurement*, *quality*, *benchmark*, or *key performance indicator (KPI)*. Even if there is some literature about defining those terms (e. g., [21], [13], [22], [23], [24]) we have to define them by ourselves. Otherwise there is the risk to compare and aggregate values with different meanings and intentions, for instance a metric value and a benchmark value. To achieve this goal, we look at a metric as mathematical function and hence, it has four main components: function domain, function image, dependencies, and meta information. With this perspective, we concentrate on the characteristics of the image or range of those functions, for instance the scale, while sorting existing terms and defining our own terminology.

B. Considering metric dependencies

The area of metric dependencies is twofold: detection of dependencies, and modeling of dependencies. Both areas have individual characteristics and challenges, which have to be analyzed. The *detection of dependencies* can be done analytically or empirically. An analytical detection would look at existing information about dependencies, for instance a Configuration Management Database (CMDB) [25], and derive dependencies from those sources. An empirical detection would collect all available data at different points in time and compare them, for instance before and after a reconfiguration. After we have detected the dependencies, we have to store them in an appropriate way, so a *data model* is necessary, which must be capable of all the objectives that were introduced in the last section. Beside the aforementioned mathematical perspective, our data model divides dependencies into *reciprocity*–dependencies and *aggregation*–dependencies: the first one models correlations between metrics – for instance, improving CPU energy efficiency potentially decreases interconnect energy efficiency. The second one models the aggregation of metrics to form new statements (cf. Section IV). Interesting questions in the context of the data model are, whether there are fields that all metrics have in common, and if those fields could be placed into an abstract metrics class. This would lead to a very efficient and handy data model.

C. Derivation and generation of new statements

In the next step, we have to shape the holistic view out of the low-level approaches and thereby close gap 1. Possible solution paths are bottom-up, hypothesis generation on middle, and top-down. *Bottom-up* means that we use existing data from low abstraction levels and try to aggregate them iteratively until we reach the values that are displayed in the management cockpit. The most difficult task while doing a bottom-up generation is the “correct” selection of attributes/values at the lowest level. *Hypothesis generation* means that we formulate hypotheses on an intermediate level and try to prove or disprove those hypotheses by applying data from low abstraction levels. Those (dis)proved hypotheses are afterwards used to generate statements for the high-level management-cockpit. *Top-down* means that we start at certain points in the management cockpit and try to create the data tree beginning at the root by recursively finding suitable metrics on the next lower level. Our assumption is that we have to analyze each of those three possible ways and use a hybrid solution.

D. Target values and comparison

In order to provide “Warnings and activity recommendations” (cf. Section IV-D), target values and interpretation rules for a delta between those target values and current values are mandatory. We have to investigate how to define or rather find those target values. This step is very critical, because having wrong target values would lead to optimizing the infrastructure towards wrong values. Additionally, we have to analyze how to interpret a delta between the current value and the target value for any given metric. This interpretation has three dimensions: overall meaning, timing aspects (e. g., “delta implies the necessity to act immediately”, “delta is just for the annual, paper-based report”), and impact (e. g., “the severity of the delta is very high”, “solving the delta is very costly”).

E. Modelling environment characteristics and their connections to metrics

The environment of an IT infrastructure can be very challenging and influences the operation and assessment of an IT infrastructure heavily. The strong impact is shown by various empirical studies, for instance [26]. To respect this fact, we want to model the environment characteristics and their connections to metrics, respectively, in order to consider this impact in the statements the management cockpits produces. The questions that arise in this context are the selection of environment characteristics that shall be modeled, the design of the data scheme to store those characteristics, and on which points those characteristics shall be connected to metrics.

VI. Conclusion and Future Work

We have shown that the energy efficiency metrics that are in use today serve their purpose of benchmarking technical components quite well, but their individual expressiveness is insufficient for IT management decisions on higher abstraction levels. Using the SuperMUC HPC system as an example, we demonstrated that the information gap, the dependency gap, the environment gap, and the activity gap need to be overcome in order to gain a holistic view on the energy efficiency of a complex IT infrastructure.

We then presented the core component of our approach, the *management cockpit*, which integrates all technical aspects of organization-wide measurement and metrics management. By aggregating, processing, and transforming existing metrics, which then are visualized for different target audiences, it is intended to be a central management tool for monitoring and decision making support. Fully overcoming the four identified gaps requires addressing several research questions first. We outlined them along with our solution approach and our preliminary results.

Our next steps include the specification of a generic metrics data model that can be applied to existing metrics and also captures their interdependence. It will serve as a basis for a prototype implementation to demonstrate the benefits of our approach in the SuperMUC real-world example. We are also working on a generalization of our approach so that besides energy efficiency, also performance, quality-of-service, and security metrics can be managed and combined in an integrated manner.

Acknowledgment

The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments on previous versions of this paper. [27].

References

[1] Ludger Palm, "LRZ awarded German Data Centre Prize," *Inside – Innovatives Supercomuting in Deutschland*, vol. 10, no. 1, 2012.

[2] Christos Kozyrakis and Parthasarathy Ranganathan and Suzanne Rivoire and Mehul A. Shah, "JouleSort: a Balanced Energy-Efficiency Benchmark," in *Proceedings of the International ACM Conference on Management of Data (SIGMOD '07)*, 2007.

[3] Christos Kozyrakis and Justin Meza and Parthasarathy Ranganathan and Suzanne Rivoire and Mehul A. Shah, "Models and Metrics to Enable Energy-Efficiency Optimizations," *Computer*, vol. 40, no. 12, pp. 39–48, 2007.

[4] Robert Ayre and Jayant Baliga and Kerry Hinton and Wayne V. Sorin and Rodney S. Tucker, "Energy Consumption in Optical IP Networks," *Lightwave Technology*, vol. 27, no. 13, pp. 2391–2403, 2009.

[5] L. C. Graff and T. Talbot, "Verizon NEBS TM Compliance: TEEER Metric Quantification," Tech. Rep., 2009.

[6] Sujata Banerjee and Priya Mahadevan and Parthasarathy Ranganathan and Puneet Sharma, "A Power Benchmarking Framework for Network Devices," in *Proceedings of the 8th International IFIP-TC 6 Networking Conference (NETWORKING '09)*, 2009.

[7] Luc Ceuppens and Daniel Kharitonov and Alan Sardella, "Power Saving Strategies and Technologies in Network Equipment Opportunities and Challenges, Risk and Rewards," in *Proceedings of the International IEEE Symposium on Applications and the Internet (SAINT 2008)*, 2008.

[8] A. Alimian and B. Nordman, "Network and Telecom Equipment - Energy and Performance Assessment – Test Procedure and Measurement Methodology," Tech. Rep., 2008.

[9] "Energy Efficiency for Network Equipment: Two Steps beyond Greenwashing," Juniper Networks, Inc., Tech. Rep., 2010.

[10] Ronda Henning and Ambareen Siraj and Rayford B. Vaughn, Jr., "Information Assurance Measures and Metrics - State of Practice and Proposed Taxonomy," in *Proceedings of the 36th International IEEE Hawaii Conference on System Sciences (HICSS03)*, 2003.

[11] Samee U. Khan and Lizhe Wang, "Review of Performance Metrics for Green Data Centers: a Taxonomy Study," *The Journal of Supercomputing*, 2011.

[12] Aruna Prem Bianzino and Anand Kishore Raju and Dario Rossi, "Apples-to-Apples: a Framework Analysis for Energy-Efficiency in Networks," *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 3, pp. 81–85, 2010.

[13] S. C. Payne, "A Guide to Security Metrics," SANS Institute, Tech. Rep., 2006.

[14] Viktoria Firus and Ralf Reussner, "Basic and Dependent Metrics," in *Proceedings of the Dependability Metrics: Advanced Lectures [result from a Dagstuhl seminar]*, 2008.

[15] Victor R. Basili and David M. Weiss, "A Methodology for Collecting Valid Software Engineering Data," in *Proceedings of the 6th IEEE Transactions on Software Engineering*, 1984.

[16] Eduardo Fernndez-Medina and Mario Piattini and Carlos Villarrubia, "Metrics of Password Management Policy," in *Proceedings of the International Conference on Computational Science and Its Applications (ICCSA 2006), Part III*, 2006.

[17] Rudolf Ramler and Klaus Wolfmaier, "Issues and Effort in Integrating Data from Heterogeneous Software Repositories and Corporate Databases," in *Proceedings of the 2nd International ACM/IEEE Symposium on Empirical Software Engineering and Measurement (ESEM '08)*, 2008.

[18] Daniel Mares and Alessia C. Neuroni and Reinhard Riedl and Christoph Schaller and Sauter Urs, "Cockpits for Swiss Municipalities: a Web Based Instrument for Leadership," in *Proceedings of the 11th International ACM Digital Government Research Conference on Public Administration Online: Challenges and Opportunities (dg.o '10)*, 2010.

[19] Klaus R. Dittrich and Patrick Ziegler, "Three Decades of Data Integration - All Problems Solved?" in *Proceedings of the 18th World Computer Congress on Building the Information Society (IFIP)*, 2004.

[20] David M Pennock and Alexandrin Popescul and Andrew I. Schein and Lyle H. Ungar, "Methods and Metrics for Cold-Start Recommendations," in *Proceedings of the 25th International ACM Conference on Research and Development in Information Retrieval (SIGIR '02)*, 2002.

[21] Horst Lichter and Holger Schackmann, "Process Assessment by Evaluating Configuration and Change Request Management Systems," in *Proceedings of the Warm Up Workshop for ACM/IEEE ICSE 2010 (WUP '09)*, 2009.

[22] Reijo Savola, "Towards a Security Metrics Taxonomy for the Information and Communication Technology Industry," in *Proceedings of the International IEEE Conference on Software Engineering Advances (ICSEA 2007)*, 2007.

[23] John I. Alger, "On Assurance, Measures, and Metrics: Definitions and Approaches," in *Proceedings of the Workshop on Information-Security-System Rating and Ranking (WISSSR)*, 2002.

[24] "System Security Engineering - Capability Maturity Model – Model Description Document (Version 3.0)," Tech. Rep., 2003.

[25] S. Knittl, "Werkzeugunterstützung für interorganisationales IT-Service-Management - ein Referenzmodell für die Erstellung einer ioCMDB," Ph.D. dissertation, Technische Universität München (TUM).

[26] Bin Gu and Gautam Ray and Ling Xue, "Environmental Uncertainty and IT Infrastructure Governance: A Curvilinear Relationship," *Information Systems Research (INFORMS)*, vol. 22, no. 2, pp. 389–399, 2011.

[27] "Munich Network Management Team (MNM)," www.mnm-team.org, 2012.

Design, Development and Implementation of a Hybrid Network with Smart Sensors and Power Line Communication for Monitoring of Underground Electricity Substation

Paulo Sausen and Airam Sausen
Technology Department
Regional University of Northwest State
Ijuí, Brazil
{sausen,airam}@unijui.edu.br

Renê R. Emmel Jr.
State Company for Electric Power Distribution of Rio Grande do Sul – CEEE/RS
Porto Alegre, Brazil
ReneEJ@ceee.com.br

Mauricio de Campos and Camila S. Gehrke
Electrical Engineering Department
Federal University of Campina Grande
Campina Grande, Brazil
{decampos.mauricio,camila.gehrke}@gmail.com

Fabiano Salvadori
Electrical Engineering Department
Federal University of Paraiba
João Pessoa, Brazil
salvadori.fabiano@cear.ufpb.br

***Abstract*— The Smart Sensor Networks not only collect data, but also perform local processing, and may even operate in the system, and thereafter, if necessary, carry out the transmission. However, in some cases hybrid networks systems, that combine wireless with wired structures, may be more appropriate. The objective of this work is to develop a system that integrates a set of smart sensors and communication systems for use in an underground distribution power substation. The underground substation of the distribution system chosen belongs to the spot network located in Porto Alegre city, Brazil. Among all challenges of this work, establish the communication system installed inside the substation with the outside world, is without doubt the most complex, because, there are no commercial solutions for this problem. This paper presents the development of a hybrid smart system based on wireless sensor network combined with Power Line Communication. This system allows real time monitoring of the substation without the need to make any significant changes.**

Keywords-Smart Sensors Networks; Monitoring Underground Power Substation; Hybrid Networks Systems.

I. INTRODUCTION

The need to manage the processes, combined with advances in electronics and wireless communication technologies have allowed the design of the Wireless Sensor Networks (WSN) [1]. The technology applied to these sensors, the data processing and communication networks, has allowed the evolution of these systems, which came to be called smart sensor networks. The sensors not only collect data, but also perform local processing and can act on the system and thereafter, when necessary, perform data transmission. These intelligent sensor networks enable a more effective monitoring and fault detection system, improving reliability and network maintenance [2].

Among the challenges of design, development and installation of smart sensor network can be pointed out environments where electromagnetic interference reduces performance and can also making it inoperable [3,9]. In these cases, hybrid networks that combine wireless systems with wired structures may be more appropriate [4]. These hybrid architectures still allow better power management of these systems, since in some cases the sensor node can be installed in difficult access areas. Thus, the physical connection can also be used as the redundant system of communication system.

The objective of this work is to develop a system, integrating smart sensors and communication systems, for use in a power electric underground distribution substation. The underground substation of the distribution system chosen belongs to the spot network located in Porto Alegre city, Brazil. The depth of this substation ranges 4-5 meters, under layers of asphalt and concrete. Therefore, another challenge of this work was to establish a communication system capable of communicating from the inside to the outside of the substation, since it is not possible by radio communication and there are not available physical communication installed for this purpose.

The rest of this paper is organized as follows. Section II presents the Underground Distribution Network. Section III presents a description of the monitoring system of substation. Section IV describes the partial results obtained from the monitoring system, and Section V concludes this work.

II. UNDERGROUND DISTRIBUTION NETWORK

The underground distribution networks represent an advantageous alternative for applications in distribution systems in great urban centers, which are characterized by high concentrations of charge and require high levels of quality, continuity and reliability of electricity supplies.

There are two common ways of connecting underground distribution networks, the radial or network systems. The network system, also known as spot network system, is a low voltage distribution system, having a set of transformers connected in parallel, to supply electric energy to the load. This topology allows that the electricity supply is maintained even that one or more transformer get out the service as long as the total power of the remaining transformer is equal to, or greater than, the power consumed by the load. Moreover, it allows improving secondary voltage characteristics [5].

The spot network system is installed in Porto Alegre city, fed with 13.8 kV primary voltages, and 127/220V secondary voltages. It consists of 500kVA transformers, submersible, hosted in underground chambers.

The greatest risks in this type of system are: inundation, overheating, fault in protection system, theft, and changes in system pressure [10]. Figure 1 shows the analog monitored quantities: (i) the current in the primary; (ii) the voltage and current on the secondary; and, (iii) temperature of the transformer frame and the environment temperature. The other quantities are digital type (on or off), e.g., the states of: pumps, fans and transformer operational lights. The system must also be able to monitor substation inundation and intruders.

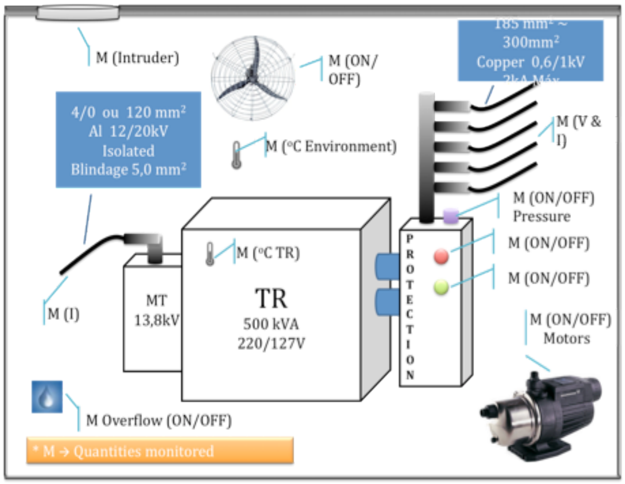


Figure 1. Monitored data by the system.

III. Designed System Architecture For Monitoring

The designed system (see Figure 2) is based upon the concept of intelligent sensors. The Intelligent Sensors Modules (ISMs) may acquire up to four magnitudes, i.e., two analogs and two digital, communicating by wireless and/or physical network. A second module is designed to be used in the acquisition of quantities with fast dynamic and need read more than four quantities, e.g., secondary voltages and currents of the transformer. This device is referred as Remote Data Acquisition Unit (RDAU).

The Gateway establishes the communication with the outside. As earlier mentioned, is not possible to carry out communication by radio or wired structure, since the characteristics of the substations does not allow the deployment of these systems. Thus, we used a Power Line Communication (PLC) system, allowing data transmission from the inside the substation. In the outside of the substation, a General Packet Radio Service (GPRS) transmits the data in 3rd Generation (3G) cellular communication system to a server.

The monitoring system has essentially the following subsystems (see Figure 2): a) Sub-system for data acquisition, b) Remote link, and c) Control subsystem.

A. *Intelligent Sensor Module – ISM*

The ISMs are devices capable of performing data acquisition functions, data processing and transmitting/receiving data. Its architecture (Figure 3) consists of a power subsystem, a sensor subsystem and communication subsystem.

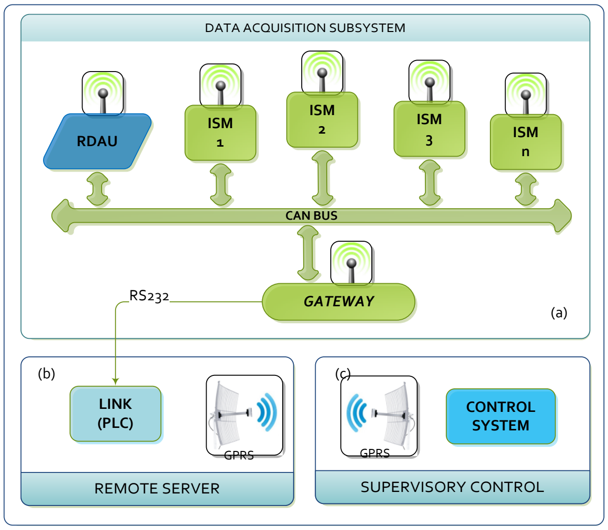


Figure 2. Complete monitoring system.

The subsystems, sensors and communication are managed by a PIC18F2580 microcontroller. This was chosen because of design requirements and also has the same integrated hardware dedicated to CAN (Controller Area Network). In addition, supports various peripherals, e.g., 10 bits Analog-Digital Converter (ADC), four timers, Universal Synchronous Asynchronous Receiver Transmitter (USART) as serial interface, among others.

The power subsystem is responsible for powering the ISM. The primary source of energy comes from the CAN bus and/or battery pack. When necessary, the CAN bus system also feeds back into recharging of batteries. This system consists of a battery with 900mAh capacity and 7.2 V.

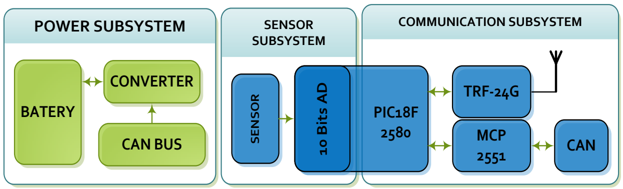


Figure 3. ISM architecture.

ISBN: 978-1-61208-226-4

The ISM is equipped with four sensing inputs, two digital and two analogs. The analog inputs designed to operate with signals in the range of 0 to 5V or 4 to 20mA, depending of characteristic of the sensor connected. If the sensor attached to the ISM need power, it is provided with the signal connector.

The ISM uses the communication subsystem to send/receive data in two distinct ways: via wireless network or through physical network. The physical network is primarily intended for the redundancy, the wireless network is the principal communication to exchange information. The device used to radio frequency (RF) communication is the TRF-24G module, which employs the nRF2401A transceiver. This device uses modulation GFSK (Gaussian Frequency Shift Keying) [12] for transmitting up to 1 Mbps. It features integrated antenna and the transmission power can be set from -20 dBm to 0, allowing a range of 250 meters (without obstacles).

The physical bus addresses the standard ISO11898-2, designed to international standard CAN communication [6, 7]. It specifies patterns relating to the physical layer of the CAN protocol, one being the use of a transceiver device that makes the interface between the sensor and CAN bus node, making certain electrical conditions provided in the standard are met. Amongst these, conditions include the protection against short circuits, voltage levels and others. Therefore, ISMs were connected to the bus via the CAN transceiver MCP2551, Microchip Technology ^{TM}.

ISM prototype (see Figure 4) was developed to experimental validation. Each ISM has an address assigned by the Gateway when installing the network, organizing themselves autonomously (plug and play).

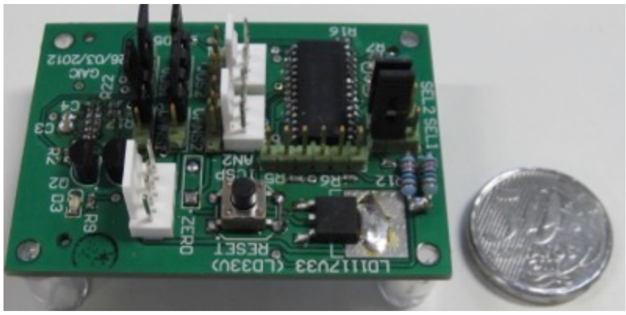


Figure 4. Intelligent Sensor Module (ISM).

B. Remote Data Acquisition Unit - RDAU

Figure 5 shows the diagram of the Remote Data Acquisition Unit (RDAU), it can be seen that the RDAU is divided into three blocks: Communication Subsystem responsible for communication device; Sensor Subsystem responsible for acquisition of the voltage and current; and Power Subsystem that provides power to the system.

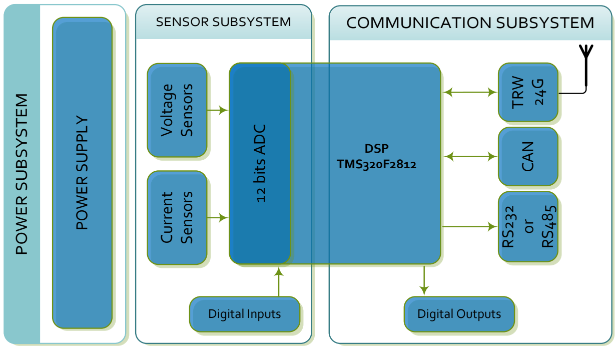


Figure 5. Remote Data Acquisition Unit (RDAU) Diagram.

The RDAU (see Figure 6) can communicate with the Remote Server (RS) via RS 232, RS 485, CAN bus or RF. If using the RS 485, CAN or RF, is it possible to connect multiple RDAUs to each Link Remote. Is controlled by a Digital Signal Processor (DSP) family TMS320F2812. The ADC is 12 bit and is programmed to 240 samples/cycle acquisition. Figure 5 shows the block diagram of RDAUs.

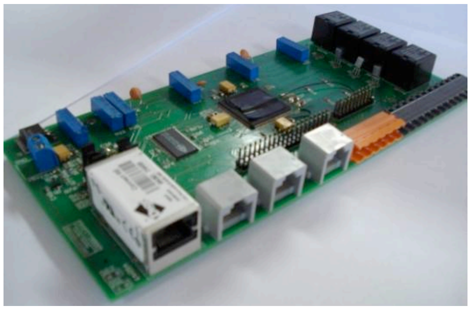


Figure 6. Remote Data Acquisition Unit (RDAU).

The RDAUs are connected by CAN bus for communication with the RS, but also has an RF communication module (model TRF-2.4G) used to perform communication with the Gateway and ISM devices.

C. Gateway

It has been also developed a Gateway, which is responsible for interconnecting the set of sensors (ISMs + RDAU) and the PLC transmission system. The essential difference from the Gateway to the ISM itself is that there is an additional RS232 serial communications port used to perform the interconnection with PLC. The Gateway physical appearance can be seen in Figure 7.

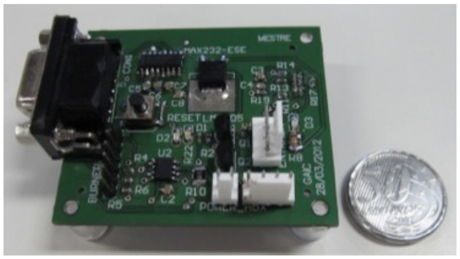


Figure 7. Developed Gateway.

The information exchange between Gateway and ISM is under MODBUS communication protocol. This is a master-slave protocol. Defines a structure of communication messages used to transfer analog and digital data between microprocessor devices, with detection and information of the transmission errors.

The MODBUS protocol is located at 7th level of the OSI Reference Model, which corresponds to the application layer that provides "client / server" communication between devices connected to different types of buses or network topologies [8]. The MODBUS also allows easy integration with SCADA systems [11], although these are not the main focus of this work.

The management and addressing the ISMs are performed by the Gateway, which in turn, updates and constantly checks the presence of new ISMs that for luck are connected to the bus.

D. *Modem PLC*

The PLC system has been installed in Porto Alegre city, in the low voltage cabling of underground network. Is a PLC, transmitter/receiver pair, developed from a MODEM PLC PL-3120, ECHELON^{TM}. In this model, a microcontroller whose functions are listed below, is connected:

- Transmitter/Receiver PLC installed in the transformer;
- Data acquisition of the environment temperature and transformer frame;
- Generation of data packet to send to the MODEM PL-3120 via the serial interface (UART);
- Management control messages sent through the electric grid, supplied by MODEM PL-3120.

Transmitter/Receiver PLC installed outside is responsible for:

- Receiving the data packets sent through the electric grid, supplied by MODEM PL-3120
- Checking validity of data received;
- Configuration of Modem GSM/GPRS;
- Generation of data packet for sending for the GSM MODEM / GPRS via UART serial interface;
- Control messages management sent via the cellular network, delivered by the GSM MODEM / GPRS.

The PLC MODEM PL-3120 NEURON incorporates a CPU, 4 Kbytes to application memory and 2 kbytes of RAM. The NEURON^{TM} processor executes routines for nodes protocols interconnection in a network PLC, Interoperable Self Installation (ISI), besides communication protocols, with the option to activate or not the CENELEC protocol. All these protocols are proprietary and are stored in ROM memory on the device. In Figure 8, the blocks diagram can be seen, with the constituent parts of a node based on PLC PL-3120.

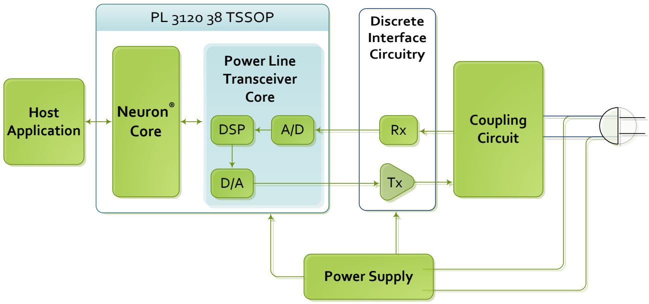


Figure 8 PLC node based on PL-3120.

The MODEM PL-3120 can operate in bands A and C defined in CENELEC STANDARD, which are selected from the crystal used to trigger the MODEM. The selection of the CENELEC band also defines the rate of data transmission on the network. By selecting the A band, the communication will occur at a rate of 3.6 kbps.

As presented in the block diagram (Figure 8), it is necessary for integration, between the PL-3120 and the circuit that couples the modulated carrier to the electric network, an interface circuit. The interface circuit is composed mainly of an amplifier that can be applied to an electric network signal at operation carrier frequency of the PL-3120, with up to 1A peak-to-peak. Figure 9 shows the circuit diagram of the amplifier output, which forms part of the interface circuit. It is a transistor discrete circuit in a push-pull modified configuration.

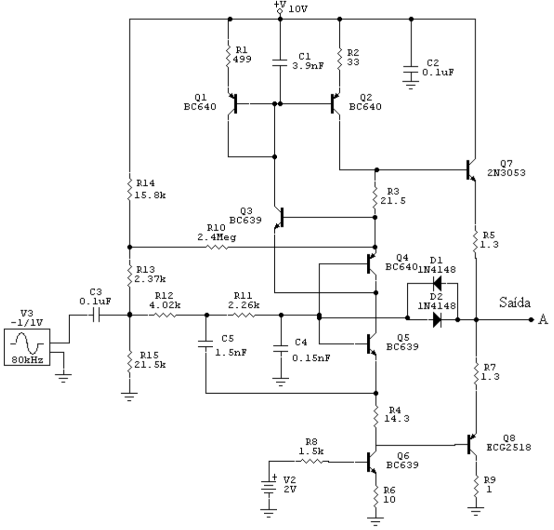


Figure 9. TX amplifier.

Figure 10 presents a frequency response analysis of the power amplifier for PLC transceiver. There is a practically flat response in the frequency range of 1kHz to 20kHz. In the frequency range corresponding of the band A, of the STANDARD CENELEC, there is a peak in the curve of the amplifier gain. The maximum peak occurs at 100kHz, falling abruptly after this frequency.

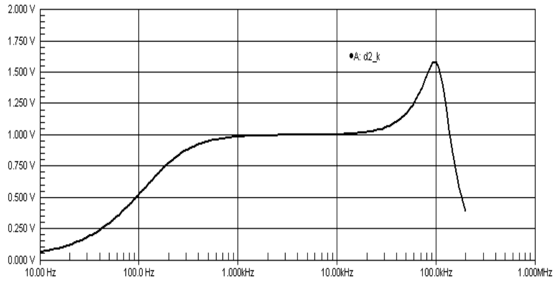


Figure 10. TX amplifier frequency response.

IV. PRATICAL RESULTS

The tested system was installed in the spot network system of the CEEE-D (State Company for Electric Power Distribution) in Porto Alegre city.

The monitoring system (ISMs and RDAU) was installed in the northeast spot network system (RNE), box-manufacturing T-103-7A (code CEEE-D), which has the feeder 2RNE as energy supplies. The developed Gateway manages the receipt of the data system and is connected to the PLC signal transmitter, in the low voltage transformer output. The approximate distance from the transmitter to the receiver is about 250 meters, Figure 11. There is no direct path between them.

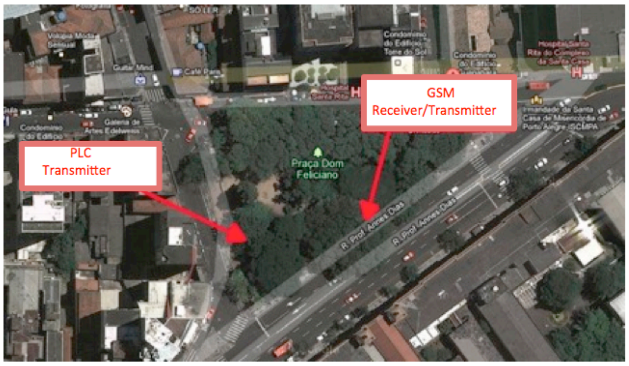


Figure 11. PLC transmitter and receiver signals location.

Due to the robustness provided by the adoption of hybrid structure for the ISMs and RDAU, packet losses in communication did not occur. The most critical data such as voltage and current, travel through CAN system when the data is not received properly by transceivers.

The proposed system was tested for 90 days collecting data at intervals of 10 seconds. In this period, more than 2 Gb of information that is stored in a database were transmitted, the packet loss rate was less than 1%, which proves the robustness of the proposed system. A graphical interface able to access this database was developed, which can be executed from any desktop or mobile device. More details on this interface are presented in sequence of the paper.

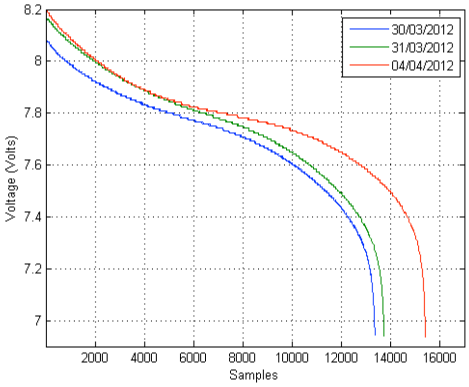


Figure 12. Backup system test (critical case).

The battery system of sensors nodes operates as backup in cases, which the redundancy occurs. In these cases, the worst possible condition occurs when the sensor node is continuously processing and transmitting data, where his current drawn reaches 57mA peak. Thus, we tested a set of batteries in extreme conditions of use, so that could be evaluated its durability. Figure 12 shows the results obtained in the discharge process. The sampling time is 1 second.

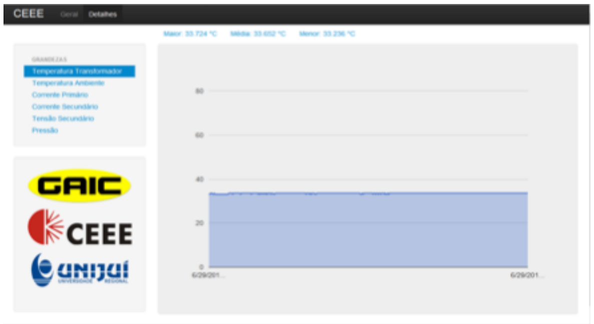


Figure 13. Real time interface for WEB.

Figure 13 presents the WEB application developed to access the data collected and stored in the database. Any user can to access this application from a computer with Internet access.

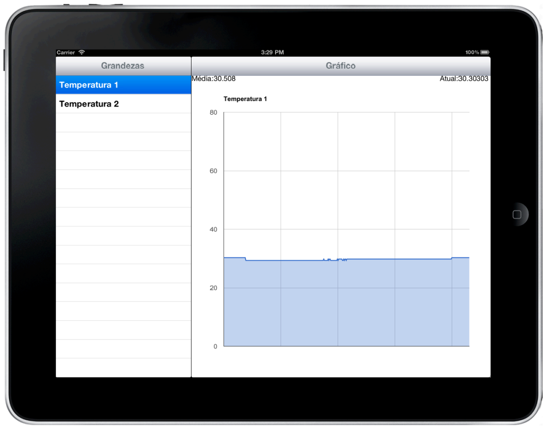


Figure 14. Real time interface for IOS (e.g.,IPAD).

ISBN: 978-1-61208-226-4

Figure 14 shows the application developed specifically to run on devices with IOS operating system (e.g, ipad and iphones). Finally, Figure 15 shows the application designed to run on devices with Android operating system.

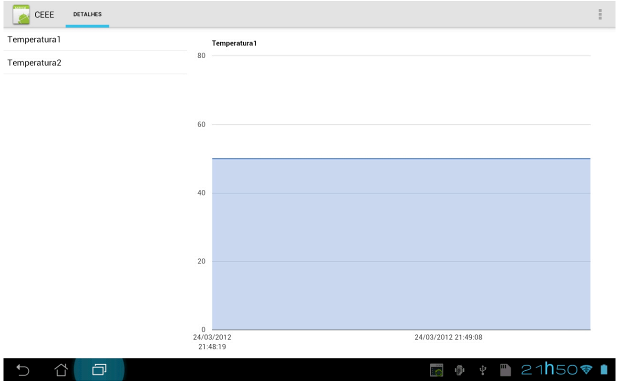


Figure 15. Real time interface for Android.

V. Conclusions And Future Work

This paper presented a monitoring system designed to monitor an underground substation power distribution. Advances in electronic communication systems and processing, and the high degree of integration, enabled the development of a high performance system for these applications.

Among the challenges of this application may be highlighted the communication between indoor and the outdoor of the monitored substation. Furthermore, considering the difficulty of access to the system, determined the use of a hybrid system eliminating the necessity of regular maintenance of the batteries.

The system allows its application in intelligent systems and fault detection applications in underground spot network system. As future work we intend to investigate different modulation types for communication PLC in order to improve the transmission rate.

Acknowledgment

This work was partially funded by Companhia Estadual de Energia Elétrica do Rio Grande do Sul (CEEE-RS, Brazil) and Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS, Brazil).

References

[1] P. S. Sausen, J. R. B. Sousa, M. A. Spohn, A. Perkusich and A. M. N. Lima, "Dynamic power management with scheduled switching modes in wireless sensor networks," *15th IEEE MASCOTS,* pp. 1–8, 2007.

[2] F. Salvadori, M. de Campos, P. Sausen, R. de Camargo, C. Gehrke, C. Rech, M. Spohn and A. Oliveira, "Monitoring in Industrial Systems Using Wireless Sensor Network With Dynamic Power Management," *IEEE Transactions on Instrumentation and Measurement,* vol. 58, no. 9, pp. 3104-3111, Sept 2009.

[3] F. Salvadori, M. Campos, R. Camargo, C. Gehrke, C. Rech, P. Sausen, M. A. Spohn and A. Oliveira, "Monitoring and diagnosis in industrial systems using wireless sensor networks.," *Intelligent Signal Processing, WISP 2007 IEEE International Symposium on,* pp. 1–6, 2007.

[4] G. Sharma and R. R. Mazumdar, "A case for hybrid sensor networks.," *IEEE/ACM Transactions on Networking,* vol. 16, no. 5, pp. 1121-1131, Oct 2008.

[5] M. R. Gouvêa, E. C. Belvedere, J. J. Oliveira, P. E. Mascigrande, A. P. Costa and R. Brunhetoro, "Development of Standards for Underground Hybrid Networks (in portughese)," Aneel, 2005. [Online].Available:<http://www.aneel.gov.br/biblioteca/downloads/livros/desen_redes.pdf>. [Retrieved: June, 2012].

[6] P. Richards, *AN228 - A CAN Physical Layer Discussion. Microchip Technology Inc.,* 2002. [Online]. Available: <http://pt.scribd.com/doc/99612719/00228a>. [Retrieved: June, 2012].

[7] M. Esro, A. A. Basari, S. Kumar and A. a. S. Z. Sadhiqin, "Controller Area Network (CAN) Application in Security System.," *World Academy of Science, Engineering and Technology,* 2009. Available: <http://sharepdf.net/view/48565/controller-area-network-can-application-in-security-system>. [Retrieved: June, 2012].

[8] M. ORG., MODBUS over Serial Line Specification and Implementation guide. V1.0., 2002. [Online]. Available: <http://www.modbus.org/docs/Modbus_over_serial_line_V1.pdf>. [Retrieved: June, 2012].

[9] S. Roostaee, R. Hooshmand and Mohammad Ataei, "Substation Automation System Using IEC 61850," Proc. IEEE Power Engineering and Optimization Conference (PEOCO), 2011, IEEE Press, june. 2011, pp. 393-397, doi:10.1109/PEOCO.2011.5970443.

[10] F. Mioqi, W. Jian, X. Xianghua and w. Guangrong, "System for Temperature Monitor in Substation with ZigBee Connectivity," Proc. IEEE Communication Techology (ICCT), 2008, IEEE Press, Nov. 2008, pp. 26-28, doi:10.1109/ICCT.2008.4716095.

[11] N. Yellamandamma, T. Sai Kumar, KVH Rao and A. Aggarwal, "Low Cost Solution for automation and control of MV substation using MODBUS-SCADA," Proc. IEEE Power Sytems (ICPS), 2009, IEEE Press, Dec. 2009, pp. 1-6, doi:10.1109/ICPWS.2009.5442735.

[12] M. Gast, "802.11 Wireless Networks: The Definitive Guide", Second Edition, O'Reilly Media, Inc., April, 2005, 656 p.

Computing Individual Mobility Profiles from Mobile Phone Usage Traces

Miguel Á. Rodríguez-Crespo, Ana Armenta, Alberto Martín-Domínguez, Rocío Martínez-López, Rubén Lara
Telefónica I+D
Madrid, Spain
[miguel, aalv, amd, rml, rubenlh]@tid.es

Abstract—**This paper describes the procedures used to automatically generate a complete individual mobility profile of users of mobile services, based exclusively on geo-located phone usage information and without requiring any customer interaction. It also presents the results of applying these procedures in a test with real data. The individual mobility description includes metrics such as area of activity, diameter of influence area, location and meaning of the user's points of interest, and frequent itineraries.**

Keywords*— *Individual mobility; mobile phone usage; center of mass; radius of gyration; points of interest; itineraries

I. INTRODUCTION

The study of human mobility patterns has received growing attention over the past few years, especially due to the increasing availability of location data coming from both global positioning systems (GPS) and mobile telephone usage, which leaves geo-located traces on the operators networks [1][2][3][4].

Understanding how and when human movements take place across our towns, cities or countries is of interest in many areas, such as traffic management, transport network design or diseases spread control [5][6][7]. However, not only a global view of population flows, but also individual mobility patterns of a user, are of great interest in a number of fields [8]. The knowledge of what locations a user periodically visits, over what period, with what frequency, what days of the week and at what times of the day [9][10] can be used for the provision of contextual services, relevant advertising, targeted offers to address the particular mobility needs of the user, itinerary planning… In general, knowing locations that are relevant for a user can enable the personalization of commercial communications or service interactions and improve their relevance.

This paper describes the set of procedures implemented to obtain a complete description of the mobility profile for mobile phone users. These procedures have been tested over several million customers from the same country, treated in an anonymous way. Future utilization of these procedures will imply customers give their previous authorization to treat their unanonymized id so they can be offered customized services or applications.

The paper is structured as follows. Section 2 describes the general framework and the kind of geo-located data we use to obtain these profiles. In Section 3, we explain how we compute certain areas for every customer. Section 4 explains how we detect and label the points of interest of each customer. Section 5 describes how we detect the frequent itineraries of a user. Finally, in Section 6, we present some conclusions and ideas for future work.

II. GENERAL FRAMEWORK

The method used for computing individual mobility profiles works on geo-located events generated by using mobile phones. These geo-located events are obtained in a non-intrusive way for the users. These events are traces that mobile phone usage leave on the network and include initiation and termination of voice calls, sending short text messages (SMSs) or multimedia messages (MMSs), signaling (as data sessions attach/detach events), and so on.

Geo-located events must contain, at least, the following information: an anonymized user id associated to the event, and the date, time and location of the event.

Location information is usually available at a Base Transceiver Station (BTS) level. BTSs are the places where the antennas that receive and transmit the radio signals from and to the mobile phone users are located. So the location information in fact refers to an approximate area, not an exact place. Even more, the customers' location is not known as a continuous function of time; it is only known at certain points in time when customers are interacting with the mobile phone network.

These kinds of events are collected over a period of time to obtain the dataset used to compute the mobility profiles of the users.

The work and results presented in this paper are based on data collected from Call Detail Records (CDRs) of voice calls of customers from a Latin American country over a six-month period. These data comprise about 7 billion CDRs, from more than 16 million customers. There are more than 5000 different locations (BTSs) over the whole country.

III. CENTER OF MASS, RADIUS OF GYRATION AND DIAMETER OF INFLUENCE AREA

First, some simple but informative descriptors are calculated. These descriptors give information about the areas where the customers are usually located over the time period considered.

For each customer, a set of locations is obtained, each with a count indicating the number of times (n_i) we observe the customer at that location. Each location is represented by a pair of 2-D planar coordinates *(x_i, y_i)*. The center of mass (*CM*) for a customer is obtained as the location whose *CMx* and *CMy* coordinates are calculated as the weighted average of all the locations known for that user over the time period.

$$CMx = \sum_{i=1}^{N} n_i x_i / \sum_{i=1}^{N} n_i \qquad (1)$$

Copyright (c) IARIA, 2012.   ISBN: 978-1-61208-226-4

$$CMy = \sum_{i=1}^{N} n_i y_i / \sum_{i=1}^{N} n_i \tag{2}$$

As the activity of the customers can be very different for workdays and weekends, two centers of mass are calculated: a workdays center of mass (CMx_{wd}, CMy_{wd}) from the activity observed for the user from Monday to Friday, and a weekends center of mass (CMx_{we}, CMy_{we}) from the activity observed on Saturdays and Sundays.

Once the two centers of mass are obtained, two radii of gyration (R_{wd} and R_{we}) are calculated for each one. A radius of gyration is computed as the weighted average of the distances from the customer registered locations to his/her center of mass.

$$R = \sum_{i=1}^{N} n_i \sqrt{(x_i - CMx)^2 + (y_i - CMy)^2} / \sum_{i=1}^{N} n_i \tag{3}$$

The combination of center of mass and radius of gyration defines a circular area where the customer concentrates most of his activity. Two circular areas are obtained for each customer, one for workdays, given by (CMx_{wd}, CMy_{wd}) and R_{wd}, and another one for weekends, given by (CMx_{we}, CMy_{we}) and R_{we}.

Fig. 1 shows an example of the activity areas for a user. This user clearly changes his/her known phone usage locations from workdays to weekends.

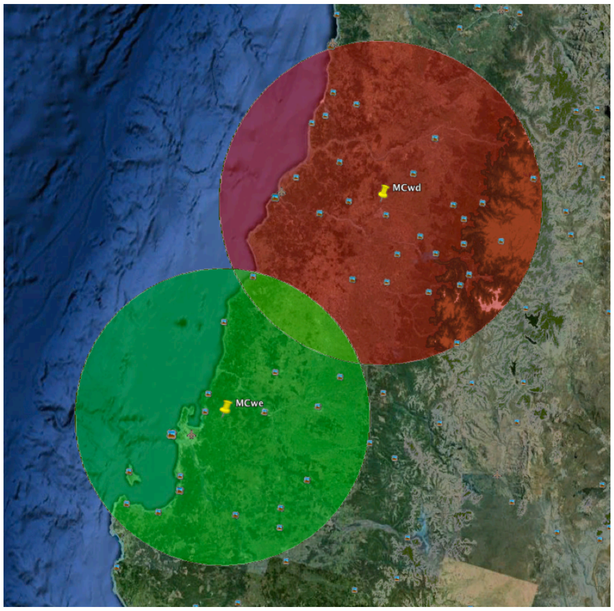


Figure 1. Workdays (red) and weekends (green) activity areas defined by centers of mass and radii of gyration for a user.

Another parameter is calculated to complement the information given by the centers of mass and radii of gyration: the maximum distance between two registered locations for a customer over the period of study (diameter of the influence area). This information allows identifying users that have gone to very distant places during the time period and users that have stayed in a more limited area.

From a general analysis of the locations visited by the customers, we can derive some information:

- 50% of the customers use less than 9 different BTSs during workdays and less than 7 on weekends. Just 5% of the customers visit more than 51 different BTSs during workdays.
- The radii of gyration are less than 4 km for 50% of the customers and are very similar for workdays and weekends. Just 5% of the customers have radii of gyration greater tan 17 km.
- 50% of the customers have an influence area diameter lower than 20 km in workdays and lower than 18 km in weekends.

IV. Detection and Labelling of Points of Interest

The second component of the individual mobility profiles is the detection and labeling of the users' points of interest (PoIs). The goal is to find the most relevant locations for every user, from the set of locations he/she had activity at, and to characterize them by labels that give information about what each PoI means for that user.

A. Detection of PoIs

First, the CDRs of users that have too many or too few calls during the time period are filtered out. This way we avoid the noise introduced by users not corresponding to an individual usage (i.e., switching devices or PBXs of a company or business), or corresponding to individuals whose activity is too low to extract a meaningful usage pattern. A lower limit of 200 calls and an upper limit of 5000 calls over the time period were established. More than 8 million customers are between these thresholds. For these customers, BTSs having at least 5% of the total number of each customer's calls are selected. The customer-BTS pairs that satisfy these requirements define the PoIs of that customer.

B. Labelling of PoIs

Every PoI is characterized by a communication vector that represents its temporal usage pattern. For one PoI (customer-BTS pair) this communication vector initially contains the number of calls the customer did in the BTS at every different hour of the different weekdays (from Monday to Sunday), aggregating the values for the same weekdays of the whole time period. As not every weekday has the same meaning in terms of life activity patterns, it was decided (after a previous analysis) to group Monday, Tuesday, Wednesday and Thursday in one single kind of weekdays (Monday-Thursday), and to maintain Friday, Saturday and Sunday as separate kinds of weekdays.

So, for every customer there are several communication vectors (curves), one for each of his/her PoIs, containing the number of calls made by that customer in that location at every hour of 4 kinds of weekdays (Monday-Thursday, Friday, Saturday and Sunday). Each vector has $4 \times 24 = 96$ values.

As the number of different kinds of weekdays is not the same, a first normalization is done dividing each vector value by the number of days of its kind of weekday that occurs in the time period. This allows comparing the 4 different parts of the curves.

In order to allow the comparison between different curves from different PoIs with very different activity levels (number of calls), a second normalization is done dividing each vector value by the sum of all the vector values. All normalized PoI communication vectors will have a resultant sum equal to 1. This way the differences between vectors focus on the curve shape itself reducing the importance of the curve amplitude levels.

Fig. 2 shows the normalized communication vectors of two PoIs, with different shapes that represent different kinds of phone usage. The first one is more uniform and regular over the different weekdays. The second only presents activity from Monday to Friday, especially in the evenings.

In order to label the communication vectors, it is assumed that different shapes (usage patterns over the week) imply activities of different nature at those locations. A set of clusters (groups) is obtained from the whole set of PoI communication vectors of all the customers. A meaningful label is assigned to each cluster, related to the usage pattern over time.

About 24 million PoIs (communication vectors) were obtained from the dataset. A representative sample of this set was selected to run the clustering algorithm (clustering training). A partitioning around medoids (PAM) algorithm was used. This algorithm allows the use of different types of distances to represent the dissimilarity between vectors. In particular, a distance based on the Pearson correlation coefficient between any two communication vectors a and b was used. The distance (dissimilarity) between two vectors was calculated as $1 - \rho_{ab}$, where ρ_{ab} is the Pearson correlation coefficient (similarity). So, the distances have a minimum value of 0 (exactly the same shape) and a maximum value of 2 (exactly the opposite shape).

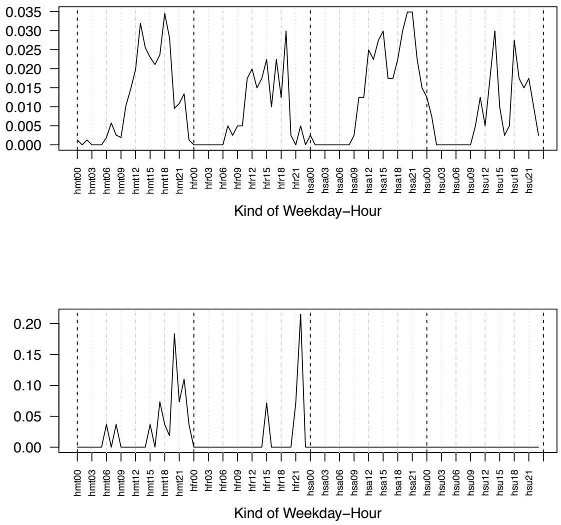


Figure 2. Normalized communication vectors of two PoIs.

The clustering over the sample of PoI vectors produces a vector cluster id (group) assigned to each vector in the sample. All the vectors with the same cluster id belong to the same group, and a representative vector for the cluster id is also obtained. The representative vector of each cluster id is the cluster medoid, that is one of the vectors assigned to that cluster whose average dissimilarity to all the vectors in the cluster is minimal. The cluster centroid can also be obtained as an average of all the sample vectors found inside the same cluster.

Once the medoids are obtained, the PoI vectors not used in the clustering (or any other PoI vector that could be built later) can be assigned a cluster id calculating the distances based on the Pearson correlation to the different medoids and choosing the cluster id of the representative that is closest to the new input vector (has the minimum distance to it).

The number of cluster or groups has to be given as an input to the training phase of the clustering algorithm. As we would like to detect many different groups for the PoI vectors to find enough different types of locations for a user, a value of 20 clusters was used.

Using the medoid and the centroid for each cluster, a label is assigned to them based on the knowledge of the social habits and cultural characteristics of the country. These are the level 0 labels (detailed set of labels).

Fig. 3 shows 4 examples of cluster representatives, along with the level 0 labels assigned to each. The percentage of vectors of the training sample found in each cluster is also shown.

The cluster representatives are later grouped into 5 level 1 labels, thinking on practical applications that will not need as much detail as given by some of the 20 level 0 labels. So, any PoI vector is assigned the level 0 label of its closest medoid, and the level 1 label is assigned based on a table that associates a level 1 label to each level 0 label.

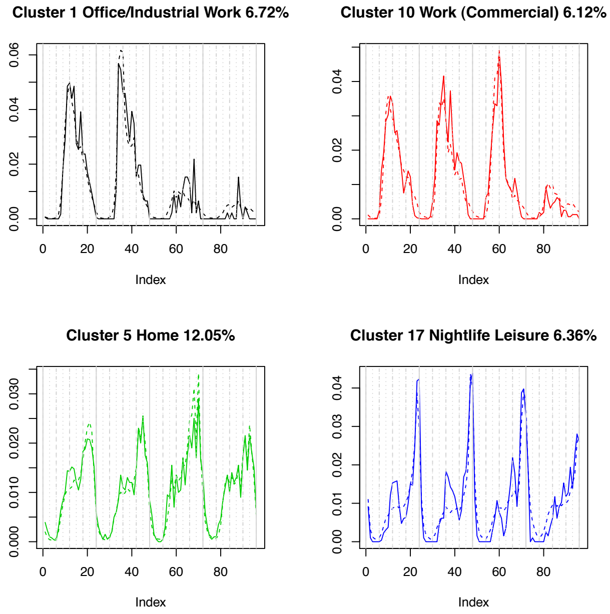


Figure 3. Example of medoids (continuous lines) and centroids (dotted lines) for 4 clusters, with their level 0 labels and percentages in the sample of PoI vectors used in training the clustering.

The level 1 labels and the percentage of training vectors in each of them are:

- Office/Industrial Work 10.03%.
- Commercial Work 16.67%.

- Home 39,00%.
- Night/Evening Leisure 9.31%.
- Afternoon Leisure / Shopping 24.99%

A labeling confidence flag (0/1 value) is obtained based on a distance threshold per cluster. The distances of the training vectors to the medoids of their assigned cluster are used to compute the distances mean and standard deviation for each cluster. The sum of the mean and standard deviation is the threshold distance for each cluster. If the distance of a PoI vector to its closest cluster representative is higher than the threshold distance for that cluster, then the label assigned to that PoI vector is not considered reliable enough (the labeling confidence flag is 0). Otherwise the labeling confidence flag is 1. About 15% of the PoI labels are not considered reliable enough using this criterion.

C. Comments on the Results of PoI Detection and Labelling

The output of the PoIs detection and labelling step is a set of locations (BTS's) of special interest for each customer. Each PoI is automatically labelled at two different levels (level 0 and level 1) including a labeling confidence flag. The labels express the particular meanings of the locations for each user.

Fig. 4 shows some distributions of the number of PoIs and labels for the whole set of customers whose data have been processed. More than 8 million customers have at least one reliable PoI (labeling confidence flag=1) and more than 20 million reliable PoIs are detected for those customers, giving an average number of 2.5 PoIs per customer. The black line shows that the highest number of PoIs reaches a value of 12. But, the highest number of different level 0 labels (red line) does not exceed the value of 9 (one customer can have more than one location with the same label or meaning). Most of the customers have 5 or less PoIs, and also 5 or less different level 0 labels. Regarding the number of different level 1 labels (green line), the upper limit of 5 is only reached by a very low percentage of customers.

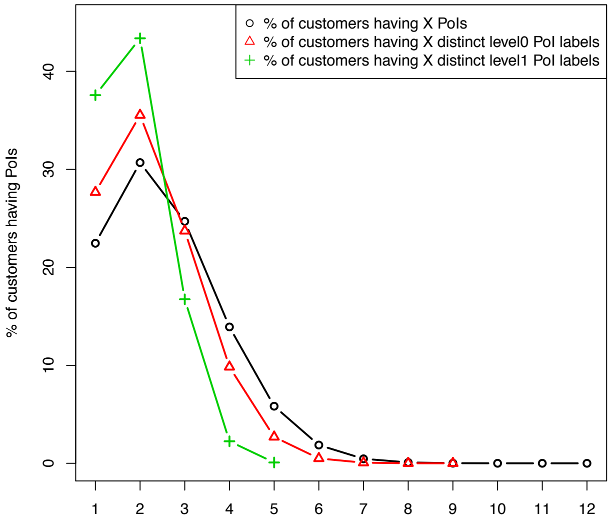


Figure 4. Percentage of customers that have X PoIs with labelling confidence flag=1 (black line), X distinct level 0 labels (red line) and X distinct level 1 labels (green line).

As each customer profile includes his/her centers of mass and radii of gyration, the customer's PoIs information is enriched by flags indicating whether a PoI is located inside or outside the activity area defined by one center of mass and radius of gyration. This is done both for workdays and weekends activity areas to observe if there are any variations depending on the type of PoI.

Fig. 5 shows the percentage of PoIs of a particular type (level 1 label) outside the workdays activity area (red line) and outside the weekends activity area (green line). It includes all the PoIs (even those with labeling confidence flag=0). The two horizontal dashed lines show the mean values of being outside the activity areas (workdays and weekends) for the whole set of PoIs.

As expected, most of the PoIs are found inside the customer's activity areas (almost 80%), as they represent important places for the customers, which concentrate most of their activity. Some types of PoIs are clearly above or below the mean value. There are variations in the percentages of the PoIs being outside depending on considering the workdays activity area or the weekends activity area. Fig. 6 illustrates those variations, showing the percentage of increment of a PoI type being outside the weekends activity area relative to being outside the workdays activity area. The PoI type labeled as "Office/Industrial Work" has the highest weekends/workdays variation (an 80% variation). It is a positive variation, indicating that is much more likely that an "Office/Industrial Work" PoI is outside the weekends activity area than it is outside the workdays activity area (this is consistent with the meaning of that label). The PoI labeled as "Home" has the highest negative variation. It is less likely that the "Home" PoI is outside the weekends activity area than it is outside the workdays activity area (people usually spend more time at home during weekends than during workdays).

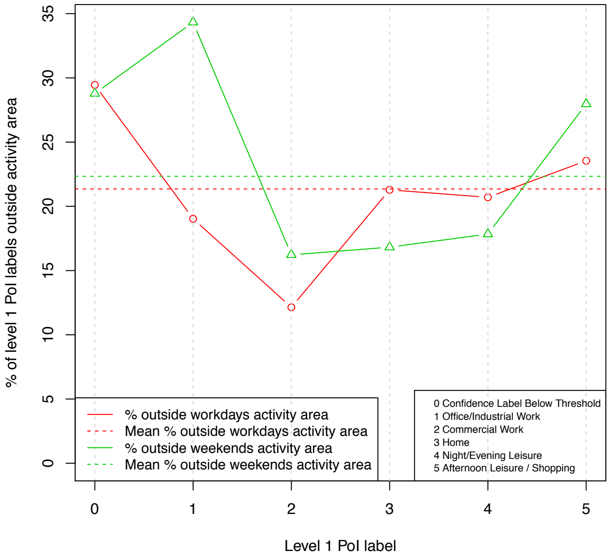


Figure 5. Percentage of level 1 PoI labels outside their customer's activity area (red, workdays activity area; green, weekends activity area)

ISBN: 978-1-61208-226-4

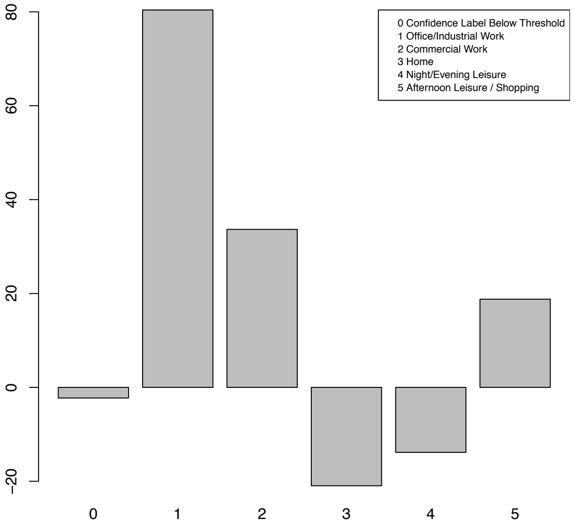


Figure 6. Percentage of increment in the percentage of level 1 PoI labels outside the weekends activity area relative to the percentage of level 1 PoI labels outside the workdays activity area

The relationship between PoI label meanings and their location inside/outside the user's activity areas can be further exploited to refine or even correct the initial PoI labels assignment. For instance, "Home" PoIs detected out of the workdays activity area and far enough from other existing ”Home” PoIs inside the same user's workdays activity area can be labeled as "Second Residence".

V. Frequent Itineraries: Movements between Points of Interest

Focusing exclusively on calls done or received by each customer at his/her PoIs, we consider a movement between two points A and B when we find two consecutive calls, the first one located at point A and the second one located at point B. We establish a maximum time difference of 6 hours between calls to consider the movement is valid. Each movement is spread as a probability over the time period between the two consecutive calls, because the exact time point when the movement takes place is unknown.

A movements curve is obtained as a sum of those probabilities over the six months period under analysis, describing the global probability that the customer moves from PoI A to PoI B at a certain hour of the week. When we find a minimum number of 6 movements between the same 2 points of interest of a given customer, we say we have detected a frequent itinerary between those 2 points. Finally, the itinerary is built as the parts (slots) of this curve above a percentage (20%) of its absolute maximum.

An itinerary description is composed of the peaks from the movements' curve that remain above the amplitude threshold, which represent the most probable time points when such itinerary takes place. The itinerary description also comprises the time-points where the curve intersects the threshold, which describe wider time intervals when the itinerary between the two PoIs implied would probably take place.

For the movements curve depicted at Fig. 7, the corresponding itinerary would contain a set of parameters describing the time positions of the 6 peaks that remain above the threshold (20% of the maximum amplitude), and also the parameters describing the width of those peaks.

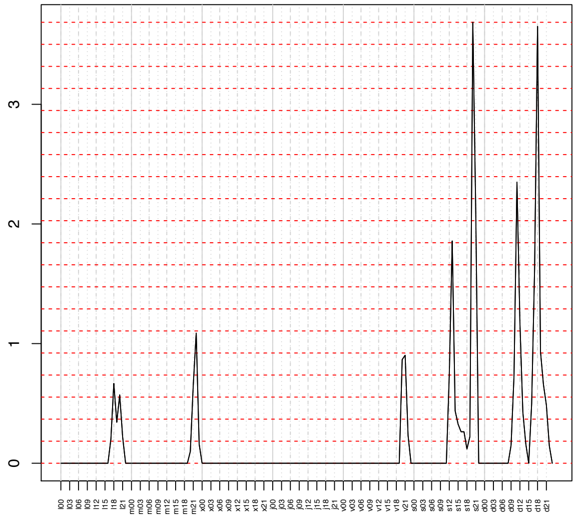


Figure 7. Movements curve between two PoIs of a given customer across the 168 hour intervals of the week (red dotted lines every 5% of the max. amplitude).

Under those assumptions, around 4.5 million customers have more than one frequent itinerary between two of his/her PoIs. The total number of itineraries detected is about 14 million, which gives an average ratio of 3 itineraries per customer. The total number of peaks that reach the 20% maximum amplitude threshold is about 125 million, which implies an average of 8 peaks per itinerary.

The real distributions are not uniform as the two curves in Fig. 8 clearly show. The red line represents the percentage of customers (from the 4.5 million customers that have at least one itinerary) that have X number of itineraries. It has a clear maximum at 2 itineraries per customer and decays quickly to 7-8 itineraries (higher number of itineraries have very low frequency). The green line represents the percentage of itineraries (from 14 million itineraries) that have X number of peaks that exceed the cutoff amplitude threshold. In this case the maximum is located between 5 and 7 peaks per itinerary.

Taking into account the kinds of PoIs that are destination of the detected itineraries, results are aggregated in order to validate internal coherence of the followed procedure. Fig. 9 shows the sum of itineraries across the 168=7*24 hours of the week that start at any kind of PoI with 5 curves for the 5 different categories of PoI as destination. Itineraries to home are majority and take place mainly at evenings of workdays, and at both mornings and evenings of weekends. Itineraries to "Office/Industrial Work" are mainly detected on mornings of workdays. Itineraries to "Commercial Work" are detected more likely from Monday to Saturday during the day hours, similarly to the itineraries to PoIs labeled as "Afternoon Leisure / Shopping". Itineraries to PoIs labeled as

“Night/Evening Leisure” are mainly detected on late night hours.

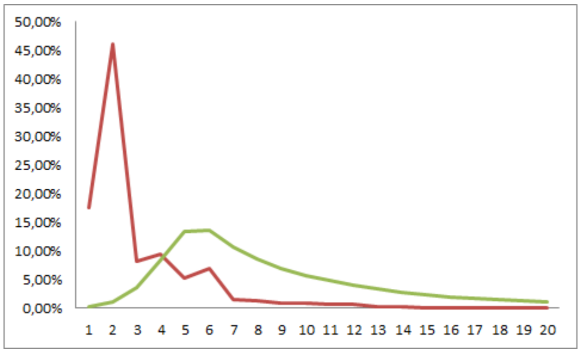


Figure 8. Percentage of customers with X number of itineraries (read line) and percentage of itineraries with X number of peaks (green line).

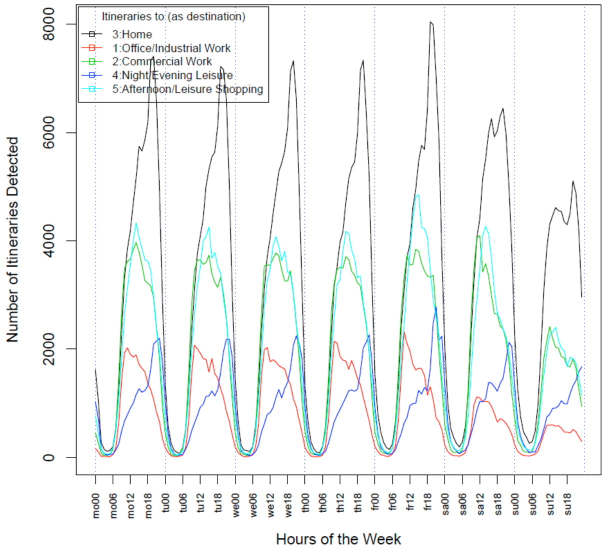


Figure 9. Itineraries detected from any PoI as origin to the different kind of level 1 PoI categories as destination (results based on a sample containing a portion of the total number of itineraries)

VI. Conclusion and Future Work

The mobility profiles we have presented provide a complete description of how mobile phone users move across space and time. The mobility profiles are obtained in a fully automatic way, without any customers’ special interaction. Computed individual mobility profiles comprise workdays and weekends activity areas, influence area diameter, points of interest locations and labels, and frequent itineraries. These mobility profiles have been applied to around 16 million customers of a Latin American country.

Aggregated analysis of individual metrics show internal coherence between several independent procedures. The labels of the PoIs that fall outside workdays activity areas tend to be mainly related to leisure activities while labels of PoIs that fall outside weekends activity areas are much more related to work activities. The hour of the week when itineraries are detected match well with the label of the destination PoI. Itineraries to work PoIs are detected mainly on workdays in the mornings, while itineraries to home PoIs are more frequent in the evenings during the whole week.

Although the described method has only been applied to voice call events, other kinds of geo-located events could also be used to build the individual mobility profiles. The higher geo-located event density over space and time, the higher precision of the individual mobility profile in particular and the social dynamics in general.

This picture of social dynamics is of great interest for companies that want to customize services and applications, to better fit each individual’s needs. Mobility profile based customer segmentation will probably be one of the direct applications of the mobility metrics.

Obviously, also governments and administrations can benefit from the knowledge of human dynamics seen as a whole. In that sense, our purpose is to link computed mobility profiles (both at individual and joint levels) with social variables, to be able to extract a clear picture of how population moves across space and time, and analyze the differences found for different ages, socioeconomic levels, regions, countries, and other segmentation criteria.

References

[1] M. C. González, C. A. Hidalgo, and A. L. Barabási, ‘Understanding individual mobility patterns’, *Nature 453, pp. 779-782* (Jun. 2008).

[2] C. Song, Z. Qu, N. Blumm, and A. L. Barabási, ‘Limits of predictability in human mobility’, *Science 327, pp. 1018–1021* (Feb. 2010).

[3] F. Calabrese, G. Di Lorenzo, and C. Ratti, ‘Human mobility prediction based on individual and collective geographical preferences’, *13th International IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 312-317* (Sep. 2010).

[4] H. Firouzi, Y. Liu, and A. Sadrpour ‘Mobility pattern prediction using cell-phone data logs’, *EECS 545 Final Report* (2009).

[5] F. Calabrese, F. C. Pereira, G. Di Lorenzo, L. Liang, and C. Ratti, ‘The geography of taste: analyzing cell-phone mobility and social events’, *Proc. of the 8th International Conference on Pervasive Computing, pp. 22-37* (May, 2010).

[6] C. Cariou, C. Ziemlicki, and Z. Smoreda, ‘Paris by night’, *NetMob 2010, pp. 62-66* (May, 2010).

[7] R. A. Becker, R. Cáceres, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, and C. Volinsky, ‘A tale of one city: using cellular network data for urban planning’*, IEEE Pervasive Computing, Vol. 10, No. 4, pp. 18-26* (Oct.-Dec. 2011).

[8] B. Csáji, A. Browet, V. Traag, J.-C. Delvenne, E. Huens, P. Van Dooren, Z. Smoreda, and V. D. Blondel, ‘Exploring mobility of mobile users’*, NetMob 2011, pp. 35-37* (Oct. 2011).

[9] R. Becker, R. Cáceres, C. Han, K. Hanson, J. M. Loh, S. Urbanek, A. Varshavsky, and C. Volinsky, ‘Finding meaningful usage clusters from anonymized mobile call detail records’ *Netmob 2011, pp. 47-49* (Oct. 2011).

[10] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Vasharsvky, ‘Identifying important places in people’s lives from cellular network data’, *Proc. of 9th International Conference on Pervasive Computing, pp. 133-151* (Jun. 2011).

User Involvement and Social Networking for Information System Development

Malgorzata Pankowska
Information Systems Department
University of Economics
Katowice, Poland
malgorzata.pankowska@ue.katowice.pl

Abstract— **Since the beginning of business information system (BIS) development, the problem of end-user involvement in the system design and implementation has been discussed. For years, BISs for users have been changed into the systems constructed by end-users. The Web 2.0 technologies should strongly support this tendency. The paper aims to answer the question if end-users are involved and actively participate in BIS development processes. The paper covers results of the survey done in 270 companies in Poland.**

Keywords - innovation, user involvement, user participation, social networking, new media, end-user support

I. INTRODUCTION

User participation in the BIS development seems to be considered as essential for the business success and economic benefits of business organization. However, the placement of end-users in the information system development process is constantly problematic and changeable. Barki and Hartwick [5] proposed a distinction between user involvement and user participation. They define user participation as the assignments, activities and behaviours that users and their representatives perform during the system development process. User involvement refers to the subjective psychological state reflecting the importance and personal relevance that a user attaches to a given system [5]. User participation is defined as the degree to which the user is involved in producing and delivering the service.

This research on end-user participation has focused on how to engage users to increase productivity in the BIS delivering context without failures. The purpose of this research is to collect IT staff opinions on end-user behaviour to fulfil the gap in the literature on user role in BIS development. Furthermore, the theoretical framework linking BIS development to social networking and new media application is also analysed and some empirical findings are given.

Generally, BIS user involvement can take different forms, e.g.,:

- Customer participation in the new service development process directly [11].
- Co-operation of technology providers and users on exploration of the technology in a specific industry [3].
- Partnership i.e., a formal relationship between the customers and the company [6].
- Prosumption, i.e., the dual role played by the customer as a BIS provider as well as a customer [12].

Users involved in BIS development are grouped into the three classes: lead users, normal users and community users [15]. Lead users are defined as members having two characteristics. First, they anticipate obtaining relatively high benefits from the developed information technology (IT) solution according to their needs. Second, they are at the leading edge of important trends in a market place under study. Although normal users' involvement might help provide innovative ideas, their limited understanding of new technologies could threaten the executability of the ideas, therefore there is a need to carefully select normal users for further co-development of IT products. Community users seem to have expertise in a specific field. They are willing to spend more time online on innovations. Business organizations are now exploring ways to provide a platform (i.e., Websites) through which users can generate and contribute content, resulting in a cooperative experience between users and organizations.

For the purpose of this research two groups of users are considered i.e., normal users and community users. The research question guiding the paper concerns the factors influencing user involvement in the BIS development. The paper consists of three parts. At first, theoretical frameworks of user involvement are discussed. The next part covers presentation of research methodology and results. The last part includes findings and conclusion.

II. THEORETICAL FRAMEWORKS OF USER INVOLVEMENT IN INFORMATION SYSTEM DEVELOPMENT

The critical role of end-user in the manufacturing or research and development process encourages the companies to invite users to the value adding processes as partners. Companies should focus on employees' creativity as the essence of innovation and growth. Workers ought to be adaptable and interested in new knowledge and skills acquiring. Lately, new modes of innovation have been introduced and they require openness, i.e., an ability for firms to access and integrate others' ideas in their own practice. "Others" can mean other firms, non-profit units, universities and also other actors - users, consumers, amateurs, volunteers, who are influencing the demand for innovation [9].

Firms are conscious that they are able to turn their intangible assets into value. Beyond research and

ISBN: 978-1-61208-226-4

development (R&D), patents, software, the intangible assets covering human resources and new organizational structures are the drivers of firm's growth. In Europe, the current downturn will be to depress investments in innovation and increase investments in the innovation infrastructure that extends beyond R&D to cover human capital, IT and entrepreneurship for sustainable development. The crisis can be an opportunity for business to consider new policy instruments, which have not yet been implemented at a substantial scale. Since innovation is closely linked to the demand from users, business organizations can promote innovations by being a recipient of end-user creativity results and by signalling an acceptance of new inventions provided by end-users. While more work is needed to ensure that such mechanisms do not distort competition or deliver suboptimal performance, such "pull from the customers" mechanisms can provide incentives for the development and commercialisation of technologies closer to the market - an important complement to traditional innovation policies that have primarily "push" strategy including pushing the technologies to the business.

The IT innovations can start in different parts of a business organization. They can be initiated:

- From technology: new technology or better use of existing technologies make it possible to change work practices.
- From information: intention to use different information or to provide information in a different form or level of detail leads to innovative use of existing or new technology.
- From participants: providing training on technology in a work system leads to new possibilities for doing work differently.
- From work practices: change the business processes to enable use technology more effectively for better results.
- From products and services: improvement of a work system's products and services by incorporating digitized information or even new hardware enables providing additional value for customers [2].

Innovations are always realized in a certain business organization context, where the real value of invention can be estimated. Organizational context is determined by the organizational culture, which includes the shared values, the beliefs, the history, the intellectual and operational traditions, the rules of conduct, and the business organization's general philosophy of operation. Business organization should be strongly engaged in the preparation of workers to develop and implement innovations.

Creating an atmosphere of innovations permits users to behave as innovators. Sometimes user-led innovation involves a community, which creates and exploits innovative solutions on a continuing basis. Good examples of this include the Linux community around computer operating systems or the Apache server community around Web server development applications, where communities have grown up and the resulting range of applications is constantly growing. A growing range of Internet-based applications make use of communities - for example Mozilla Firefox. Social networking and crowd-sourcing approach support co-creating the innovations and new ways of creating and working with such communities need to be developed. In innovation communities, an increasingly significant strategy of prosumption has been implemented, which demands seeing users as active players in that process. Their ideas and insights can provide the starting point for every new direction and create new markets, products and services. The problem is how to find ways of identifying and working with such community users. For example, Microsoft maintains a group of so called Microsoft buddies who work as Web masters, programmers, and software vendors [15]. High technology firms have recognized the importance of linkages and connections - getting close to users to understand their needs, working with suppliers to deliver innovative solutions, linking up with collaborators, research centres, even competitors to build and operate innovative systems. There are many advantages of pushing the social networking into innovation development process, i.e., collective efficiency, collective learning, bringing new insights and supporting shared experimentation, collective risk taking, development of different relationships to build across firms' frontiers [14].

Baldwin et al. reports that users do not anticipate selling goods or services based on their innovations, although they may later go into business as users-manufacturers [4]. Users-innovators are motivated to design new IT products and services, because they believe that new designs can enhance the things they do and in the way, which is the most appropriate for them. Prior to the advances in IT (e.g. online communities, wikis, user-generated content websites) users were bound by physical limitations for their social environments. Content in a digital form allows users to modify, share, use and reuse information, regardless of the creators' original purpose [7]. Websites such as Twitter, Facebook, MetaCafe, Wikipedia, Flickr have all been introduced within the last decade and rapidly grow in user membership. Organizations are beginning to invest in development of social media, to capitalize on a growing user population that is interested in creating, retrieving and exploring the Websites. Lindermann et al. have noticed that using Web 2.0 applications within SMEs implies consequently breaking down innovation process to employees level [10]. In daily business practice, Web 2.0 (e.g., wikis and blogs) has been observed as primarily being restricted to communication with the user and internal information and knowledge management.

Social network services enable people to connect online based on shared interests, hobbies or causes. Social networking inside an enterprise is valuable when the organization rewards individual effort but needs to encourage knowledge sharing and connection with others-across geographical or functional boundaries [8]. Social networking permits a social presence online that is the degree to which a medium allows the user to feel socially present in a situation that is mediated via technology [13]. Aggarwal argues that social networks provide rich content-based knowledge, which can be exploited for data mining purposes [1]. Schuster et al. add that social networking offers users autonomy in a unique way. Users can be independent from computer scientists, engineers and web designers to have a presence on the Internet. Sometimes, people overvalue the Internet information and hence integrate the pieces of information into their decision making process [17]. On the other side, people carefully select information provided on social network platforms. Currently social network sites are adopted in organizations for recruiting, advertising and internal collaboration [16].

III. SURVEY METHODOLOGY AND RESULTS

The literature studies create very optimistic view on user involvement in BIS development. Particularly, proponents of Web 2.0 strongly support the thesis that users are involved and participate or create the BIS by themselves and for themselves. This very optimistic view should be verified, therefore that empirical research was done in 2011 in Polish micro, SMEs and big companies. The research covered interviews with information technology (IT) personnel (i.e., CIOs) responsible for contact with end-users. The respondents were gathered from 270 firms. Characteristics of surveyed firms are presented in Table I.

Involvement of the end users in IT projects covering IS development is presented in Table II. The following activities of users have been specified: goal specification and project concepts (GSPC), business logic analysis and business process modelling (BLA BPM), requirements engineering (RE), information system design (ISD), information system implementation (ISI), information system testing (IST), information system installation and migration to a new IT environment (ISE), information system maintenance (ISM), security of information system (SIS), information system usage (ISU).

TABLE I. SURVEYED COMPANIES FEATURES

Feature	N=270
Number of employees	
Micro Enterprises (1-9 employees)	44,4%
Small Enterprises (10-49 employees)	29,3%
Medium Enterprises (50-250 employees)	15,2 %
Big Companies (more than 250 employees)	11,1 %
Dominating Activities	
Production	9,3%
Commerce	22,6%
Services	50,4%
Mixture of above activities	17,8%
Main Clients	
Individual	61,1%
Institutional	38,9%
Scope of Activities	
Local market	27,8%
Regional market	23,7%
National market	35,6%
International market	7,4%
Global market	5,6%

In the Table II six different profiles of users has been included. Passive users and users-evaluators are oriented towards the observation and acceptance of other people efforts. Co-creator supports IT staff in business information system development works. User as the partner plays equally important role as IT professional in the system development process. User as the producer is self-dependent and has got sufficient competencies to utilize IT independently of the IT staff help. The last, i.e., prosumers are able to utilize IT by themselves and for their work purposes. In this paper the definition of prosumption was adapted from the work of Xie et al. [18].

TABLE II. PARTICIPATION OF USERS IN IT PROJECTS

	User					
	Passive	***Evaluator***	***Co-creator***	***Partner***	***Producer***	***Prosumer***
GS PC	15%	17%	**33%**	24%	10%	1%
BLA BPM	**32%**	22%	19%	20%	5%	1%
RE	**37%**	19%	17%	16%	10%	1%
ISD	**34%**	20%	19%	15%	10%	1%
ISI	**39%**	16%	19%	15%	9%	1%
IST	18%	20%	**24%**	21%	15%	2%
ISE	22%	**27%**	24%	14%	10%	2%
ISM	20%	23%	**24%**	19%	12%	3%
SIS	**33%**	18%	20%	14%	13%	1%
ISU	7%	23%	20%	**30%**	15%	4%

Taking into account the results included in Table II you can notice that users are rather inactive. CIOs evaluate users as inactive at business analysis and business process modelling stages as well as at requirements engineering, system design and implementation. IT people do not demand the technical expertise from users, they should be helpful at the initial stages of business information system development process. Users were evaluated as co-creator in project concepts specification, information system testing and maintenance. Security of IS is the domain of IT professionals, and of course the strong activity of users is revealed at the business information system exploitation stage.

IV. USER INVOLVEMENT EVALUATION

Further analyses were realized for each of 4 groups of companies separately. In micro and small companies end-users are assumed to have direct, face-to-face (F2F) contact with IT staff, therefore they know more about requirements of each individual. In medium and big companies, the contact between the end-user and IT personnel is indirect and online, occasional, therefore the procedures of registration of user needs are implemented, and the end-user has no chance to be personally involved in the BIS development process.

The first question concerns the expected benefits and potential impediments. At big companies, over 70% of respondents admit that the most important benefits of end-user involvement in BIS development process cover better understanding of end-user requirements (83.3%), reduction costs of research and development works (73.3%), opportunities for market offer differentiation (73.3%), and improvement of company image (over 70%). Similarly at medium companies, over 70% of respondents argue that the most important benefits of end-user involvement in BIS development process comprise better understanding of end-user requirements (80.5%), development of strong relationships with user (80.5%), reduction of the cost of knowledge acquisition (75.6%) and improvement of company image (75.6%).

At small companies, the most important benefits of end-user involvement in BIS development include: supporting user education (72.2), better understanding of user requirements (72.2), taking better market position (67.1%). At micro companies, the most important benefits of end-user involvement in BIS development cover: better understanding of the user requirements (over 83%), moving to the better market position (over 71%), and improvement of company image (70%).

The most important impediments of end-user involvement in BIS development process comprise:

- At micro companies, end-user lack of knowledge and skills (73% of all respondents mention that), inevitability to learn new technologies (66%), lack of incentives and encouragement from the BIS producer (65%),.
- At small companies, necessity to learn new technologies (72% of all respondents state it), end-user lack of knowledge and skills (71%), and end-user lack of incentives provided by BIS producers (62%).
- At medium companies, end-user lack of knowledge and skills (according to 80% of respondents), the end-user necessity to learn new technologies (68%), and threat of theft of end-user ideas (61%).
- At big companies, necessity to know new technologies (80%), end-user lack of knowledge and skills (73%), and lack of incentives provided by BIS producers (57%).

The next question in this survey concerns methods of activation of end-user to encourage them to the cooperation for BIS development. So, in the survey the following methods have been identified:

- At big companies, participation of end-user in training courses and workshops (90% of respondents emphasize that), constant discussions of IT personnel with end-users (86.7%) and participation of end-user in reviewing processes covering interfaces reviews and use case analyses (77%).
- At medium companies, participation of end-users in BIS testing (indicated by 90% of respondents), constant discussions of IT staff with end-users (85%), and participation of end-users in quality management team work (76%).
- At small companies, participation of end-users in training courses and workshops (75% of respondents answered that), constant discussions of IT personnel with end-users (72%), and occasional interviews and meetings with end-users (72%).
- At micro companies, interviews and meetings of IT staff with end-users (75%), participation of end-users in courses and workshops (73%), participation of end-users in quality management team works (71% of respondents), and distribution of free and open source software (70%).

None of the respondent groups emphasizes agile methods application for software development or for project management. IT personnel and end-users are observed as conservatively minded persons. Similarly, the corporate architecture model discussions as well as IT product customisation opportunities have not be perceived as valuable for end-user encouragement. Beyond that, end-users are not interested in control and evaluation of BIS administrator works.

The fourth question concerns the knowledge from end-users demanded by IT staff. For micro and small companies knowledge on personal computer construction and usabilities and end-user tasks are the most important characteristics, although for medium and big companies, knowing business processes is enhanced (Table III).

TABLE III. END-USER KNOWLEDGE DEMANDED BY IT STAFF

User Knowledge	**Acceptance [%] at Companies:**			
	Micro	***Small***	***Medium***	***Big***
Computer usabilities	**50,8**	**57.0**	**58.5**	60.0
User tasks	**50.8**	**57.0**	51.2	**76.7**
Business processes	30.8	45.6	**61.0**	**66.7**
BIS technology	30.8	30.4	41.5	13.3
BIS interface technology	19.2	20.3	29.3	3.3
Software engineering	14.2	12.7	19.5	13.3

The next question focuses on end-user involvement in the works on BIS development. The IT staff considered separately end-users' involvement and engagement of online communities. In this research, the following activities have been analysed: basic research works, industry research works, development works, pilot implementations and product exploitations. Generally, the presence of end-users in BIS development process was accepted. However, the level of acceptance was different for different size companies:

- For big companies, end-users and online community were considered as required but not necessary for BIS development, and only in some cases the activities of community of users were treated as useless for high quality of BIS.
- At medium and small companies, involvement of end-users was perceived as demanded and necessary to ensure the high quality of BIS, however activities of online community were less

ISBN: 978-1-61208-226-4

important, and even in some cases, the involvement of end-users and community of users was considered as impediments for high quality of BISs.

- At micro companies, end-user involvement in BIS development works was accepted as necessary for high quality of IT products, but online communities were treated as neutral for BIS development.

The sixth question concerns the usability of virtual communities as well as social media for BIS development. The controversial opinion results are presented in the Tables IV-VII. The IT staff representatives were asked if the virtual communities and new media are important for high quality of business information systems. Their opinions were distinguished as:

- Compulsory (C): the Internet solution is necessary to ensure a high quality of BIS.
- Required (R): the proposed solution seems to be required, but not so strong demanded as above.
- Neutral (N): the solution is indifferent to the BIS development and without impact on it.
- Useless (U): the Internet solution is superfluous for BIS development.
- Impediment (I): the solution is harmful and detrimental for the BIS development (i.e., design, implementation and exploitation).

TABLE IV. VIRTUAL COMMUNITY AND SOCIAL MEDIA AT MICRO COMPANIES - IT PERSONNEL ATTITUDE

Media	**For high quality of BIS**				
	C	***R***	***N***	***U***	***I***
Newsletters	14	29	**42**	13	2
Company staff blogs	13	34	**42**	10	1
Users' blogs	9	29	**51**	10	1
Facebook	11	26	**52**	8	3
Twitter	8	24	**57**	9	2
ITproduct sale portals	9	32	**44**	13	2
IT product exploitation portals	9	35	**46**	9	1
Social networking	9	23	**52**	14	2
Chat room	6	18	**62**	13	1

Tables IV-VII include the percentage of positive responses in each of the 4 companies groups. The presented in Tables IV-VII information reverse a theory concerning very positive acceptance and necessity to develop virtual communities and social media implementing for BIS development. Mostly, the new media solutions are treated as required and neutral, but they are not necessary.

The IT staff is able to tolerate the mentioned in Tables IV-VII solutions i.e., newsletters, company staff blogs, users' blogs, Facebook and Twitter presence, IT product sale and exploitation portals, social networking and chat room, but they do not admit that the mechanisms are valuable for BIS implementation and exploitation.

TABLE V. VIRTUAL COMMUNITY AND SOCIAL MEDIA AT SMALL COMPANIES - IT PERSONNEL ATTITUDE

Media	**For high quality of BIS**				
	C	***R***	***N***	***U***	***I***
Newsletters	18.9	31.6	**37.9**	10.1	1.5
Company staff blogs	11.3	37.9	**43.0**	6.3	1.5
Users' blogs	13.9	32.9	**40.5**	8.9	3.8
Facebook	18.9	22.8	**35.4**	18.9	4.0
Twitter	16.5	16.5	**43.0**	21.5	2.5
ITproduct sale portals	13.9	**36.7**	29.1	16.5	3.8
IT product exploitation portals	17.7	**36.7**	30.4	15.2	0.0
Social networking	18.9	27.8	**31.6**	15.2	6.5
Chat room	12.7	29.1	**43.0**	10.1	5.1

TABLE VI. VIRTUAL COMMUNITY AND SOCIAL MEDIA AT MEDIUM COMPANIES - IT PERSONNEL ATTITUDE

Media	**For high quality of BIS**				
	C	***R***	***N***	***U***	***I***
Newsletters	14.6	31.7	**41.5**	12.2	0.0
Company staff blogs	12.2	**43.9**	36.6	7.3	0.0
Users' blogs	12.2	**41.5**	29.3	14.6	2.4
Facebook	12.2	21.9	**53.7**	12.2	0.0
Twitter	9.8	19.5	**60.9**	9.8	0.0
ITproduct sale portals	7.3	24.4	**51.2**	14.6	2.5
IT product exploitation portals	9.8	36.6	**39.0**	12.2	2.4
Social networking	9.8	29.3	**46.3**	12.2	2.4
Chat room	7.3	21.9	**53.7**	14.6	2.5

TABLE VII. VIRTUAL COMMUNITY AND SOCIAL MEDIA AT BIG COMPANIES - IT PERSONNEL ATTITUDE

Media	**For high quality of BIS**				
	C	***R***	***N***	***U***	***I***
Newsletters	13.3	**40.1**	40.0	3.2	3.4
Company staff blogs	13.3	20.0	**56.7**	6.7	3.3
Users' blogs	13.3	33.3	**43.4**	10.0	0.0
Facebook	16.7	16.7	**46.6**	16.7	3.3
Twitter	13.3	16.7	**50.0**	16.7	3.3
ITproduct sale portals	13.3	23.3	**46.7**	13.3	3.4
IT product exploitation portals	20.0	30.0	**33.3**	13.3	3.4
Social networking	20.0	26.7	**40.0**	13.3	0.0
Chat room	13.3	16.7	**56.7**	6.7	6.6

ISBN: 978-1-61208-226-4

The last question asked for the survey concerned attitudes of end-users towards traditional solutions implemented for their support, i.e., Customer Relationship Management (CRM) systems, insourced and outsourced Help Desk, IT service anticipation systems and providing consultancy by CIOs. The end-user support mechanisms are accepted as required. For big and medium companies, the respondents have argued that only insourced Help Desk is necessary, the other solutions are required and neutral. At small companies, the IT service anticipation systems were mostly preferred. For micro companies, respondents have no special preferences.

In this survey, social networking and IT service support mechanisms were presented from the IT staff point of view. They seem to be pragmatic and prefer traditional and verified solutions instead of strong acceptance of new media. They perceive new media as attractive but not necessary to support users and to involve them in BIS development.

V. Conclusion

Literature studies lead to the conclusion that business organizations are beginning to realize the potential benefits that can be captured when users and IT firms co-create values. Companies benefit from a large membership of users. They get benefits such as marketing insights, cost savings, brand awareness and idea generation. Users benefit from a positive experience that fulfils personal needs and interests. Experience is defined as an intensive individually involved event.

In the survey done in 270 firms these theses were verifies. So, IT professionals have been observed as very sceptical about utilisation of new media and social networking, although in Internet several positive examples are registered. The research revealed important problems of lack of knowledge and skills of end-users as well as a lack of incentives necessary for their deeper involvement in BIS development. Therefore a huge social capital is unused.

In the research, the quantitative methods are applied to reveal the influence of new media and social networking on information system development. The research does not provide an optimistic view to encourage for further development of new media and social networking. But the social networking tools' providers should not be disappointed, in particular cases recognized through a qualitative approach their tools can be recognized as well accepted. The future research works will cover cloud computing tools for end user support.

Acknowledgment

The research work is funded by the National Science Centre in Poland. The grant number is 4100/B/H03/2011/40.

References

[1] Ch.C. Aggarwal, "An Introduction to social network data analytics", in Social Network Data Analytics, Aggarwal Ch.C. (eds.), Springer, New York, 2011, pp.1-16.

[2] Alter S (2004) IT innovation through a work systems lens. IT innovation for adaptability and competitiveness, IFIP TC8/WG8.6. Fitzgerald B., Wynn E. (eds.) Kluwer Academic Publishers, Boston, pp.43-64.

[3] W.L. Anderson, W.T.Crocca, "Engineering practice and co-development of product prototypes", Communication of the ACM, 1993 36(6), pp. 49-56.

[4] C.Baldwin, Ch.Hienerth, E.von Hippel, "How user innovations become commercial products: a theoretical investigation and case study", Research Policy 35(9) 2006, pp. 1291-1313.

[5] H.Barki, J.Hartwick, "Rethinking the concept of user involvement, and user attitude", MIS Quarterly, 1994, 18(1), pp. 59-79.

[6] A.J.Campbell, R.G.Cooper, "Do customer partnerships improve new product success rates?" Industrial marketing management, 1999, 28(5), pp. 507-519.

[7] P.M. Di Gangi, M.Wasko, M. " The Co-Creation of Value; Exploring User Engagement in User-Generated Content Websites", Proceedings of JAIS Theory Development Workshop, Sprouts: Working Papers on Information Systems 9(50), 2009, http://sprouts.aisnet.org/9-50, accessible May 2012.

[8] M.G.Festinger, "Informal Social Communication", Psychological Review, 57(5), 1950, pp. 271-282.

[9] S.Justesen, Innoversity in Communities of practice, in Knowledge networks: Innovation through communities of practice, Kimble Ch., Hildreth P (eds) IGP. Hershey, 2004, pp.79-95.

[10] N.Lindermann, S.Valcarcel, M.Schaarschmidt, H.von Kortzfleisch, "SME2.0: Roadmap towards Web 2.0 - Based Open Innovation in SME-Networks - A Case study based researcg framework", in Dhillon G., Stahl B.C., Baskerville R (eds.) Information Systems - Creativity and Innovation in Small and Medium-Sized Enterprises, Springer Heidelberg, 2009, pp.28-42.

[11] C.R.Martin, D.A.Horne, "Level of success inputs for service innovations in the same firm", International Journal of Service Industry Management, 1995, 6 (4), pp. 40-56.

[12] J.Matthing, B.Sandem, B.Edvardsson, "New service development: learning from and with customers", International Journal of Service Industry Management, 2004, 15(5), pp. 479-498.]

[13] H.Rheingold, The Virtual Community, Homesteading on the Electronic Frontier, Reading PA Addison-Wesley, 1993.

[14] J.Tidd, J.Bessant, K.Pavitt, Managing innovation, Integrating technological, market and organizational change, 3rd edition, J.Wiley & Sons, Chichester, 2005.

[15] J.Tidd, J.Bessant, Managing innovations, integrating technological, market and organizational change. John Wiley and Sons, Chichester, 2009.

[16] D. Sandy Staples, "Web 2.0 Social Networking Sites", in Social Web Evolution: Integrating Semantic applications, and Web 2.0 technologies, Lytras M.D., Ordonez de Pablos P. (eds.), IGI Global. Hershey, 2009, pp.57-76.

[17] J.Schuster, Y. Su Lee, D.Kabothanassi, M. Bargel, M.Geierhos, "SCM - A Simple. Modular and Flexible Customer Interaction Management System", in International Conference on Information Society (i-Society 2011) Proceedings, IEEE UK/RI Computer Chapter, London, pp.169-175, http://www. i-society.eu, accessible May 2012.

[18] Ch.Xie, R.P. Bagozzi, S.V.Troye, "Trying to prosume: toward a theory of consumers as co-creators of value", Journal of the Acad.Mark.Sci (2008) 36: pp.109-122.

A New Approach To Solve Aircraft Recovery Problem

Congcong Wu
The Scientific Research Academy
Shanghai Maritime University
Shanghai 200135, P. R. China
meilongle@hotmail.com

Meilong Le
The Scientific Research Academy
Shanghai Maritime University
Shanghai 200135, P. R. China
lemeilong@126.com

***Abstract*—When disruptions occur, the airlines have to recover from the disrupted schedule. The recovery usually consists of aircraft recovery, crew recovery and passengers' recovery. This paper focuses on aircraft recovery. Take the total cost of assignment, cancelation and delay as an objective; we present a more practical model, in which the maintenance and other regulations are considered. Then, we present a so-called iterative tree growing with node combination method. By aggregating nodes, the possibility of routings is greatly simplified. So, it can give out the solution in more reasonable time. Finally, we use data from a main Chinese airline to test the solution algorithm. The experimental results state that this method could be used in aircraft recovery problem.**

Keywords-aircraft recovery; airlines optimal recovery ; airlines recovery; recovery algorithm

I. Introduction

When disruptions caused by severe weather conditions, air traffic control or mechanical failures occur, the airlines have to recover from the disrupted schedule. The airlines recovery usually consists of aircraft recovery, crew recovery and passengers' recovery. Since the aircraft is viewed as the most important scarce resource, the most work on operational recovery problems has been reported on the aircraft recovery.

Aircraft recovery problem (ARP) is to determine new flight departure times, cancellations and rerouting for affected aircrafts including ferrying, diverting, swapping and so on. Besides that several decision rules, such as, aircrafts balance requirements, maintenance requirements and station departure curfew restrictions should also be considered. At the end of the recovery period, aircrafts should be positioned to resume operations as planned.

Being different to the aircraft rotation problem in the planning stage, the method to solve the ARP should calculate the problem in reasonable time, which is very difficult to most optimization solvers under most reasonable disruption scenarios. How to solve the ARP in reasonable time and meet these decision rules has been one of the most important keys in airline recovery study. Teodorvic and Gubernic (1984) [1] are one of the first to study the aircraft recovery problem, using a branch and bound (B&B) algorithm [2] to solve the aircraft recovery model (ARM) , but the research does not satisfy the constraints of station curfews, maintenance requirements and aircrafts balance at the recovery period in the modeling. Arguello et al. (1997) [3] creates a greedy randomized adaptive search procedure (GRASP) to reconstruct aircraft routings, which is a fast heuristic based on randomized neighborhood search, but they don't consider the maintenance requirements and crew requirements after the aircraft routings altered. Afterwards, Bard et al. (2001) [4] develops a time-band optimization model to reconstruct cost-effective aircraft routings. The disadvantage is that the research excludes the maintenance requirements and crew requirements. Thengvall (2003) [5] presents a bundle algorithm to solve a multi-commodity network model. As in Petersen et al. (2010) [6], they integrate all kinds of recovery simultaneously, and employ the Bender's decomposition to decompose the model into a master problem (airline schedule recovery) and three sub-problems (aircraft recovery, crew recovery and passenger recovery), using an optimization-based approach to solve the situation of hub closure.

In our paper, modeling is based on flight strings instead of flights as well as defining recovery scope, in order to solve the model in reasonable time. We assign specific aircraft to flight strings while meeting maintenance requirements, station departure curfew restrictions and other aircraft requirements. As to the solution methodology, firstly, we transform our model into time-space network. Then, we create a new method (a so-called iterative tree growing with node combination method) to solve the network model, which is the most important part of our paper. We test our intelligent method with data from Chinese airlines. Computational results are presented for a daily schedule recovery, showing that the proposed approach provides faster times to optimality in some cases and always obtains feasible, near-optimal solutions for medium-size airlines recovery problem much more quickly than can be found using CPLEX. In our future study, we should do much more experience to test our new method, try it on the large-size airlines recovery problem and use it much more widely, for example, in integrated recovery combining with crew recovery or passenger recovery or all of the three.

The reminder of the paper is organized as follows. We first give in Section II a literature review of the aircraft recovery problem. In Section III, we build our aircraft recovery model. The solution methodology is described in Section IV and two scenarios are presented to test the intelligent method. We give our conclusion in Section V.

II. Literature Review

When one or more aircrafts are out of service, the airlines have to operate the flight schedule with a reduced number of planes. Teodorvic and Gubernic (1984) [1] are the pioneers to study ARP. The paper tries to minimize total the passenger delay by swapping or delaying flights and solved exactly by branch and bound. Subsequently, Teodorovic and Stojkovic (1990) [7] formulates a heuristic algorithm to solve the same problem as Teodorvic and Gubernic (1984) [1]. But, in their paper the chief objective is to minimize the total passenger delay with an equal total number of cancelled flights. In addition, neither of these models considers flight delay and cancelation cost.

Yan and Yang (1996) [8] are the first to allow for delays and cancelations simultaneously. Four systematic strategic models are developed by perturbing the BSPM (basic schedule perturbation model) and combining various scheduling rules. The BSPM is designed to minimize the schedule-perturbed period after an incident and to obtain the most profitable schedule given the schedule-perturbed period. These network models are formulated as pure network flow problems or network flow problems with side constraints. With real flight data from Taiwan Airlines, the former was solved by the network simplex method while the latter was solved by Lagrangian relaxation with subgradient methods. However, the constraints of aircraft maintenance and crew scheduling are overlooked.

An extension to the network model of Arguello et al. (1997) [9] is presented by Thengvall et al. (2000) [10]. The authors presents a model in which they penalize in the objective function the deviation from the original schedule and they allow human planners to specify preferences related to the recovery operations.

Rosenberger et al. (2003) [11] models ARP as a set-packing problem with a time window and slots restrictions. In this model the objective is to minimize the cost of assigning routes to aircraft and the cost of cancelling the unassigned legs. Being different from Arguello et al. (1997) [3] and Bard et al. (2001) [4], their paper assumes an aircraft selection heuristic (ASH) for ARO (an optimization model for aircraft recovery), which selects a subset of aircraft for optimization prior to generating new routes. Compared with network model, this model can check maintenance feasibility using column generation.

Eggenberg et al. (2007) [13] introduces an extension of the time-space network model to minimize delays, cancellations and plane swappings, and make span cost.

In Massound Bazargan (2010) [14], the paper introduces the airline irregular operation in detail and uses the time-band optimization method to solve the aircraft recovery as an example.

Le et al. (2011) [15] provides an overview recent years' of disruption management of schedule, aircraft, crew, passenger and the integrated recovery.

Something is done in our aircraft recovery model, aiming to minimize the aggregate cost comprised of assigning cost and recovering cost. We transform the aircraft recovery problem as a multi-commodity network with side constraints and using a so-called iterative tree growing with node combination method to solve the disruption.

III. The Aircraft Recovery Problem

A. Sets

F_n set of all flight legs in recovery scope N

$F_n^{mandatory}$ set of mandatory flight legs

$F_n^{optional}$ set of optional flight legs that are candidates for deletion

E_n set of fleet types in recovery scope N

S_n set of flight string s in recovery scope N

$K(e)_n$ set of aircraft of fleet type in recovery scope N

$H(e)_n$ set of aircraft of fleet type requiring maintenance within T in recovery scope N

A set of airports

$A^{ma\text{int}}(e)$ set of stations that are capable of performing schedule maintenance of aircraft of fleet type e

G^k set of ground arcs of aircraft k which cross the count time

B. Datas

y_j^k a ground variable used to count the number of aircraft k on the ground j

AN_e the number of aircrafts in fleet type

$c_{e,s}^k$ cost of assigning aircraft $k \in K(e)_n$ to flight string s

td_f actual departure time of flight f

td_f actual departure time of flight f

ta_f actual arrival time of flight f

ta_f actual arrival time of flight f

$A_{e,f}^k$ ready time of aircraft k to operate flight f

CC_f cost of canceling flight f

CD_f cost of 1-min delay of flight f

DT_f expected trip (block-to-block) time of flight f

N recovery scope index

T_f the scheduled departure time of flight f

U minimum connection time

r_s^k the number of flight string s is being executed by aircraft k cross the count time

C. Variables

$$I_{m,s} = \begin{cases} 1 \ \ if \ \ an \ eligible \ ma\text{int}enance \ station \ m \in A^{ma\text{int}}(e) \\ \quad is \ visited \ by \ flight \ string \ s \\ 0 \ \ otherwise \end{cases}$$

$$a_{f,s} = \begin{cases} 1 \ indicator \ \ \text{var}iable, \ if \ \ flight \ f \in F \ \ in \ flight \ string \ s \\ 0 \ otherwise \end{cases}$$

$$x_{e,s}^k = \begin{cases} 1 \ \ if \ \ aircraft \ k \in K(e)_n \ \ \text{cov}ers \ flight \ string \ s \\ 0 \ \ otherwise \end{cases}$$

$$z_f = \begin{cases} 1 \ \ if \ \ flight \ f \ \ is \ canceled \\ 0 \ \ otherwise \end{cases}$$

$$p_j^k = \begin{cases} 1 \ \ ground \ \ arc \ \ j \in G^k \ \ for \ aircraft \ k \\ \quad crosses \ the \ count \ time \\ 0 \ \ otherwise \end{cases}$$

D. Mathematical formulation

$$\min \sum_{k \in K(e)_n} \sum_{s \in S_n} c_{e,s}^k x_{e,s}^k + \sum_{f \in F_n} CD_f (1 - z_f)[td_f - T_f] + \sum_{f \in F_n} CC_f z_f \quad (1)$$

Subject to:

$$\sum_{k \in K(e)_n} \sum_{s \in S_n} x_{e,s}^k a_{f,s} + z_f = 1, \forall f \in F_n \quad (2)$$

$$\sum_{k \in K(e)_n} \sum_{s \in S_n} x_{e,s}^k a_{f,s} = 1, \forall f \in F_n^{mandatory} \quad (3)$$

$$\sum_{k \in K(e)_n} \sum_{s \in S_n} x_{e,s}^k a_{f,s} \leq 1, \forall f \in F_n^{optional} \quad (4)$$

$$\sum_{s \in S_n} \sum_{m \in A^{ma\text{int}}(e)} I_{m,s} x_{e,s}^k = 1, \forall k \in K(e)_n \quad (5)$$

$$\begin{gathered} x_{e,s}^k a_{f,s} + \sum_{f'} x_{e,s'}^k a_{f',s'} \leq 1 \\ f \in first \quad flight \quad of \quad s \in S_n, \\ f' \in \{ first \quad flight \quad of \quad s' \in S_n \,|\, T_{f'} > T_f, \\ T_{last \ \ flight \ \ of \ \ s} + DT_{last \ \ flight \ \ of \ \ s} \geq \\ T_{f'} + Max \ \ Delayed \ \ allowed \}, \forall k \in K(e)_n \end{gathered} \quad (6)$$

$$\sum_{k \in K(e)_n} \sum_{s \in S_n} r_s^k x_{e,s}^k + \sum_{k \in K(e)_n} \sum_{j \in G^k} p_j^k y_j^k \leq N_e, \forall e \in E_n \quad (7)$$

$$x_{e,s}^k a_{f_{i+1},s} - x_{e,s}^k a_{f_i,s} = 0, \forall k \in K(e)_n, f_i \in s \in S_n \quad (8)$$

$$\begin{gathered} td_f \geq A_{e,f}^k x_{e,s}^k a_{f,s}, \forall f \in first \ \ flight \ \ of \ \ S \ \ , \\ k \in K(e)_n \end{gathered} \quad (9)$$

$$\begin{gathered} td_{f'} \geq ta_f \, x_{e,s}^k a_{f,s} x_{e,s'}^k a_{f',s'} + U \\ f \in last \ \ flight \ \ of \ \ string \ \ s \ \ \in S_n, \\ f' \in \{ first \ \ flight \ \ of \ \ string \ \ s' \ \ \in S_n \\ T_f + DT_f \leq T_{f'} + Max \ \ Delayed \ \ allowed \} \end{gathered} \quad (10)$$

$$td_{f_{i+1}} \geq ta_{f_i} + U, \forall f \in flight \ \ of \ \ string \ \ s \ \ \in S_n \quad (11)$$

$$td_f \geq T_f, \forall f \in F_n \quad (12)$$

$$ta_f = td_f + DT_f (1 - z_f), \forall f \in F_n \quad (13)$$

$$x_{e,s}^k \in \{0,1\} \quad (14)$$

$$z_f \in \{0,1\} \quad (15)$$

$$td_f, ta_f \geq 0 \quad (16)$$

The objective (1) tries to minimize the aggregate cost comprised of assigning strings (assignment cost) and recovering aircrafts (delay cost and cancellation cost) in the recovery scope. Either a flight must be contained in exactly one string or cancelled, as seen in (2). The cover constraints are split into (3) and (4) to distinguish between the mandatory and optional leg sets, ensure each aircraft is assigned to no more than one string. Maintenance cover constraints are seen in (5). This simply ensures a maintenance opportunity is built in, and the specific maintenance planning can be done post-optimization. Constraint (6) ensures that each available aircraft cannot be assigned to two different strings in the same time. The count constraint (7), make sure that the total number of aircraft in the air and on the ground does not exceed the size of fleet type e. Constraint (8) defines rotations aircraft usage. All flights in a rotation use one aircraft not different ones. By using the concept of rotation and defining rotations in the model, aircraft balance at each airport is satisfied. Constraints (9)–(12) determine the departure time of each flight. A flight cannot depart earlier than the ready time of its assigned aircraft, as stated in (9). Constraints in (10) ensure that when two flight strings are flown by the same aircraft, the second string cannot depart earlier than real arrival time of first string (because of the minimum connecting time). In a flight string, the departure time of a flight cannot be earlier than the arrival time of its previous flight, as stated in (11).

Constraints in (12) state that no flight is allowed to depart before its scheduled departure time. Constraints in (13) relate the departure and arrival times for each flight. Constraints (14)–(16) ensure that the x, z are binary variables.

IV. SOLUTION METHODOLOGY

Even by limiting the scope of the problem to get computational result, to most airlines, the problem is likely too large and complex to return a globally optimal solution with optimization solver for most reasonable disruption scenarios. Thus, we seek the hybrid method, which is optimization method with heuristic approach. The heuristic we used is so-called iterative tree growing with node-combination. The time-space graph is used to describe our heuristic method. In the graph, the cities and times are represented horizontally and vertically respectively. Each node represents an airport-departure or airport-arrival event. All the arcs denote flights. Except first node (time-earliest node) and last node (time-termination node, usually night curfew time for departure), we draw all parallel arcs (copy arcs) if the arc lies above the node and originates from the same node (airport). As the flight arcs are placed in the graph iteratively, the tree grows downward. There are three kinds of arcs in the graph. One is original flight arcs. The other is copy arcs, which is actually opportunity flight arcs due to flight cancellation. The third is overfly arcs, which is actually delay flight arcs. Most probably, each copy arc generates a new node. From this new node, copy arcs and overfly arcs can be originated or connected again. It is an iterative procedure. By so stretching, the tree grows downward.

Obviously, there will be more and more nodes and arcs as the graph stretches downward. Every route from top to down represents a routing. In order to simplify such a combinatorial problem, we use circle to replace a dot to represent a node. In other words, a node does not represent a single airport-departure or airport-arrival event, but a cluster of airport-departure or airport-arrival events. All the airport-departure or airport-arrival dots within the certain time circle are aggregated to this node. The delay time is counted from departure circle node to arrival circle node, not the difference between real departure and arrival time. Under the extreme condition, such aggregating method may calculate delay time the whole circle diameter difference.

The test instances used as benchmark problems in this study are acquired from real flight schedule of one medium-size airlines in China. The schedule consists of 170 flights served by 5 fleets, 35 aircrafts over a network of 51 airports all over the country.

We choose test instances from the flight schedule. The relative data is listed in TABLE I. The computations also use the following assumptions:

- Each station requires a minimum of 40 minutes turnaround time;
- Execute midnight arrival/departure curfew (no arrival or departure after midnight is allowed);
- Each minute of delay on any flight costs the airline $20.

TABLE I THE FLIGHT SCHEDULE AND CANCELATION COST

Fleet type	Flight string	Aircraft	Flight	Pax	DStat	STD1	AStat	STA1	Duration	Cancelation cost
737-800	S1	1	9131	100	SHA	815	TSN	1005	1:50	$17,490
			9125	72	TSN	1100	SZX	1350	2:50	$15,780
			9126	100	SZX	1450	TSN	1755	3:05	$21,050
			9132	100	TSN	1855	SHA	2040	1:45	$16,980
737-800	S2	2	9380	14	SZX	845	SHA	1050	2:05	$14,120
			9371	14	SHA	1305	SZX	1515	2:10	$14,870
			9372	49	SZX	1610	SHA	1820	2:10	$17,120
			9369	150	SHA	1910	SZX	2115	2:05	$19,870
737-800	S3	3	9304	104	CAN	1130	SHA	1335	2:05	$18,740
			9375	78	SHA	1435	SZX	1645	2:10	$17,290
			9376	78	SZX	1750	SHA	2015	2:25	$18,110
			9303	78	SHA	2100	CAN	2315	2:15	$17,890

We use two scenarios to test the method.

Scenario 1—Delay

The aircraft 2 in airport SZX must be grounded at 8:00 and is available until 15:00. That is, aircraft 2 is unavailable from 8:00 to 15:00. The trivial solution 1 is to cancel flights 9380 and 9371 which are flown by aircraft 2 during 8:00 to 15:00. The total cancellation cost is $28,990. The trivial solution 2 is to delay flight string 2(9380, 9371, 9372, 9369). The ready time of flight 9369 is 23:25. Against the curfew, so the flight 9369 should be canceled. The solution got from our method is listed in TABLE II. The total cost is $38.270.

Scenario 2—Delay and grounded combination

In this case, we assume that aircraft 1 in airport SHA becomes grounded owing to some mechanical failure at 8:00 and is unavailable for the rest of the day. The obvious solution without permitting any rerouting of other aircraft is to cancel flights 9131, 9125, 9126 and 9132. These cancellations cost the airline a total of $71,300 (the sum of all cancellation costs for flight string1).

TABLE II TRIVAL OPTION 2

Aircraft tail	Flight	Pax	DStat	STD1	AStat	STA1	Option	Cancelation cost	Delay cost
1	9131	100	SHA	815	TSN	1005	/	/	/
	9125	72	TSN	1100	SZX	1350	/	/	/
	9126	100	SZX	1450	TSN	1755	/	/	/
	9132	100	TSN	1855	SHA	2120	/	/	/
2	9380	14	SZX	1500	SHA	1745	Delay	/	$7,500
	9371	14	SHA	1745	SZX	2035	Delay	/	$5,600
	9372	49	SZX	2035	SHA	2325	Delay	/	$5,300
	9369	150	SHA	1910	SZX	2115	Cancel	$19,870	/
3	9304	104	CAN	1130	SHA	1335	/	/	/
	9375	78	SHA	1435	SZX	1645	/	/	/
	9376	78	SZX	1750	SHA	2015	/	/	/
	9303	78	SHA	2100	CAN	2315	/	/	/
Total								$19,870	$18,400

The according graph is drawn in Figure 1. The figure on the arc is flight number. The figure besides the node is departure or arrival time. The node is marked according to vertical time coordinate and horizontal airport coordinate. In order to reflect whether two flight legs can be connected, the arrival time has been added turnaround time. For example, flight 9131 arrives in TSN at 10:05 and connects to flight 9125 which is available for departure at 10:45. We use 30 minutes as the diameter of the circle. So, flight 9131 is not ready for departure at 10:45 but 11:00.

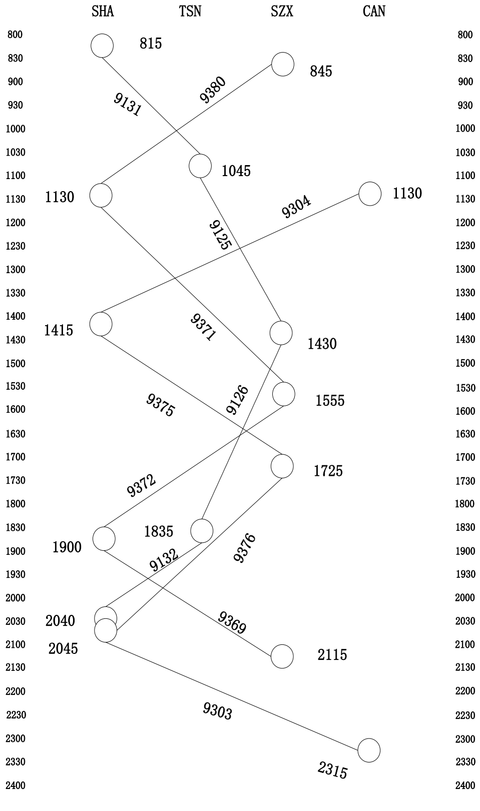


Figure 1 The flights graph

In the Figure 1, one arc is drawn from node 2 to node 16, it represents flight 9131. Actually it is a copy arc representing flight 9131, the delay time is 210 minutes, not the actual time minus the schedule time. This is because flight 9131 was scheduled to leave SHA at 8:15. If this flight occurs in node 2, the departure time is calculated as 11:30. Considering the nodes are within 30 minutes circle, this delay spans from 8:00 to 11:30, a total of 210 minutes. Each minute of delay costs the airline $20, so flight 9131 has a delay cost of $4,200 if it departs from node 2.

Figure 2 is deduced form Figure 1 based on the method mentioned above.

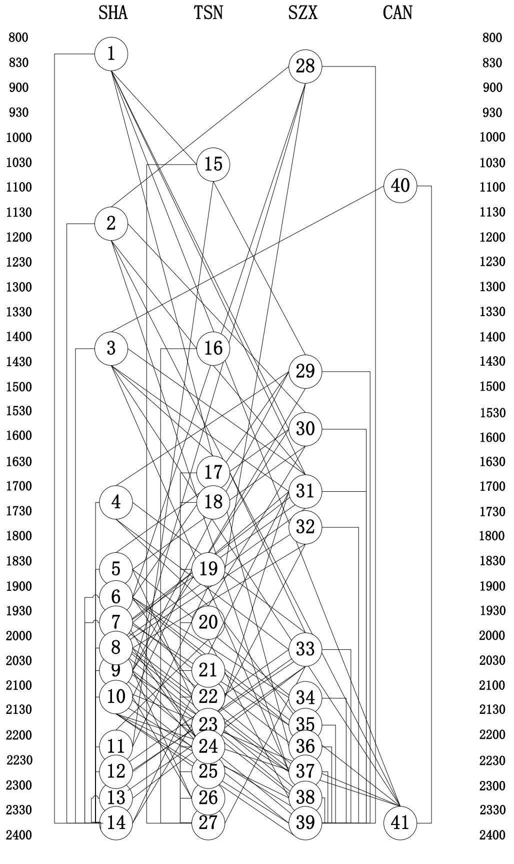


Figure 2 The stretch graph

TABLE III presents the non-zero delay costs for all flight arcs in Figure 2.

TABLE III NONE-ZERO DELAYCOSTS

Flight	Pax	Ori.node	Dest.node	Delay cost($)
9125	25	19	36	9000
9125	25	16	32	3600
9125	25	20	38	10200
9125	25	17	33	6600
9126	100	33	27	6600
9126	100	32	22	3600
9126	100	30	20	1200
9126	100	31	21	3000
9126	100	31	22	3000
9131	100	4	20	10800
9131	100	2	16	4200
9131	100	5	22	12600
9131	100	3	17	7200
9131	100	6	23	13200
9131	100	7	24	13800
9131	100	8	25	14400
9131	100	9	26	15000
9131	100	10	26	15600
9132	100	20	11	1200
9132	100	20	36	1200
9132	100	23	14	3600
9132	100	22	13	3000
9369	150	7	37	600
9369	150	8	37	1200
9369	150	8	38	1200
9369	150	9	39	1800
9369	150	10	38	2400
9371	14	4	33	4800
9371	14	5	35	6600
9371	14	6	35	7200
9371	14	3	31	1200
9371	14	7	37	7800
9371	14	8	38	8400
9371	14	9	39	9600
9371	14	10	39	9900
9372	49	33	12	4800
9372	49	32	8	1800
9372	49	33	13	4800
9372	49	31	7	1200
9372	49	31	8	1200
9375	78	4	33	3000
9375	78	5	35	4800
9375	78	6	35	5400
9375	78	7	37	6000
9375	78	8	38	6600
9375	78	9	39	7200
9375	78	10	39	7800
9376	78	33	13	3000
9380	14	29	4	7200
9380	14	33	12	13800
9380	14	32	8	10800
9380	14	30	5	8400
9380	14	33	13	13800
9380	14	31	7	10200
9380	14	31	8	10200

TABLE IV RECOVERY SOLUTION FOR SCENARIO 1

Aircraft tail	Flight	Pax	Dstat	Ori.node	Astat	Dest.node	Option	Cancelation cost	Delay cost
1	9131	100	SHA	1	TSN	15	/		
	9125	72	TSN	15	SZX	29	/		
	9380	14	SZX	29	SHA	4	Delay		$7,200
	9375	78	SHA	4	SZX	33	Delay		$3,000
	9376	78	SZX	33	SHA	13	Delay		$3,000
2	9126	100	SZX	30	TSN	19	Delay		$600
	9132	100	TSN	19	SHA	10	/		
	9303	78	SHA	10	CAN	39	Delay		0
3	9304	104	CAN	40	SHA	3	/		
	9371	14	SHA	3	SZX	31	Delay		$1,200
	9372	49	SZX	31	SHA	7	Delay		$1,200
	9369	150	SHA	7	SZX	37	Delay		$600
Total									$16,800

Through a series of aircraft rerouting and cancellations in an effort to minimize the total cost to the airlines, the total cost of the solution is $16,800, less than the above two trivial options. In contrast with the method of branch and bound (B&B), using an intelligent algorithm we can quickly obtain feasible, near-optimal solutions faster times in some case study than using CPLEX (Thinkpad X201S). Through the computation results we can see our model has quite a good effect on aircraft recovery optimization. The total passenger delay is 40825 minutes. The total cost for this actual flight schedule is $16,800, which is similar to the solution got from our method.

Using the so-called iterative tree growing with node combination method as scenario1， we can get the recovery solution for scenario2.

TABLE V RECOVERY SOLUTION FOR SCENARIO 2

Aircraft	Flight	Pax	DStat	Ori.node	AStat	Dest.node	Option	Cancelation cost	Delay cost
1	9126	100	SZX	29	TSN	19	Cancel	$21,050	/
	9132	100	TSN	19	SHA	10	Cancel	$16,980	/
	9371	14	SHA	2	SZX	30	Cancel	$14,870	/
2	9380	14	SZX	28	SHA	2	/	/	/
	9131	100	SHA	2	TSN	16	Delay	/	$4,200
	9125	72	TSN	16	SZX	32	Delay	/	$3,600
	9372	49	SZX	32	SHA	8	Delay	/	$1,800
	9369	150	SHA	8	SZX	38	Delay	/	$1,200
3	9304	104	CAN	40	SHA	3	/	/	/
	9375	78	SHA	3	SZX	31	/	/	/
	9376	78	SZX	31	SHA	9	/	/	/
	9303	78	SHA	9	CAN	41	/	/	/
Total								$52,900	$10,800

The total cost for scenario 2 by our method is $63,700, smaller than the trivial solution of $71,300 resulting from canceling all flights operated by aircraft 1. The total cost for this actual flight schedule is $63,400, which is similar to the solution got from our method. In this scenario we can see the aircraft is one of the most important resources in airlines recovery. The shortage of aircraft resources limited the degree of airlines recovery.

V. CONCLUSION

The paper presents a more practical formulation for airline optimal recovery. In order to get the solution in a reasonable time, a new approach to solve the problem is studied. The computational results state the method could be used in airline recovery. On average, for medium-size airline recovery, the algorithm finds a feasible solution twice as fast as an exact algorithm, obtaining a high-quality feasible solution in half the time is an important improvement for our application. Often in our method, having several near-optimal solutions provide decision makers much more flexibility.

Airlines recovery is a more complex and large-scale problem. Not only should aircrafts be considered, but crew and passengers should be considered, too. In the future, a more comprehensive recovery model should be studied. Meanwhile, a more systematic evaluation of the method should be carried out.

ACKNOWLEDGMENT

The work was partially supported by research grants from Shanghai Municipal Natural Science Foundation [No. 10190502500], Shanghai Maritime University Start-up Funds, Shanghai Science & Technology Commission Project [No. 09DZ2250400] and Shanghai Education Commission Project [No. J50604].

REFERENCES

[1] D.Teodorvic and S.Gubernic. Optimal dispatching strategy on an airline network after a schedule perturbation, European Journal of Operations Research , 1984, 15: 178_182.

[2] S P. Bradley, A. Hax and T L. Magnanti. Applied Mathematical Programming. Addison-Wesley Publishing Company, Reading, Massachusetts,1977.

[3] M F.Arguello and J F.Bard. A GRASP for Aircraft Routing in Response to Groundings and Delays, Journal of Combinatorial Optimization 5, 1997, 211‑228.

[4] J F. Bard, G.Yu and F Michael. Optimizing aircraft routings in response to groundings and delays. IIE Transactions, 2001, 33:931-947.

[5] J M. Rosenberger, E L.Johnson and G L.Nemhauser. Rerouting Aircraft for Airline Recovery, Transportation Science , 2003, Vol. 37, No. 4, pp. 408‑421.

[6] Johnson D. Petersen, Gustaf Solveling, Ellis L. Johnson, et al. An Optimization Approach to Airline Integrated Recovery[J]. 2010.

[7] D.Teodorovic and G. Stojkovic. Model for operational daily airline scheduling. Transportation Planning and Technology 14, 1990, 273-285.

[8] S.Yan and D. Yang. A decision support framework for handling schedule perturbations. Transportation Research. Part B:Methodology, 1996, 30(6), 405-419.

[9] M F.Arguello and J F.Bard. A GRASP for Aircraft Routing in Response to Groundings and Delays, Journal of Combinatorial Optimization 5, 1997, 211‑228.

[10] M F. Arguello, J F.Bard and G.Yu. Framework for exact solutions and heuristics for approximate solutions to airlines ’ irregular operations control aircraft routing problem[D]. Department of Mechanical Engineering, 1997, University of Texas, Austin.

[11] B.Thengvall, J F.Bard and G.Yu. Balancing user preferences for aircraft schedule recovery during irregular operations. IIE Transactions 2000; 32:181-93.

[12] B G. Thengvall, J F.Bard and G. Yu. A bundle algorithm approach for the aircraft schedule recovery problem during hub closures, Transportation Science, 2003, Vol.37, No.4.

[13] N.Eggenberg, M.Bierlaire and M.Salani. A column generation algorithm for disrupted airline schedules. Technical report, 2007, Ecole Polytechnique Federale de Lausann.

[14] Massoud Bazargan. Airline Operations and Scheduling(2nd edition)[M], 2010.

[15] M L.Le and C C. Wu et al. Airline recovery optimization research: 30 year’s march of mathematical programming—a classification and literature review. Proceeding of 2011 International Conference on Transportation and Mechanical & Electrical Engineering, IEEE Catalog Number: CFP1120R-CDR ISBN: 978-1-4577-1699-7, 161-165, 2011.

A Semi-supervised Approach for Industrial Workflow Recognition

Eftychios E. Protopapadakis, Anastasios D. Doulamis,
Konstantinos Makantasis
Computer Vision and Decision Support Lab.
Technical University of Crete
Chania, Greece
eft.protopapadakis@gmail.com
adoulam@ergasya.tuc.gr
konst.makantasis@gmail.com

Athanasios S. Voulodimos
Distributed Knowledge and Media Systems Group
National Technical University of Athens
Athens, Greece
thanosv@mail.ntua.gr

Abstract—**In this paper, we propose a neural network based scheme for performing semi-supervised job classification, based on video data taken from Nissan factory. The procedure is based on (a) a nonlinear classifier, formed using an island genetic algorithm, (b) a similarity-based classifier, and (c) a decision mechanism that utilizes the classifiers' outputs in a semi-supervised way, minimizing the expert's interventions. Such methodology will support the visual supervision of industrial environments by providing essential information to the supervisors and supporting their job.**

Keywords-semi-supervised learning; activity recognition; pattern classification; industrial environments.

I. Introduction

Visual supervision is an important task within complex industrial environments; it has to provide a quick and precise detection of the production and assembly processes. When it comes to smart monitoring of large-scale enterprises or factories, the importance of behavior recognition relates to the safety and security of the staff, to the reduction of bad quality products cost, to production scheduling, as well as, to the quality of the production process.

In most current approaches, the goal is either to detect activities, which may deviate from the norm, or to classify some isolated activities [1],[2]. Modern techniques are based on supervised training using large data sets. The need of a significant amount of labeled data during the training phase makes classifiers data expensive. In addition, that data demands an expert's knowledge that increases further the cost.

Modern industry is based on the flexibility of the production lines. Therefore, changes occur constantly. These changes call for appropriate modifications to the supervising systems. A considerable amount of new training paradigms is required in order to adjust the system [3] at the new environment. In order to provide all the training data an expert, whose services will not be at a low-cost, is needed.

A variety of methods has been used for event detection and especially human action recognition, including semi-latent topic models [4], spatial-temporal context [5], optical flow and kinematic features [6], and random trees and Hough transform voting [7]. Comprehensive literature reviews regarding isolated human action recognition can be found in [8],[9].

The idea of this paper is the creation of a decision support mechanism for the workflow surveillance in an assembly line that would use few training data, initially; as time passes could be self-trained or, if it is necessary, ask for an expert assistance. That way, the human knowledge is incorporated at the minimum possible cost.

The innovation can be summarized to the following sentence: *We propose a cognitive system which is able to survey complex, non-stationary industrial processes by utilizing only a small number of training data and using a self-improvement technique through time.*

This paper is organized as follows: Section 2 provides a brief description of the proposed methodology. Section 3 refers to the data extraction methodology. Section 4 describes the genetic algorithm application. Section 5 presents the main classifier for the system. Section 6 presents the semi-supervised approach. Section 7 explains the decision mechanism of the system, and Section 8 provides the experimental results.

II. The Proposed Self Cognitive Visual Surveillance System

The proposed system was tested using the NISSAN video dataset [10], which refers to a real-life industrial process videos regarding car parts assembly. Seven different, time-repetitive, workflows have been identified, exploiting knowledge from industrial engineers. Challenging visual effects are encountered, such as background clutter/motion, severe occlusions, and illumination fluctuations.

The presented approach employs an innovative self-improvable cognitive system, which is based on a semi-supervised learning strategy as follows: Initially, appropriate visual features are extracted using various techniques (Section 3). Then, visual histograms are formed, from these features, to address temporal variations in executing different instances of the same industrial workflow. The created histograms are fed as inputs to a non-linear classifier.

The heart of the system is the automatic self-improvable methodology of the classifier. In particular, we start feeding the classifier with a small but sufficient number of training samples (labeled data). Then, the classifier is tested on new incoming unlabeled data. If specific criteria are met, the classifier automatically selects suitable data from the set of the unlabeled data for further training. The criteria are set so that only the most confident unlabeled data will be used on the new training set.

If a vague output occurs, for any of the new incoming unlabeled data, a second classifier, which exploits similarity measure among the in-sampled and the unlabeled data, is used. If classifiers disagree, an expert is called to interweave at the system to improve the classifier accuracy. The intervention is performed, in our case with a totally transparent and hidden way without imposing the user to acquire specific knowledge of the system and the classifier.

III. Visual Representation of Industrial Content

From all videos, holistic features such as Pixel Change History (PCH) are used. These features remedy the drawbacks of local features, while also requiring a much less tedious computational procedure for their extraction [11]. A very positive attribute of such representations is that they can easily capture the history of a task that is being executed. These images can then transformed to a vector-based representation using the Zernike moments (up to sixth order, in our case) as it was applied at [12].

The video features, once exported, had a 2 dimensional matrix representation of the form $m \times l$, where m denotes the size of the 1×m vectors created using Zernike moments, and l the number of such vectors. Although m was constant, l varies according to the video duration. In order to create constant size histogram vectors, which would be the system's inputs, the following steps took place:

1. The hyperbolic tangent sigmoid transformation was applied to every video feature. As a result the prices of the 2-d matrices range from -1 to 1.
2. Histogram vectors of 33 classes were created. The number of classes was defined after various simulations. Higher number of classes leads to poor performance due to the small training sample (in our case 48 vectors). Fewer classes also caused poor performance probably due to loss of important information from the original features. Each class counts the frequency of the appearance of a value (within a specific range) for a particular video feature.
3. Finally, each histogram vector value is normalized. Thus, the input vectors were created.

It is clear that each histogram vector describes a specific job among seven different. These histograms, one at a time, are the inputs for a feed forward neural network (FFNN). The target vectors are seven-element arrays. The value at each array will be either one or zero. The number one denotes in which category is categorized the video (e.g., 0 0 0 1 0 0 0 correspond to assembly procedure number four).

IV. The Island Genetic Algorithm

The usefulness of the genetic algorithms (GAs) is generally accepted [13]. The island GA uses a population of alternative individuals in each of the islands. Every individual is a FFNN. While eras pass networks' parameters are combined in various ways in order to achieve a suitable topology.

A pair of FFNNs (parents) is combined in order to create two new FFNNs (children). Children inherit randomly their topology characteristics from both their parents. Under specific circumstances, every one of these characteristics may change (mutation). The quartet, parents and children, are then evaluated and the two best will remain, updating that way the island's population. An era has passed when all the population members participate in the above procedure. In order to bate the genetic drift, population exchange among the islands, every four eras. The algorithm terminates when all eras have passed. Initially, the parameters' range is described in Table 1 and the main steps of the genetic algorithm are shown in Figure 1. The algorithm is used to parameterize the topology of the non-linear classifier (Section 5).

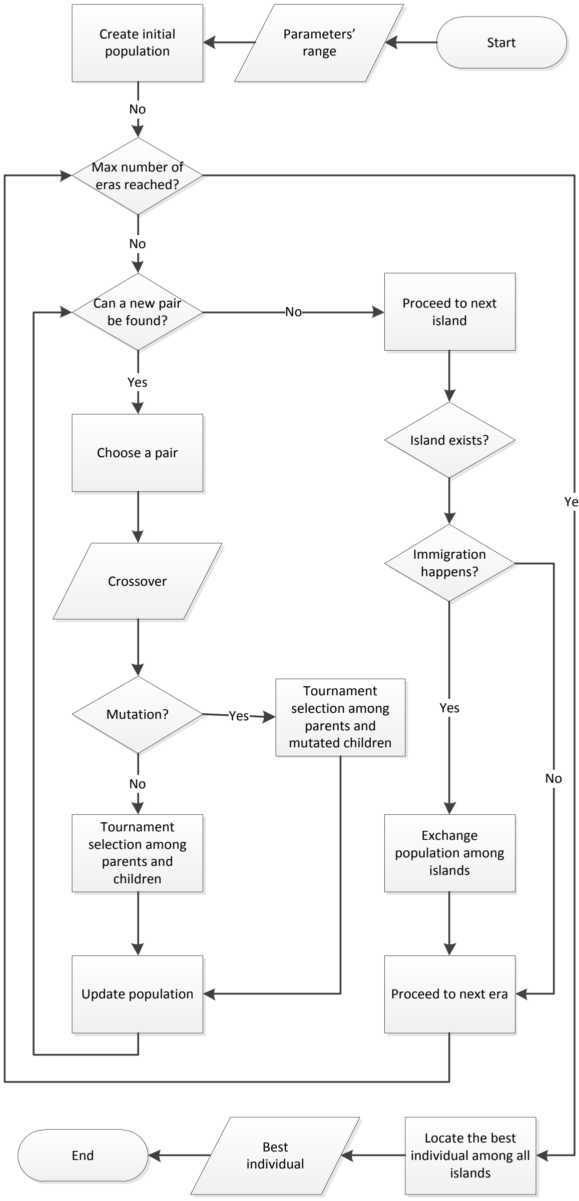

Start
Parameters' range
Create initial population
No
Max number of eras reached?
No
Can a new pair be found?
No
Proceed to next island
Yes
Choose a pair
Island exists?
Yes
Crossover
Immigration happens?
Mutation?
Yes
Tournament selection among parents and mutated children
Yes
No
No
Tournament selection among parents and children
Exchange population among islands
Update population
Proceed to next era
End
Best individual
Locate the best individual among all islands

Figure 1. The island genetic algorithm flowchart.

Regarding the activation functions, the alternatives were five: tansig, logsig, satlin, hardlim, and hardlims. Individuals may mutate at any era. Mutation can change any of the, previously stated, topology parameters therefore individuals' parameters outside the initially defined range may occur. The fitness of a network is evaluated using the following equation:

$$f_i = \lambda p_i + (1-\lambda)a \qquad (1),$$

where f_i denotes the network's fitness score, p_i is the percentage of the correct in-sample classification and a is the average percentage difference, between the two greatest prices, among all the individual's outputs.

TABLE 1 ISLAND GENETIC ALGORITHM PARAMETERS' RANGE.

Parameter	Min value	Max value
Training epochs	100	400
Number of layers	1	3
Number of neurons (per layer)	4	10
Number of islands	3	3
Number of eras	10	10
Population (per island)	16	16

V. THE NONLINEAR CLASSIFIER

In this paper, the nonlinear classifier is a genetically optimized (topologically) feed forward neural network, according to the training sample. The neural network's topology is defined by the number of hidden layers, the neurons at each layer, the activation functions. All of the above as well as the number of training epochs were optimized using an island genetic algorithm.

Synaptic weights and bias values are, also, major factors of a network's performance. Nevertheless, since the initial training sample is small and noise exist at the data a good weight adaptation, for the in sample data, would not lead, necessarily, at an acceptable for the out of sample, performance.

Once the training phase is concluded, we start feeding the optimal network unlabeled data. Since the output vector of the classifier contains various values (its actual size is 1×7 as the number of the possible tasks), the output element with the greatest value will be turned into 1 while all the other ones will be set to 0. This is performed only if the greatest value is reliable. The conditions for the reliability are explained at the following section.

VI. THE SEMI SUPERVISED APPROACH

The main issue, in order to improve network's performance, is the reliability of labeling the new data, deriving from the pool of the unlabeled ones, exploiting network's performance in the already labeled data. In this approach output reliability is performed by comparing the absolute value of the greatest output element with the second greatest according to some criteria. If these criteria are not met, the output is considered vague, otherwise the classifier output is considered as reliable.

An unsupervised algorithm, like the k-means [14], is used in case of ambiguous results to support the decision. In particular, the unlabeled input vector that yields the vague output, say **u**, is compared with all the labeled data, say $\mathbf{l}_i$, based on a similarity distance and then the distance values are normalized in the range of [0 1] so that all comparisons lie within a pre-defined reference frame, say $d(\mathbf{u},\mathbf{l}_i)$. Then, the k-means algorithm is activated to cluster, in an unsupervised way, all the normalized distances $d(\mathbf{u},\mathbf{l}_i)$ into a number of classes, equal to the number of available industrial tasks (7 in our case). In the sequel, the cluster that provides the maximum similarity (highest normalized distance) score, of the unlabeled data that yield the vague output and the labeled ones, is located. Let us denote as K the cardinality of this cluster (e.g., the number of its elements). In the following, the neural network output for the given unlabeled datum is linearly transformed according to the following formula,

$$\mathbf{n}_f = \mathbf{n}_p + \sum_{i=1}^{K} d(\mathbf{u},\mathbf{l}_i)\cdot \mathbf{v}_i \qquad (2),$$

where **n** is the modified output vector, $\mathbf{n}_p$ the previous network output before the modification, while $d(\mathbf{u},\mathbf{l}_i)$ is the similarity score (distance) for the i-th labeled datum $\mathbf{l}_i$ and the unlabeled datum **u** within the cluster of the highest normalized distance, while $\mathbf{v}_i$ is the neural network output when input is the i-th labeled vector $\mathbf{l}_i$ and K is the cardinality of the cluster of the maximum highest similarity.

The modified output vector **n** which is the base for the decision is created using both manifold (FF neural network) and cluster assumption (similarity mechanism) [15].

VII. THE DECISION MECHANISM

According to the nonlinear classifier output, there are three possible cases:

1. The network made a robust decision that should not be defied. Therefore, the unlabeled data is used for further training but it is not incorporated at the initial training set.
2. The output is fuzzy, in other words, the difference among the two greatest prices does not exceed the threshold values. The similarity-based classifier is activated. If both systems indicate the same then the unlabeled data is used for further training but it is not incorporated at the initial training set.
3. The two classifiers do not agree. Therefore, an expert is called and specifies where the video should be classified. That video is added to the initial training data set.

The combination of these cases leads to a semi-supervised decision mechanism. Threshold values define which from the above scenarios will occur. The threshold value is defined as the percentage of the difference between the two greatest prices at the output vector. The overall process for the decision making is shown in Figure 2.

Initially, the first threshold value is set to 0.6. That value means that if the percentage difference of the two greatest values is above or equal to 60% we will be at scenario No 1.

The second threshold value is set to 20%. If the percentage difference of the two greatest values is less than that, the system is unable to make a decision and an expert is needed to interfere. Therefore, scenario No 3 will occur. Any value between these two thresholds activates scenario case No 2.

Since the model is self-trained, the first threshold value does not need to be so strict. The model learns through time, thus a reduction at that value would be acceptable. Nevertheless, at the beginning small threshold value could lead the model to wrong learning. Using simulated annealing method, the threshold descents to a 40% through time.

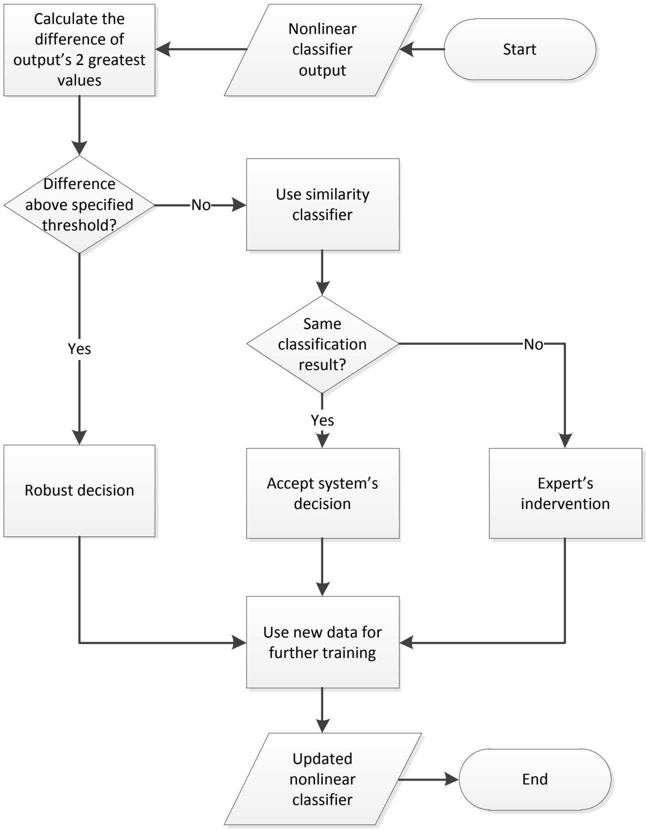

Start
Nonlinear classifier output
Calculate the difference of output's 2 greatest values
Difference above specified threshold?
No
Use similarity classifier
Same classification result?
No
Yes
Yes
Robust decision
Accept system's decision
Expert's indervention
Use new data for further training
Updated nonlinear classifier
End

Figure 2. The decision mechanism flowchart.

VIII. Experimental Validation

The production cycle on the industrial line included tasks of picking several parts from racks and placing them on a designated cell some meters away, where welding took place. Each of the above tasks was regarded as a class of behavioral patterns that had to be recognized. The behaviors (tasks) we were aiming to model in the examined application are briefly the following:

1. One worker picks part #1 from rack #1 and places it on the welding cell.
2. Two workers pick part #2a from rack #2 and place it on the welding cell.
3. Two workers pick part #2b from rack #3 and place it on the welding cell.
4. One worker picks up parts #3a and #3b from rack #4 and places them on the welding cell.
5. One worker picks up part #4 from rack #1 and places it on the welding cell.
6. Two workers pick up part #5 from rack #5 and place it on the welding cell.
7. Workers were idle or absent (null task).

For each of the above scenarios, 20 videos were available. An illustration of the working facility is shown in Figure 3.

A. Experimental setup

Initially, the best possible network is produced using the island genetic algorithm and 40% of the available data. The remaining data are fed to the network, one video at a time, and the overall out of sample performance is calculated.

In every case, all the data that activated scenario No 3 is excluded. Then, we reefed the network, one by one, with the rest data. If the network's suggestions were correct it will perform better since more training data (excluding these from scenario No 3) were used for further training.

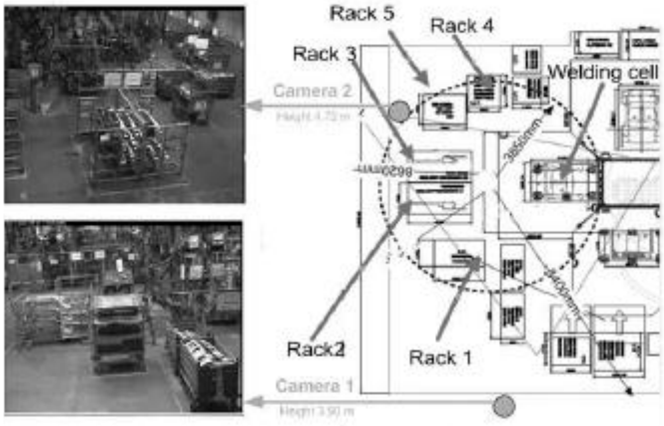


Figure 3. Depiction of a work cell along with the position of camera 1 and the racks #1-5.

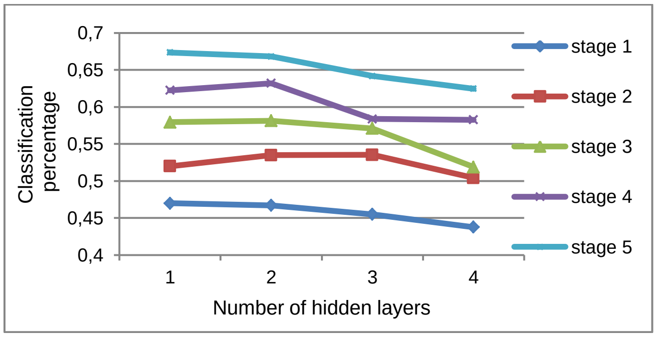


Figure 4. Classification percentages for each of the 5 evaluation stages – out of sample data.

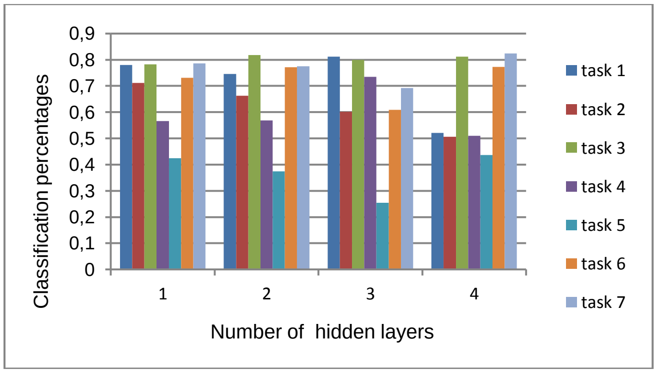


Figure 5. Stage 5 results for each one of the 7 tasks – out of sample data.

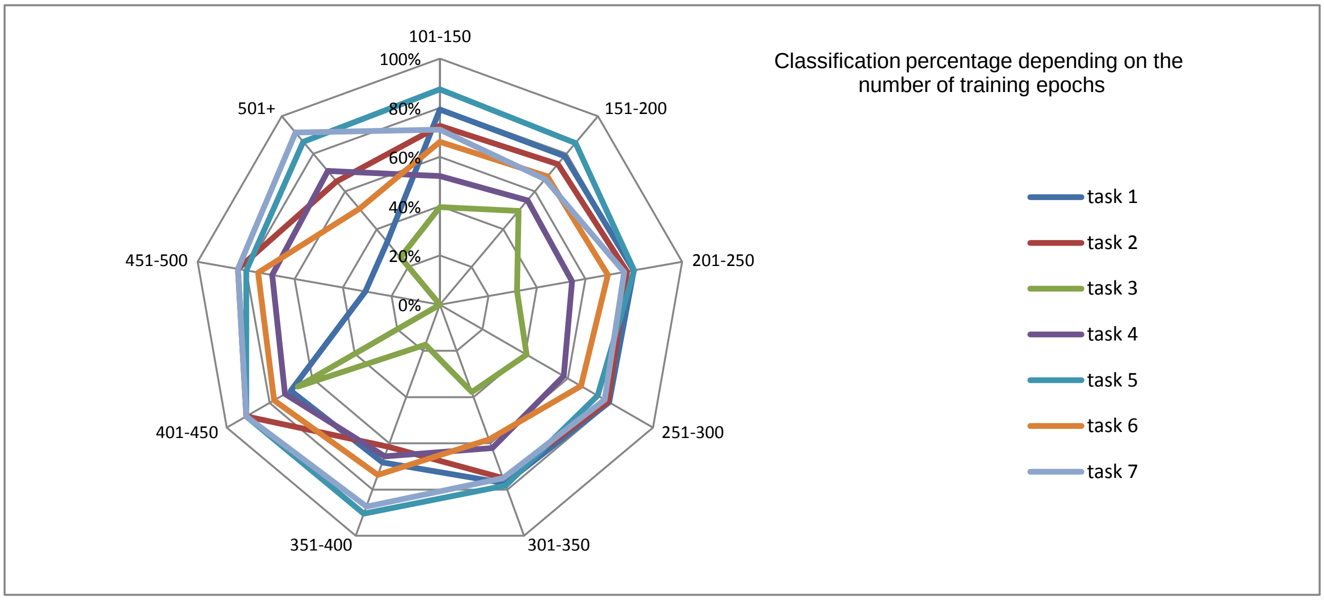


Figure 6. Classification percentage of the system depending on the number of training epochs of the nonlinear classifier.

By doing so, the unlabeled data fall below 60% and training data increases further. The above procedure concludes after five iterations. At that time the ratio between in sample data and out of sample data does not exceed 50%.

B. *Results*

The results displayed below are the average numbers after a total of 150 simulations of the proposed methodology. It appears that a two hidden layers neural network using tansig or logsig activation functions with an average of 9 neurons in each layer is the most suitable solution.

The proposed system is able to use the new knowledge to its benefit. The overall performance increases through iterations, using a small amount of data, as it is shown in Figure 4. Actually, by using additionally 10% of the videos, the system reached a 75% correct classification. This is important because the system saves time and resources during the initialization and provides good classification percentages using less than 50% of the available data.

The impact of the training epochs at the overall performance is shown at Figure 6. There appear to be a tradeoff between overall and individual task classification. Although 200 up to 300 training epochs provide significant classification accuracy further training increases partially the accuracy only on specific tasks in expense on others.

IX. Conclusion and Future Work

In this work, we have proposed a novel framework for behavior recognition in workflows. The above methodology handles with an important problem in visual recognition: it requires a small training sample in order to efficiently categorize various assembly workflows. Such methodology will support the visual supervision of industrial environments by providing essential information to the supervisors and supporting their job.

Improvements at any stage of the system can be made in order to further refine the system's performance. Future work will be based on the usage of different classifiers (e.g. neuro-fuzzy, linear Support Vector Machines) and decision mechanism (e.g. voting-based). In addition, instead of using all frames of a specific task to create classifiers' input, only a subset of them may be used providing equivalent results.

Acknowledgment

The research leading to these results has been supported by European Union funds and national funds from Greece and Cyprus under the project ”POSEIDON: Development of an Intelligent System for Coast Monitoring using Camera Arrays and Sensor Networks” in the context of the inter-regional programme INTERREG (Greece-Cyprus cooperation) - contract agreement K1 3 10–17/6/2011.

References

[1] Y. Kim and H. Ling, “Human Activity Classification Based on Micro-Doppler Signatures Using a Support Vector Machine,” IEEE Transactions on Geoscience and Remote Sensing, vol. 47, no. 5, pp. 1328 –1337, May 2009.

[2] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, “Machine Recognition of Human Activities: A Survey,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, no. 11, pp. 1473 –1488, Nov. 2008.

[3] F. I. Bashir, A. A. Khokhar, and D. Schonfeld, “Object Trajectory-Based Activity Classification and Recognition Using Hidden Markov Models,” IEEE Transactions on Image Processing, vol. 16, no. 7, pp. 1912 –1919, Jul. 2007.

[4] Y. Wang and G. Mori, “Human Action Recognition by Semilatent Topic Models,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 31, no. 10, pp. 1762 –1774, Oct. 2009.

[5] Q. Hu, L. Qin, Q. Huang, S. Jiang, and Q. Tian, “Action Recognition Using Spatial-Temporal Context,” in Pattern Recognition (ICPR), 2010 20th International Conference on, 2010, pp. 1521 –1524.

[6] S. Ali and M. Shah, “Human Action Recognition in Videos Using Kinematic Features and Multiple Instance Learning,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 32, no. 2, pp. 288 –303, Feb. 2010.

[7] Yao, J. Gall, and L. Van Gool, “A Hough transform-based voting framework for action recognition,” in Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 2061 –2068.

[8] R. Poppe, “A survey on vision-based human action recognition,” Image and Vision Computing, vol. 28, no. 6, pp. 976–990, Jun. 2010.

[9] W. Hu, T. Tan, L. Wang, and S. Maybank, “A survey on visual surveillance of object motion and behaviors,” Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 34, no. 3, pp. 334 –352, Aug. 2004.

[10] Voulodimos, D. Kosmopoulos, G. Vasileiou, E. Sardis, A. Doulamis, V. Anagnostopoulos, C. Lalos, and T. Varvarigou, “A dataset for workflow recognition in industrial scenes,” in 2011 18th IEEE International Conference on Image Processing (ICIP), 2011, pp. 3249 –3252.

[11] M. Ahad, J. Tan, H. Kim, and S. Ishikawa, “Motion history image: its variants and applications,” Machine Vision and Applications, vol. 23, no. 2, pp. 255–281, 2012.

[12] D. I. Kosmopoulos, N. D. Doulamis, and A. S. Voulodimos, “Bayesian filter based behavior recognition in workflows allowing for user feedback,” Computer Vision and Image Understanding, vol. 116, no. 3, pp. 422–434, 2012.

[13] D. Whitley, S. Rana, and R. B. Heckendorn, “The island Model Genetic algorithm : On separability, population size and convergence,” CIT. Journal of computing and information technology, vol. 7, no. 1, pp. 33–47.

[14] J. Wu, “Cluster Analysis and K-means Clustering: An Introduction,” in Advances in K-means Clustering, Springer Berlin Heidelberg, 2012, pp. 1–16.

[15] P. K. Mallapragada, R. Jin, A. K. Jain, and Y. Liu, “SemiBoost: Boosting for Semi-Supervised Learning,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 11, pp. 2000 –2014, Nov. 2009.

Analysis of Network Heterogeneity by Using Entropy of the Remaining Degree Distribution

Lu Chen, Shin'ichi Arakawa, and Masayuki Murata
Graduate School of Information Science and Technology
Osaka University
Osaka, Japan
$\{$*l-chen, arakawa, murata*$\}$*@ist.osaka-u.ac.jp*

***Abstract*—As the Internet becomes the social infrastructure, a network design method that has the adaptability against the failure of network equipment and has the 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. Our results show that the mutual information is high at the most of router-level topologies, which indicate that the route-level topologies are highly designed by, e.g., the network operators. We then discuss and show that the mutual information represents the heterogeneity of topological structure through illustrative examples.**

Keywords-power-law network; router-level topology; topological structure; mutual information; network heterogeneity; degree distribution.

I. Introduction

As the Internet becomes the social infrastructure, it is important to design the Internet that has adaptability and sustainability against environmental changes. 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 [1]. 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 [2], 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 Internet topology show that the degree distribution exhibits a power-law attribute [3]. 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 which obey power-law degree distribution are studied widely, and Barabáshi-Albert (BA) model is one of it [4]. In BA model, the topology increases incrementally and links are placed based on the connectivity of topologies in order to form power-law networks. The resulting topology has a large number of links connected with a few nodes, while a small number of links connected with numerous nodes. Topologies generated by BA model are used to evaluate various kind of network performances [5], [6].

However, it is not easy to explain topology characteristics of router-level topology by such models because topology characteristics are hardly determined only by degree distribution [7], [8]. Li et al. [7] 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 topologies. 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 investigate the diversity of router-level topologies by using mutual information of remaining degree. Here, the diversity of topology 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 diversity of topology 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. [2] studied complex networks by using remaining degree distribution as the random variable. They calculated the mutual information of remaining degree 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 use this mutual information to evaluate the diversity of topology.

Milo et al. [9] have introduced a concept called Network Motif. The basic idea is to find several simple sub graphs in complex networks. Arakawa et al. [10] shows the characteristic of router-level topologies by counting the

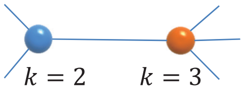


Figure 1. Remaining degree

number of each kind of sub graph which consists of 4 nodes respectively. They conclude that router-level topology has 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. Mutual information of several router-level topologies are calculated, and shown in Section III. In Section IV, we investigate the topological characteristic by changing the mutual information through a rewiring process. Finally, we conclude this paper in Section V.

II. Definitions

Mutual information of remaining degree is defined by Solé et al. [2]. Remaining degree k is the number of edges leaving the vertex other than the one we arrived along. 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. This distribution $q(k)$ is obtained from:

$$q(k) = \frac{(k+1)P_{k+1}}{\Sigma_k k P_k}, \tag{1}$$

where $P(P_1, \dots, P_x, \dots, P_K)$ is the degree distribution, and K is the maximum degree.

The distribution of mutual information of remaining degree, $I(q)$, 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.

The first term $H(\mathrm{q})$ is entropy of remaining degree distribution:

$$H(\mathrm{q}) = -\sum_{k=1}^{N} q(k)\log(q(k)). \tag{3}$$

Within the context of complex networks, it provides an average measure of network's heterogeneity, since it measures the diversity of the link distribution. $H = 0$ in a homogeneous networks such as ring topology. As network become more heterogeneous, the entropy H gets higher. For example, Abilene inspired topology [7] shown in Figure 2 is heterogeneous in the degree distribution, thus it has higher entropy.

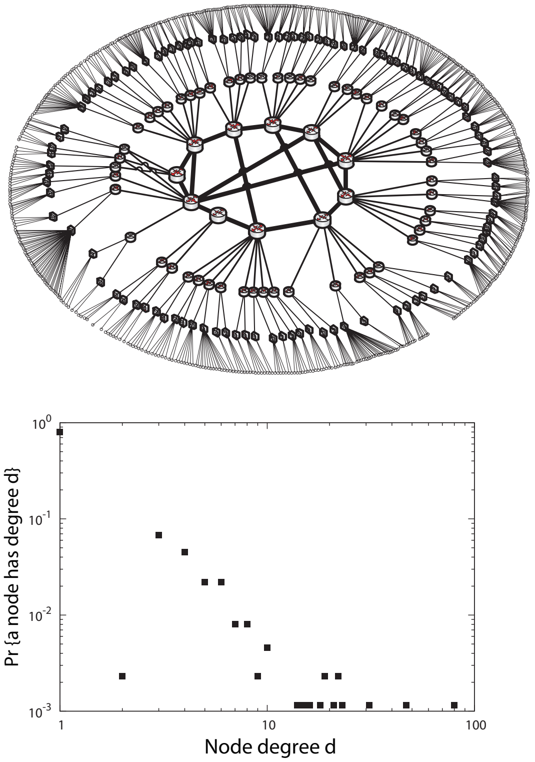


Figure 2. Abilene ($H = 3.27$,$H_c = 2.25$)

Table I
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
Random	523	1304	3.22	3.15	0.07

The second term $H_c(\mathrm{q}|\mathrm{q'})$ is the conditional entropy of the remaining degree distribution,

$$H_c(\mathrm{q}|\mathrm{q'}) = -\sum_{k=1}^{N}\sum_{k'=1}^{N} q(k')\pi(k|k')\log\pi(k|k'), \tag{4}$$

where $\pi(k|k')$ are conditional probability. They 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. For Abilene inspired topology, combinations of remaining degrees which are the ones of a pair of linked nodes are biased; therefore, the conditional entropy H_c is low.

ISBN: 978-1-61208-226-4

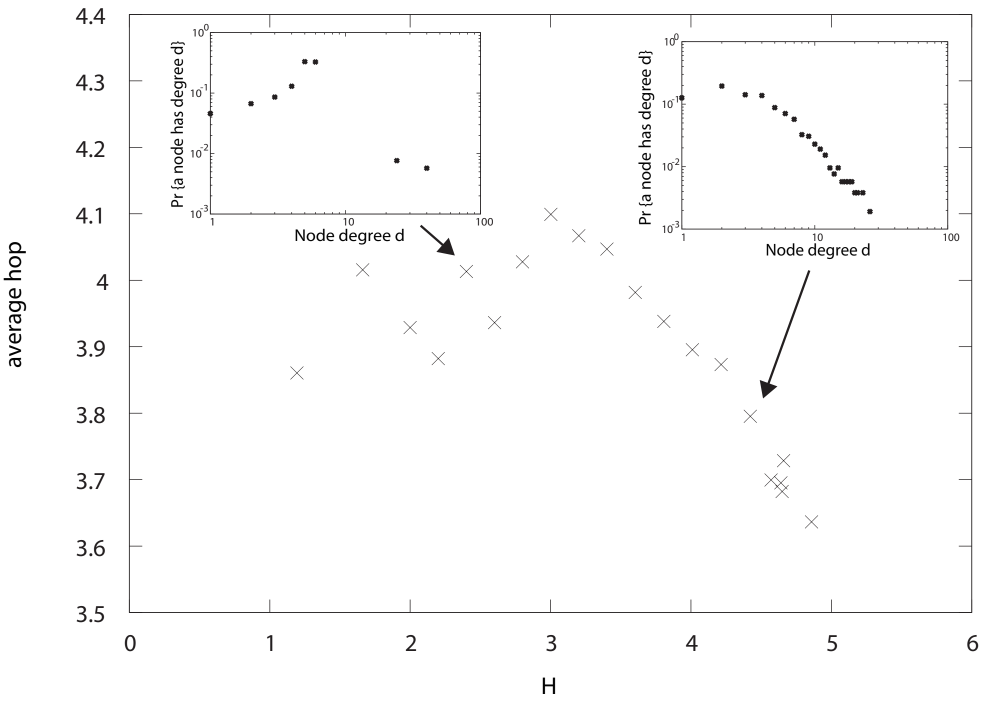


Figure 3. Average hop distance

III. Diversity of router-level topology

In this section, we show the mutual information of some router-level topologies: Level3, Verio, AT&T, Sprint and Telstra. The results are summarized in Table I. The router-level topologies are measured by Rocketfuel tool [11]. To compare with those router-level topologies, topologies made by BA model [4] and ER model [12] which has the same number of nodes and links with AT&T are also shown. From Table I, we can see that, except Verio, the mutual information of router-level topologies are high, and that of model-based topologies, such as the ones generated by BA model and ER model, are low. This can be explained by a design principle of router-level topologies. Because router-level topology is 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 the mutual information of Verio is low. This can be explained by its growing history. Because Verio grows big with small ISPs [13], it contains various kinds of design principles conducted in each ISP. Therefore, Verio is more diverse than other router-level topologies.

IV. Mutual information and the characteristic of topologies

As we mentioned in Section II, mutual information is defined by entropy and conditional entropy. In this section, we explore the relationship between entropy, conditional entropy and the characteristic of topologies respectively.

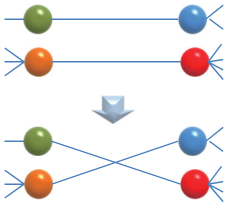


Figure 4. Rewiring method to leave the degree distribution unchanged

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

A. Entropy $H(q)$ and the characteristic

To show the relationship between degree distribution 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

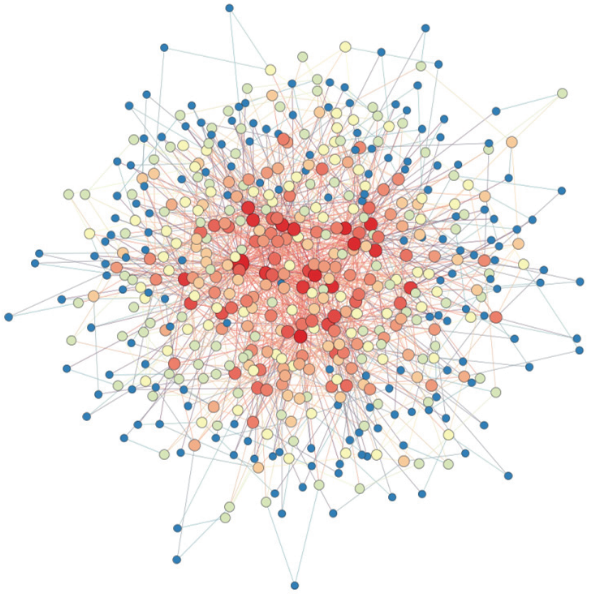


Figure 5. T_{Imin} with minimum mutual information

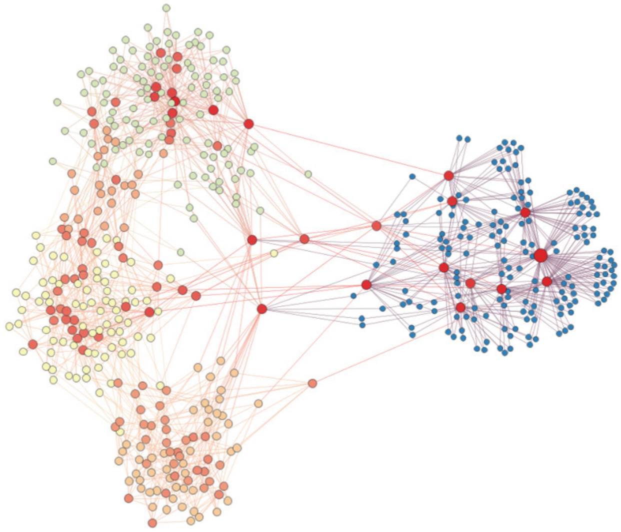


Figure 6. T_{Imax} with maximum mutual information

looks for candidate networks 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 the topology obtained by BA model which has the same number of nodes and links with AT&T. Topologies are changed by random rewiring, and try to minimize the following potential function:

$$U(G) = \sqrt{(H - H(G))^2 + (H_c - H_c(G))^2}. \qquad (5)$$

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 zero. Therefore, entropy and conditional entropy of the generated topologies are almost equal.

Figure 3 shows the average hop distance of topologies we generated. It can be seen 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. Conditional entropy $H_c(q|q')$ and characteristic

Next, we show the relationship between mutual information and the characteristic of topologies. Because router-level topologies obey power-law, we compare topologies having high $H(\mathrm{q})$.

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 [14] that leaves the degree distribution unchanged, i.e., by exchanging the nodes attached to any randomly selected two links (Figure 4). For the second point, the potential function we used to minimize is $U^I(G)$ defined as,

$$U^I(G) = |I - I(G)|, \qquad (6)$$

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 explain the relationship between mutual information and the characteristic of topologies, 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 5. 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 6. In both figures,

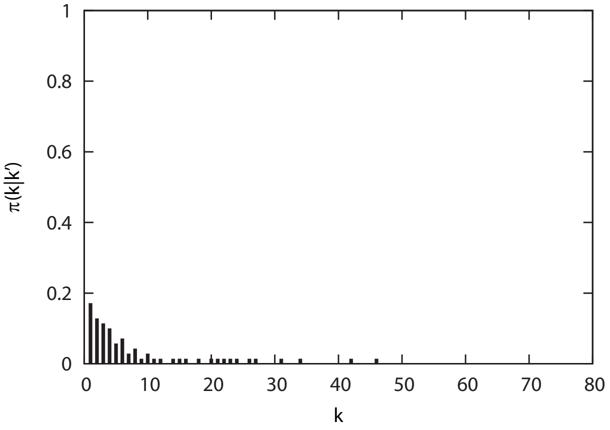


Figure 7. $\pi(k|k')$ of nodes with the largest remaining degree in T_{Imin}

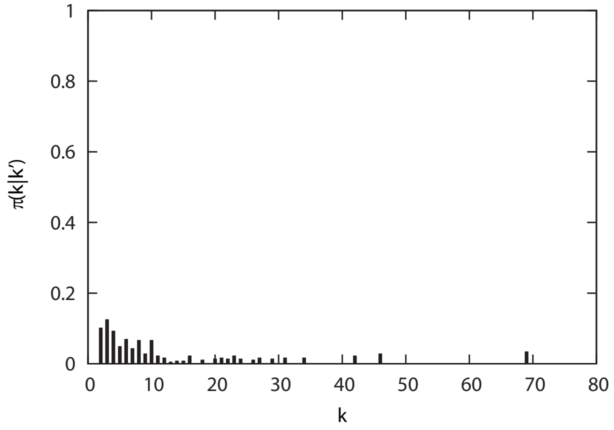


Figure 8. $\pi(k|k')$ of nodes with the smallest remaining degree in T_{Imin}

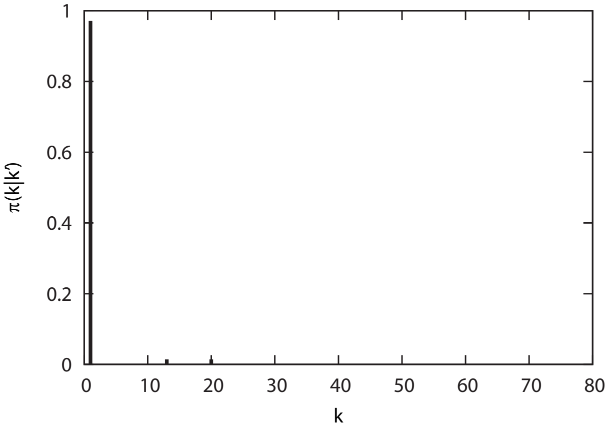


Figure 9. $\pi(k|k')$ of nodes with the largest remaining degree in T_{Imax}

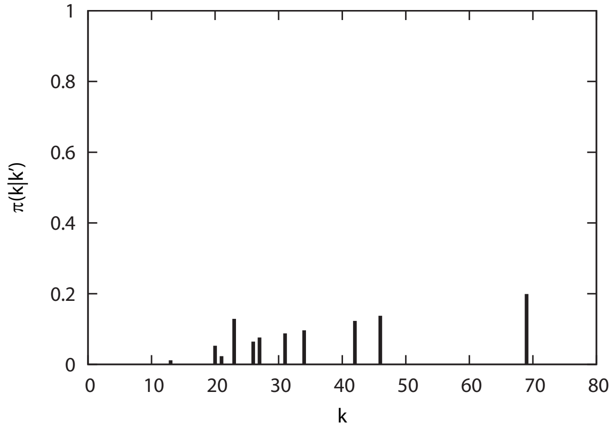


Figure 10. $\pi(k|k')$ of nodes with the smallest remaining degree in T_{Imax}

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 Figure 5 and Figure 6, 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 7 to Figure 10, 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. Figure 7 and Figure 8 show $\pi(k|k')$ of nodes with the largest remaining degree and nodes with the smallest remaining degree in T_{Imin}, respectively. Figure 9 and Figure 10 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.

V. Conclusion and future work

In this paper, we investigated the network heterogeneity of router-level topologies by using mutual information. From calculating mutual information of some router-level topologies, we found that router-level topologies have higher mutual information than model-based topologies. We also generated topologies with different mutual information, and showed that the topology is diverse when mutual information is high, and the topology has regularity when mutual information is low.

Our next work is to evaluate network performance of topologies with different mutual information, and to apply this measure to designing information network that has adaptability and sustainability against environment 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.

REFERENCES

[1] 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*, pp. 704–719, Jul. 2008.

[2] R. Solé and S. Valverde, “Information theory of complex networks: On evolution and architectural constraints,” *Complex networks*, vol. 650, pp. 189–207, Aug. 2004.

[3] M. Faloutsos, P. Faloutsos, and C. Faloutsos, “On power-law relationships of the Internet topology,” *ACM SIGCOMM Computer Communication Review*, vol. 29, pp. 251–262, Oct. 1999.

[4] A.-L. Barabási and R. Albert, “Emergence of scaling in random networks,” *Science*, vol. 286, pp. 509–512, Oct. 1999.

[5] R. Albert, H. Jeong, and A. Barabasi, “Error and attack tolerance of complex networks,” *Nature*, vol. 406, pp. 378–382, 2000.

[6] 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.

[7] 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, pp. 3–14, Oct. 2004.

[8] R. Fukumoto, S. Arakawa, and M. Murata, “On routing controls in ISP topologies: A structural perspective,” in *Communications and Networking in China, 2006. ChinaCom’06. First International Conference on*. IEEE, Oct. 2006, pp. 1–5.

[9] 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.

[10] S. Arakawa, T. Takine, and M. Murata, “Analyzing and modeling router-level Internet topology and application to routing control,” *Computer Communications*, vol. 35, pp. 980–992, May 2012.

[11] 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.

[12] P. Erdős and A. Rényi, “On random graphs,” *Publicationes Mathematicae Debrecen*, vol. 6, pp. 290–297, 1959.

[13] M. Pentz, “Verio grows big with small clients,” *Business Journals*, Feb. 1999.

[14] 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.

Understanding Virtualized Infrastructure in Grid Job Monitoring

Zdeněk Šustr
Grid Department – MetaCentrum
CESNET z. s. p. o.
Zikova 4, Prague, 160 00, Czech Republic
Email: zdenek.sustr@cesnet.cz

Jiří Sitera
Grid Department – MetaCentrum
CESNET z. s. p. o.
Zikova 4, Prague, 160 00, Czech Republic
Email: jiri.sitera@cesnet.cz

***Abstract*—This paper is the first report on a new direction in the development of the Logging and Bookkeeping service, a gLite component tracking grid job life cycle. From the early days, Logging and Bookkeeping tracks not only jobs themselves but also the wider details of the job execution environment. Since a great portion of the infrastructure is now virtualized, the work at hand concerns tracking the virtualized nature of that runtime environment. With virtualization and cloud technologies being highly flexible and dynamic, we believe it is very important to gather and keep status information for machines used to run the workload. A newly created monitoring entity (a machine) will be integrated with job state information and provide an enhanced view of the current state and history of both the job and the infrastructure. This paper focuses on motivation, requirements coming from the Czech National Grid Initiative and possible consequences rather than the actual implementation. As a report on "work in progress" it describes an idea that is now being further elaborated and implemented to provide a solution for monitoring virtualized resources in the same context as the workload they are processing.**

***Keywords-grid*; *cloud*; *virtualization*; *job monitoring*.**

I. Introduction

Logging and Bookkeeping (LB), part of the gLite grid middleware, is a monitoring tool equipped for monitoring the states of all kinds of processes related to grid computing [1]. Besides traditional gLite Workload Management System (WMS) [2] jobs and logical groupings thereof such as oriented graphs (DAGs) or collections it also monitors input/output data transfers and the states of computing tasks submitted directly to a resource manager — the CREAM Computing Element (part of the gLite middleware stack) [3] or to TORQUE (Terascale Open-Source Resource and QUEue Manager) [4].

It collects event information from various grid elements and sums it up to determine the current status of any such process at the given moment. It is designed to accept additional state diagram implementations as required, relying on essential common features such as event delivery (based either on LB's own legacy messaging layer or standard STOMP/OpenWire messaging) or the querying interface. LB is highly security-oriented and has proven itself in WLCG (Worldwide LHC Computing Grid) operations. It is widely deployed across the European Grid Initiative's infrastructure.

In this article, Section II explains what the requirements are and why LB is deemed suitable for monitoring virtualized resources. Section III outlines the proposed solution to deliver essential functionality, and Section IV discusses additional issues to consider and focus on in the future.

II. Motivation to Include Machines in the LB Model

Using LB in monitoring virtualized resources is inspired by obvious similarities with the existing processes, backed by explicit requirements from infrastructure operators.

A. Virtual Machine as a Job

LB's main objective is to know everything about job scheduling and execution, making it possible to analyze the behavior of the infrastructure (failing components, misconfiguration) and possibly even provide job provenance capability (ensuring repeatability of jobs/experiments, storing computing environment characteristics and configuration). In contemporary grids and other computing infrastructures machines running grid jobs are themselves dynamic entities following a lifecycle similar to that of the job itself. It is not unreasonable to expect further blending of cloud and grid models where grid components run either in a cloud (StratusLab [5]), or in a mix with cloud services (MetaCentrum [6], WNoDeS [7]).

All things considered, tracking virtual machines (VMs) throughout their lifecycle in contemporary grids is as important as tracking jobs. Moreover there is an added value to tracking two kinds of entities in a common manner. Not only does it provide for a better understanding of mutual relationships and dependencies, but also for a unified view for users and administrators.

Figure 1 shows a simplified and illustrative example of the new higher-level view of the infrastructure state. It maps compute jobs to the underlying VM lifecycle and provides the user with an overview of its current state and possible problems. In the case of highly dynamic virtualized infrastructure it can be used to assess efficiency and induced tradeoffs. Data collected in this manner can also be used to produce higher-level statistics and monitoring (mapping

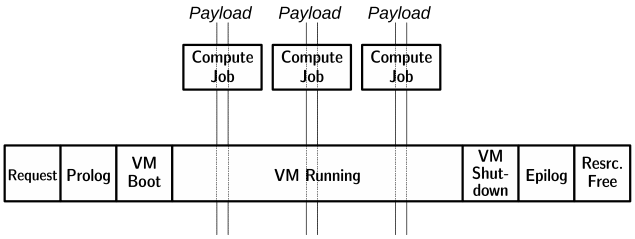


Figure 1. Viewing compute jobs as payload executing over a VM.

actual hardware resources to jobs), while the low-level information is still available for detailed inspection if required for debugging.

Key LB features (re)usable for machines:

- Recording primary events and using a state machine specific to the given type of process (job, VM) to combine all information contained therein and determine the current state of a process.
- Providing the ability to get processes grouped or annotated/tagged by the infrastructure, administrators or users.
- Architecture and implementation based on standards (messaging, authentication and authorization infrastructure, web services), allowing simple event gathering.
- Essential functions (logging events, querying for basic information) provided not only by library functions but also by command line tools.

B. Features Requested by the Czech NGI

MetaCentrum, the Czech National Grid Initiative (NGI), is designed as a mixed cloud/grid service, where resources from a single, consistently managed pool can be provided either as traditional batch system-managed resources or VMs, depending on current user needs [8]. The scheduler (Torque) can handle three types of requests:

1) Run a job
2) Run a job in a selected VM image
3) Run a VM

The desired functionality will provide a single, consistent view of the infrastructure, mapping all user requests to actual hardware. It should replace currently used data mining tools providing status feeds to the MetaCentrum portal and to the long-term usage statistics processor.

Since MetaCentrum is also involved in research of batch system scheduling strategies, gathering data relevant for this kind of assessment is another requirement.

Yet another requirement, albeit one that is already fulfilled by LB's design, calls for an ability to aggregate information from diverse sources (scheduler, virtualization hypervisor, accounting) and even manually triggered state transitions (for instance putting resources in and taking them out of maintenance).

C. Similar Works

Infrastructure monitoring tools such as *Nagios* or *Ganglia* focus primarily on the "running" state of the given process, and using them to monitor short-lived VM instances set up on demand is on the edge of practicality, anyway. Unlike them, this work is not intended to monitor infrastructure health and react to problems. There is just a minor overlap in that certain aspects of infrastructure health can be seen in job/VM status statistics provided by LB and we believe that understanding the relationship between the payload and VM layers will further improve the informative value of LB statistics.

Each infrastructure or cloud management tool has its own way (command line interface, portal) of providing users with the current VM status. But, we are not aware of any other work similar to LB – a service combining available information from different components into one higher-level view. It is one of the reasons for publishing this Work in Progress paper.

We expect that major virtualization stack implementations will be able to send raw status change events via the messaging infrastructure in the near future (indeed, some of them already do) and thus there will be interesting potential in processing them in the proposed way.

III. Proposed Solution

The proposed functionality is being implemented in progressive steps. Early phases are already in progress and can be discussed in detail, while the later phases consist mostly of open issues.

A. Implementation Phases

- Pilot implementation with a testbed instance of Open Nebula, running and keeping track of VMs and scheduled Torque jobs at the same time. This phase has already finished.
- Adjustment to MetaCentrum environment with Torque scheduling VMs as well as jobs. Making sure that the solution is adequately robust in all applicable use cases including those where some of the components (for instance some of the VMs) operate out of the scope of MetaCentrum and do not generate events. It is the current phase as of this publishing.
- Bringing in additional sources of information external to the batch system and virtualization stack: administrative operations, information system, accounting. Automated processing of information produced by LB: statistics, dashboards, etc.

B. Architecture for the First Phase

The primary goal of the first phase was to understand VM lifecycle and its relation to existing job lifecycle. The particular outcome from this phase consisted in finalizing

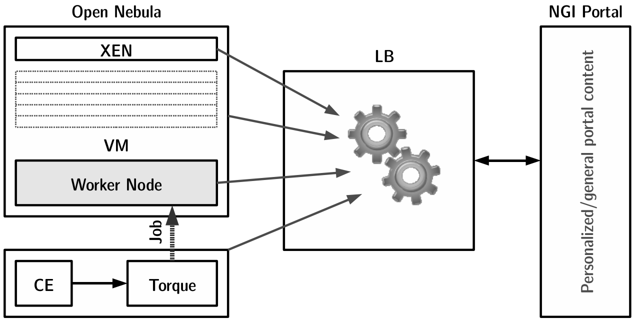


Figure 2. Architecture and components.

VM state machine design and the attribute set for the VM instance status record.

In this phase, the VM lifecycle was controlled solely by a single instance of the Open Nebula cloud computing toolkit, managed manually by administrators, while jobs were being assigned to VMs by a standard grid computing element through an instance of Torque. All job-related functionality was already in place (LB-aware Torque). The following sources of events were used to govern the VM lifecycle:

- *Open Nebula*—providing hooks for call-out scripts activated on any relevant state change
- *Hypervisor, (Specifically Xen)*—generating events showing the current VM state and parameters at hypervisor level
- *Hosted worker nodes*—Operating System running the Worker Node was instrumented (init scripts) to provide independent information from the running VM

The combination of events from all the above components into one higher-level view is a key role of LB in this concept. It makes the system more precise and robust, which has been well tested in the context of gLite job monitoring. Obviously on certain occasions, all three sources generate almost identical, i.e., redundant events. But there is still value in receiving almost identical events multiple times. It improves reliability, and the comparison between the three events provides for fine-grained job status tracking and simplifies troubleshooting. Besides that, different sources often provide values for different attributes unknown to the others.

System architecture for the initial implementation is shown in Figure 2. In that design, the only new feature that had to be implemented was VM instance support in LB (state machine, attributes, event types). Relationship to other relevant components of the system (virtual image identification, physical machine identification) is stored in the form of attributes in that instance. There are other attributes to cover the network status of the VM such as domain name, type of network connectivity (VLAN, private/public) and of course even more attributes identified as useful in the design/implementation process. The complete set of desirable attributes did not need to be pre-determined, though. LB allows any kind of additional attribute to be simply stored with the instance's status (functionality referred to as User Tags) with only slight limitations. One cannot, for instance, use relations such as "greater than" or "lower than" when querying for instances with a given value of such attribute. Since LB does not know the type of that attribute and cannot decide. The only comparison supported is string (in)equivalence.

Each instance is identified by a string constructed in the same manner as Job IDs currently used in LB, consisting of the LB server's identification, a short literal denoting the process type, and a random unique string. Domain names are not suitable for use as identifiers since they are often recycled (re-used by another instance) or even used by multiple VM instances at once.

Any event received by LB may or may not trigger a change in the state and/or attribute values of an instance. Thus the instance's current state and attributes constitute the most up-to date information set as collected from all the various sources mentioned above. LB is designed to overcome obstacles such as events delivered out of sequence, intermediate events not delivered at all, or events received from different sources with clocks skewed in different directions. This is achieved by relying on arbitrary hierarchical message sequence codes rather than time stamps in event sorting.

IV. Further Ideas and Open Issues

Given that this is still work in progress there are many concepts and ideas that deserve further investigation. Some of them, such as virtual cluster support, are necessities that must be addressed. Others fall into the "nice to have" category. They will receive attention at a later stage.

- *State Machine for Physical Machines?*—At the very least VM instance attributes will refer to a physical machine by name. But there is an obvious similarity between physical and virtual machines and a VM state diagram is easily applicable to physical machines. So the option is to register physical resources as "VM" instances as well, and reference the identifier instead. Then the same level of detail could be provided for virtual and physical machines alike, although some supported states will probably remain unused in the physical world.
- *Support of User Workflows*—Compared to traditional computing jobs, VMs are a little specific in that they always need to be assigned workload when running (i.e., having started for the first time, recovered from a downtime or finished migration), which makes them actually very similar to pilot jobs. Many user groups rely on their own workload management systems to distribute payload and it may be very convenient for them to receive notifications of relevant VM status changes.

That could be easily achieved with LB notifications generated on pre-determined conditions and sent out over LB's own legacy messaging chain or through a STOMP/OpenWire-enabled messaging broker. Users may choose, for instance, to be notified any time any of their machines reaches state *running*. More elaborate sets of conditions are also supported. The resulting notification contains the full VM status information and, if requested on registration, also the full history of events for that machine so far.

- *Virtual Cluster Implementation*—The Virtual Cluster service provided by MetaCentrum can create multiple VM instances per request [6]. All the resulting VMs have common attributes (type of network connection) and are closely related. It may be a good idea to reuse the "collections" functionality in LB, typically applied to grid jobs or sandbox transfers. From the user's point of view the state of the collection combines the states of all its members. Individual VM details are still accessible under the VM instance's own ID – the collection functionality simply adds another identifier (collection ID) to access aggregate information such as child status histograms.
- *Heterogeneous Environment (multiple hypervisors and cloud managers)*—LB should be able to provide a unified view of VMs running on different implementations of hypervisors or even cloud managers. The situation is similar to that of a unified state machine used for different job managers – CE implementations.
- *VLAN Status*—The Virtual Cluster service offered by MetaCentrum provides not only sets of machines but also networking connections in the form of virtual Ethernet (VLAN) [9]. The VLANs have their own lifecycle managed by a purpose-built VLAN manager (SBF). An ability to track the state of the network together with its attributes (private/public, additional service such as tunnel/NAT/FW) could be valuable in many scenarios.

V. Conclusion

Although this work is primarily driven by the Czech NGI's requirements, it will be found useful at a much wider scope. With instances of LB currently deployed at dozens of gLite-enabled grid sites across the European Grid Initiative's infrastructure, the VM monitoring feature – once released – will become available to a wide base of users, not only those already relying on LB for monitoring their own computing jobs, but also to those exploring the potential use of cloud services on grid-based platforms.

This paper's main goal was to show how the potential of job monitoring infrastructure can be reused in the virtualized world. Many cloud-oriented initiatives are currently looking for solutions enabling resource federation. LB, with its current presence resulting in easy adoption, will be a reasonable candidate for a monitoring and notification service.

Acknowledgment

This work is part of the National Grid Infrastructure MetaCentrum, provided under the programme "Projects of Large Infrastructure for Research, Development, and Innovations" (LM2010005).

Fundamental development and maintenance of the Logging and Bookkeeping service is co-funded by the European Commission as part of the EMI project under Grant Agreement INFSO-RI-261611.

References

[1] MetaCentrum Project, *Logging and Bookkeeping*, CESNET, 2008. [Online]. Available: http://egee.cesnet.cz/en/JRA1/LB/ [Accessed: August 28, 2012].

[2] M. Cecchi et al., The gLite Workload Management System, *J. Phys.: Conf. Ser.*, vol. 219, 2010.

[3] P. Andreetto et al., Status and Developments of the CREAM Computing Element Service, *J. Phys.: Conf. Ser.*, vol. 331, 2011.

[4] G. Staples, TORQUE resource manager, *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*, ser. SC '06, 2006, ISBN 0-7695-2700-0.

[5] StratusLab Project, *StratusLab*, StratusLab, 2012, [Online]. Available: http://stratuslab.eu/ [Accessed: August 28, 2012].

[6] M. Ruda et al., *Virtual Clusters as a New Service of MetaCentrum, the Czech NGI*, CESNET, 2009. [Online]. Available: http://www.cesnet.cz/doc/techzpravy/2009/virtual-clusters-metacentrum/ [Accessed: August 28, 2012].

[7] D. Salomoni et al., "WNoDeS, a tool for integrated Grid and Cloud access and computing farm virtualization," *J. Phys.: Conf. Ser.* 331 052017, Dec 2011.

[8] J. Sitera, M. Ruda, P. Holub, D. Antoš, and L. Matyska, *MetaCentrum Virtualization – Use Cases*, CESNET, 2010. [Online]. Available: http://www.cesnet.cz/doc/techzpravy/2010/metacentrum-virtualization-use-cases/ [Accessed: August 28, 2012].

[9] D. Antoš, L. Matyska, P. Holub, J. Sitera, "VirtCloud: Virtualising Network for Grid Environments – First Experiences" in *The 23rd IEEE International Conference on Advanced Information Networking and Applications AINA*. Bradford, UK, 2009.

A Layered Multi-Tree IP Multicast Protocol With Network Coding

Yining Li, Xuelei Tan, Hui Li
Shenzhen Key Lab of Cloud Computing
Shenzhen Graduate School, Peking University
Shenzhen, China.
liyining@sz.pku.edu.cn, tanxl@sz.pku.edu.cn,
lih64@pkusz.edu.cn

Jinguo Quan
Shenzhen Graduate School, Tsinghua University
Shenzhen, China.
quanjg@sz.tsinghua.edu.cn

Abstract—**This paper discusses the scenario of multi-rate multicasting to heterogeneous receivers. We adopt the Multi-Resolution Code as the layered source coding scheme, and proposed Forest, a layered multicast protocol based on multiple distribution trees. Each tree transmits a multicast layer, and Network Coding is allowed between different multicast layers. Compared to the existing solutions, our approach is completely distributed and with high performance and low complexity. Also, our approach provides a receiver-driven service model, as well as a complete group management that supports dynamic joins/leaves. These features are vital to the feasibility of a practical deployment. Simulation shows that performance and feasibility are well balanced in our approach.**

Keywords-layered multicast; network coding; multi-tree

I. INTRODUCTION

Multicast is the resource-saving way to transmit streaming media to multiple receivers. But different receivers have different capabilities and requirements. A single-rate multicast flow could either overwhelm the low-capacity receivers, or starve the high-capacity receivers [1]. Multi-rate multicasting has gained more attention since 1990's when single-rate multicasting was found insufficient to fulfill the conflicting requirement of a set of heterogeneous receivers. Today, the heterogeneity and scale of the Internet are growing explosively; so does the proportion of the Internet traffic consumed by streaming media applications. It is desirable to provide each receiver a rate that can commensurate with receiver's capability and requirement [2]. One instinct way to do multi-rate multicast is to split the original stream into layers, then transmit each layer of the original stream on an independent single-rate multicast sub-session [1]. Receivers adjust their number of subscribed sub-sessions according to their own demand.

Network coding (NC) is a promising paradigm in the field of information theory [3]. NC brings new features to the transmission of streaming media such as throughput gains, security and load balancing, etc. It is proven max-flow rate of a single-rate multi-cast session, which equals to the minimum value of the max-flow rate from each subscriber to the source, can be achieved by using linear network coding [4]. In the layered multi-rate scenario, NC can be applied within each layer to provide max-flow transmission layer by layer [5]. NC can also be applied across layers to provide more complete optimization.

Apart from the above, a lot of works have been done in the layered multicast direction. Eros [6] and Goyal [7], respectively, showed two mainstream layered source coding technologies that have been applied to the layered multicasting, Multi-Resolution Coding (MRC) and Multiple Description Coding (MDC). Mingkai [8] showed that if network coding (NC) is allowed, a MRC-based intra-layer NC solution always outperforms or at least performs the same as the MDC-based Uneven Erasure Protection (UEP) solution. So we adopt MRC as the source coding scheme in our approach. Kim [9] proposed a pushback algorithm to gather the requirements of the receivers before distributing data. But it did not explicitly distinguish different multicast layers. And the major disadvantage of pushback algorithm is intermediate nodes needed to perform NC decoding operations to fulfill the requirements of the receivers, which is extremely resource consuming. In our approach, intermediate node decoding is not required. Shao [10] attempted to combine linear NC with rainbow network flow and got a higher network throughput than original rainbow network. But this approach is related to the linear broadcast problem. We mainly focus on multicast. Mingkai [11] proposed an inter-layer NC approach to layered multicast that allowed NC of data in different layers. And higher throughput could be gained with the increasing of related cost. Zhao [12] proposed a heuristic algorithm to organize receivers into layered meshes. While in our approach, we use the thought of MRC in layered algorithm. Other related researches include [13].

In this paper, we propose Forest with a full name IP Multicast Forest (IPMF). It is a layered multicast protocol with NC applied. The novelty of our approach lies in the following. First, we create multiple distribution trees, one for each multicast layer. NC between distribution trees is

ISBN: 978-1-61208-226-4

allowed. Second, we introduce the Coding Matrix (CM) to be multicast to all potential subscribers. The CM indicates the distribution of the multicast layers, and NC layers, which is the linear mixing of some multicast layers. Before subscribing, potential subscribers are required to inquire the CM to decide which sub-sessions to join.

The rest of this paper is organized as follows: Section 2 describes the Forest framework. Section 3 discusses main implementation details of Forest. Simulation settings and results are presented in Section 4. Section 5 concludes the whole paper and introduces our future work.

II. FOREST FRAMEWORK

A. Basic Idea

The basic idea of Forest is to create multiple distribution trees. Forest splits and recodes the original multicast flow at the source node into sub-flows, using algorithms described in [6]. We first split out the most basic, important data, and then recode them to form a "base sub-flow", while other data "enhancement sub-flows". Each sub-flow is associated with a sub-source node and a sub-group address. Multiple multicast layers have multiple distribution trees. We call it the sub-tree for it is logically part of the original multicast distribution structure. Subscribers that want to subscribe to the original multicast session, have to collect all the sub-flows to rebuild the original flow. From this point of view, Forest is an extension to the traditional tree-like distribution structure. It inherits the convenient group member management of the distribution tree, while expanding its transmission performance.

While, this Forest structure may also encounters the so-called "Multicast Packing Problem", for sub-trees in the forest must be edge-disjoint to avoid congestion. When sub-tree collision happens, bottlenecks may emerge, and transmission performance will drop rapidly. Luckily, NC can simplify this problem. In Forest, when collision happens at a node between multicast layers, we can code them together.

B. Forest Structure

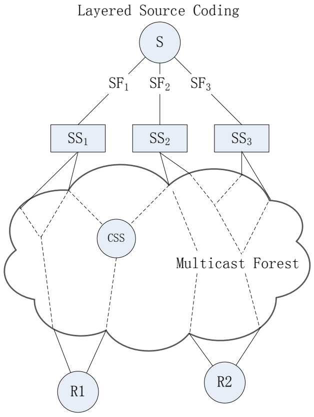


Figure 1. Main Forest structure

Figure 1 shows the main Forest structure. There are 5 different kinds of nodes in the Forest architecture: source, sub-source, coding node, forwarding node and receiver. The entire Forest architecture can be partitioned into two parts: layered source coding, and multicast forest. And some main definitions of terms as well as their features used in this paper are showed below.

In Figure 1, S and R, respectively, represent the source and receiver.

SF: A sub-flow (SF) is a data flow, formed by splitting and/or recoding the original multicast flow at the source node.

SS: A sub-source (SS) node is a different node. One sub-source is assigned to one sub-flow. A sub-source node will receive and cache sub-flow information from the source node. And it will act as a new source including constructing the tree-like multicast distribution structure, managing group members, and sending data out to the output interface list.

ST: A sub-tree (ST) is the multicast distribution tree constructed by a sub-source node, using IGMP join/prune messages, and is consisted of intermediate nodes and links. All the sub-trees that belong to the original multicast session form a so-called coding-sub-graph of IPMF.

Multicast Forest: In Figure 1, if we cover the flow-splitting part, the multicast forest part can be seen as multiple independent multicast distribution trees, each transmitting a sub-flow to the same receiver group concurrently.

CSS: A node who begins to carry out coding operations when collision happens, will automatically transform into a "coding-sub-source" (CSS). In most cases, a CSS acts just like other independent sub-sources, but, there are still differences. Firstly, a CSS receives information from some sub-flows not only one. Secondly, multiple CSSs may sit on one physical coding node like Figure 2. Thirdly, if a coding-sub-flow has *k* parent sub-flows, then, when a group receiver collects sub-flows, the coding-sub-flow can replace any one of its parent sub-flows, only when the "sub-flow coefficient matrix" of that receiver is reversible after the replacement. The last, when a CSS emerges, it will send an inform message to the source node to register itself, and notify the source node about its coding coefficient vector. It will also send a message through all its registered output interfaces to inform downstream nodes about the change.

AG: We use Auxiliary Group (AG) which has a fixed and public group address to multicast the following information to all group receivers and potential receivers:

a) Active groups and its sources in the current Automomous System (AS).

b) Sub-source addresses, sub-group addresses, coding-sub-source addresses and coding-sub-group addresses of a specific group.

c) Distribute sub-flow coefficient matrix.

ISBN: 978-1-61208-226-4

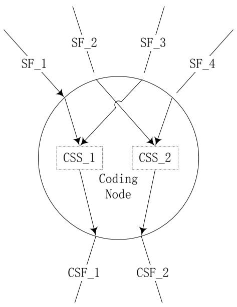


Figure 2. One physical node with two CSSs

In IPMF, we adopt a traditional Shortest Path Tree (RPT) structure as AG. Simulation results show that AG works well in doing its job.

III. FOREST IMPLEMENTATION DETAILS

A. *Coding Strategy*

IPMF uses a flow-oriented linear NC strategy. Sub-flow is the smallest unit. So coefficients are chosen for flows, not individual packets. For example, In Figure 3, if we chose α and β as coefficients for sub-flow SF_1 and SF_2, then all the packets in the coding-sub-flow *CSF* can be calculated as:

$$\mathrm{Packet}_{\mathrm{CSF}} = \alpha \times \mathrm{Packet}_{\mathrm{SF}_1} + \beta \times \mathrm{Packet}_{\mathrm{SF}_2} \qquad (1)$$

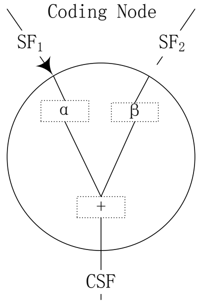


Figure 3. Flow-oriented NC

The advantages of flow-oriented NC coding are the followings:

a) Flows are easier to manage than individual packets.

b) Provide basis for routing, and decoding.

c) Reduce the size of Galois Field (GF), because less random coefficients are needed.

d) Implement "user-driven" content delivery effectively.

In IPMF, each SF including CSF is associated with a coefficient vector (CV):

$$\mathrm{CV} = [\, \mathrm{c}_1, \mathrm{c}_2 \ldots, \mathrm{c}_\mathrm{N}]^\mathrm{T} \qquad (2)$$

where N is the number of sub-flows, excluding coding-sub-flows, and $\mathrm{c_i}$, $1 \le i \le N$, is the coefficient number randomly chosen from GF (2^p). If we associate a sub-flow number from 1 to N and each sub-flow, then the CV of a sub-flow generally describes its composition. For example, in Figure 3, the CVs for SF_1, SF_2 and CSF are $\mathrm{CV}_{\mathrm{SF}_1} = [1{,}0{,}0{,}0]^\mathrm{T}$, $\mathrm{CV}_{\mathrm{SF}_2} = [0{,}1{,}0{,}0]^\mathrm{T}$ and $\mathrm{CV}_{\mathrm{CSF}} = [\alpha, \beta, 0{,}0]^\mathrm{T}$ respectively.

We use these CVs to form a coefficient matrix (CM) in the source node, then multicast the CM through auxiliary group. We assume at a certain time, there are *M* CSSs in the network.

$$CM = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & | & \alpha_{11} & \alpha_{21} & \cdots & \alpha_{M1} \\ 0 & 1 & 0 & \cdots & 0 & | & \alpha_{12} & \alpha_{22} & \cdots & \alpha_{M2} \\ 0 & 0 & 1 & \cdots & 0 & | & \alpha_{13} & \alpha_{23} & \cdots & \alpha_{M3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 & | & \alpha_{1N} & \alpha_{2N} & \cdots & \alpha_{MN} \end{pmatrix} \qquad (3)$$

The first part of the CM is a $N \times N$ unit matrix, representing N sub-flows. The second part of CM is a $N \times M$ matrix, representing M coding-sub-flows. When a group receiver collects one sub-flow, it collects one column of the CM.

When a group receiver collects N sub-flows, including CSSs, it collects a $N \times N$ matrix, called the receiving matrix (RM). If the RM is reversible, i.e., the receiver collects enough information about the original multicast flows, receivers can successfully decode and rebuild all the original information. While, due to heterogeneity, when a group receiver only collect *P* (*P*<*N*) sub-flows including coding-sub-flows, we can still decode part of the information if enough variables can be eliminated by Gaussian elimination, and if the "base sub-flow" can be decoded, then the receiver is still able to enjoy the service in a lower quality.

B. *Multicast Forest Construction*

1) *Constructing The Forest at The Beginning*

At the beginning of the multicast session, there is no CSS in the network, thus the CM initial group receivers get from the auxiliary group is simply a $N \times N$ unit matrix:

$$CM_{init} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix} \qquad (4)$$

Algorithm 1 can construct multicast forest. At the beginning, we need to state a few assumptions. Firstly, a node will operate NC, only when multiple incoming sub-flows share the same output interface, and the overall bandwidth of these incoming sub-flows exceeds the effective bandwidth of the output interface. Secondly, we assume all sub-flows have the same bandwidth, all physical links in the network have the same bandwidth, and two sub-flows cannot be transmitted through one link at the same time. The last, current version only supports *N*=2.

Algorithm 1

```
for each receiver do
```

```
    Listen to auxiliary group and update the CM.
      for each sub-flow, higher priority first do
        int $i$ = 1
        while $i$ do
        if exist unoccupied interfaces then
            Lookup the corresponding SS address in the routing table, with $i_{th}$ best path.
            if returned interface in unoccupied then
              Occupy the interface.
            else
              if $i$< number of interfaces then
                  $i$+ +; continue
              else break
              end if
          else break
          end if
        end while
      end for
      for all SFs that have occupied an interface do
        Send Source-Specific-Join with the occupied interface.
      end for
end for
```

This algorithm generally assigns a different interface for each sub-flow ordered by priority with its best effort. After join messages are sent, sub-tree collision may happen. Some nodes may transform into CSS. In this case, a solution is given in the next situation.

2) Reconstructing Forest During Run-time

The multicast session has already been activated for a while. The CM is no longer a unit matrix due to sub-tree collisions, and then:

$$CM_{rec} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & | & \alpha_{11} & \alpha_{21} & \cdots & \alpha_{M1} \\ 0 & 1 & 0 & \cdots & 0 & | & \alpha_{12} & \alpha_{22} & \cdots & \alpha_{M2} \\ 0 & 0 & 1 & \cdots & 0 & | & \alpha_{13} & \alpha_{23} & \cdots & \alpha_{M3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & | & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 & | & \alpha_{1N} & \alpha_{2N} & \cdots & \alpha_{MN} \end{pmatrix} \tag{5}$$

a) Let $CV_{CSF} = [\alpha_1, \alpha_2, \alpha_3 ..., \alpha_N]^T$.

b) Let $Prio_{id}$ *represents the priority of* SF_{id}.

c) All the non-zero elements in CV_{CSF} *form a set:* $CV'_{CSF} = \{\alpha_{i_1}, \alpha_{i_2}, \alpha_{i_s}\}$, $s \leq N$.

d) Then $\mathrm{Prio}_{\mathrm{CSF}} = \frac{\sum_{j=1}^{S} \mathrm{Pr}\,io_{i_j}}{S}$.

For example, if CV_{CSF} =[1,0,1,1,0]$^{\mathrm{T}}$, then $\mathrm{Prio}_{\mathrm{CSF}} = \frac{\mathrm{Prio}_1 + \mathrm{Pr}\,io_3 + \mathrm{Pr}\,io_4}{3}$.

Algorithm 2 reconstructs the multicast forest during run-time. This algorithm generally assigns a different interface for each sub-flow including coding-sub-flow in the reversible RM with its best effort. It also takes the priority into account.

Algorithm 2

```
for each receiver whose Join has been denied do
    Listen to auxiliary group and update the CM.
      for each SF whose Join has been denied, higher priority first do
    Pick out a CSF satisfies: The SF-id$_{th}$ element in CV is nonzero. Meaning this CSF contains information about the SF in question.
    Higher priority first.
    Send Source-Specific-Join toward the selected CSF.
    end for
end for
for each potential receiver do
    Listen to auxiliary group and update the CM.
    Pick out $N$ SFs including CSF from the CM, satisfying:
    The $N \times N$ RM constructed is reversible;
    The $N \times N$ RM constructed has the highest possible priority;
    for each SF in RM, higher priority first do
      int $i$=1
      while $i$ do
      if there are still unoccupied interfaces then
          Lookup the corresponding SS address in the routing table with $i_{th}$ best path.
          if returned interface is unoccupied then
            Occupy the interface.
          else if   $i$< number of interfaces then
                $i$ ++; continue
          else if exist SFs unselected in CM then
              Replace this SF, so that RM satisfies:
    The $N \times N$ RM constructed is reversible;
    The $N \times N$ RM constructed has the highest possible priority;
            else break
            end if
        end if
        end if
    else break
    end if
end while
end for
    for all SFs that have occupied an interface do
      Send Source-Specific-Join with the occupied interface.
    end for
end for
```

IV. SIMULATION

In this section, we present some simulation results obtained by NS2 network simulator software. In order to verify our core thoughts with low complexity, we chose the classical and simple "Butterfly Network" to run IPMF like Figure 4. IPMF emphasizes on multi-tree construction, so we choose the traditional single-tree structure provided by Protocol Independent Multicast-Sparse Mode (PIM-SM) as a comparison object.

In our simulation, there are three key aims with IPMF. Firstly, we want to test and verify the dynamic multicast forest construction algorithm of IPMF. Secondly, we want to test and verify the support for dynamic joins/prunes. The last, we want do throughput test compared with PIM-SM.

Here are details of simulation settings with NS2. The original flow is split into two SFs; node 1 and 2 are SS nodes; node 3 is a CSS node; node 5 and 6 are group receivers; all links have a bandwidth of 1Mbps; source sending rate varies from 100 Kbps to 2.4 Mbps; as to the simulation time, node 5 joins the group at 0.0s, while node 6 joins the group at 0.3s.

Here are our Simulation results with NS2:

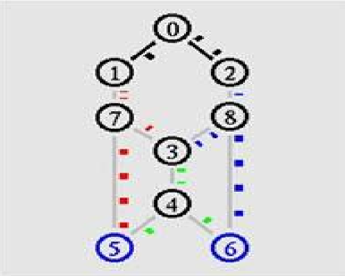


Figure 4. IPMF is well constructed in NS2

In Figure 4, at simulation time 0.3s, node 6 correctly joins the group, so we can know that the Forest multicast forest is well constructed.

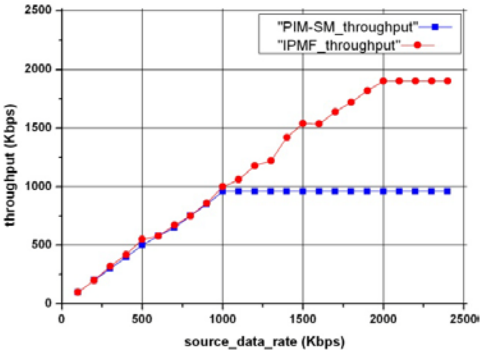


Figure 5. Throughput test

In Figure 5, the max-flow of node 5 is 2Mbps. As we change the source sending rate from 100 Kbps to 2.4 Mbps, IPMF obviously outperformed PIM-SM after the source sending rate exceeds 1Mbps.

V. Conclusions and Future Work

In this paper, we proposed Forest, a layered multi-path IP multicast protocol with network coding applied. Forest seeks the balance between performance and feasibility. We also give the IPMF construction algorithm aiming to construct a practical distribution structure and make it possible to apply network coding in a dynamic environment. Though this algorithm may fail from time to time because of its bottom-up manner, it will converge to a stable out-come by re-run. Theoretically, the more sub-flows the original multicast flow split, the better Forest performs. But more sub-flows requires more sophisticated algorithm. Simulation results show that dynamic joins/prunes are well supported by Forest. Besides, in certain situations, compared to the traditional PIM-SM protocol, Forest achieves a throughput gain up to 50%, with only two sub-flows enabled.

The current version of Forest is basically an experimental version. Optimizations will be done in the future. We will choose NS3 which has some new features compared with NS2 to implement a better and more detailed simulation. After that, we will plan to implement Forest in an embedded system to test its performance in a real network environment.

Acknowledgment

The authors would like to thank the support of National Basic Research Program of China No. 2012CB315904, NSFC No. 61179028, SZ JC201005260234A, SZ JC201104210120A, ZD201006110044A, and GDNSF No. S2011010000923.

References

[1] S. McCanne, V. Jacobson, and M. Vetterli, “Receiver-driven layered multicast,” SIGCOMM Comput. Commun. Rev, vol. 26, no. 4, pp. 117-130, 1996.

[2] B. Li and J. Liu, “Multirate video multicast over the internet: an overview,” Network, IEEE, vol. 17, no. 1, pp. 24 -29, 2003.

[3] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, “Network information flow,” IEEE Transactions on Information Theory, vol. 46, no. 4, pp. 1204-1216, 2000.

[4] S. Y. R. Li, Q. T. Sun, and Z. Shao, “Linear network coding: Theory and algorithms,” Proceedings of the IEEE, vol. 99, no. 3, pp. 372-387, 2011.

[5] N. Sundaram, P. Ramanathan, and S. Banerjee, “Multirate media stream using network coding,” in Proc. 43rd Annual Allerton Conference on Communication, Control, and Computing, 2005.

[6] M. Eros, “Universal multiresolution source codes,” IEEE Transactions on Information Theory, vol. 47, no. 6, pp. 2113 -2129, 2001.

[7] V. K. Goyal, “Multiple description coding: compression meets the network,” Signal Processing Magazine, IEEE, vol. 18, no. 5, pp. 74 -93, 2001.

[8] S. Mingkai, “Scalable Multimedia Communication Using Network Coding,” PhD thesis, McMaster University, 2011.

[9] M. Kim, D. Lucani, X. Shi, F. Zhao, and M. Medard, “Network coding for multiresolution multicast,” INFOCOM, Proceedings IEEE, pp. 1-9, 2010.

[10] M. Shao, X. Wu, and N. Sarshar, “Rainbow network flow with network coding,” Network Coding, Theory and Application, pp. 1-6, 2008.

[11] S. Mingkai, S. Dumitrescu, and W. Xiaolin, “Layered multicast with inter-layer network coding for multimedia streaming,” IEEE Transactions on Multimedia, vol. 13, no. 2, pp. 353-365, 2011.

[12] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, “Lion: Layered overlay multicast with network coding,” IEEE Transactions on Multimedia, vol. 8, no. 5, pp. 1021-1032, 2006.

[13] Z. Kiraly and E. R. Kovacs, “A network coding algorithm for multi-layered video streaming,” Network Coding, pp. 1-7, 2011.

Performance Evaluation of a Planar Layout of Data Vortex Optical Interconnection Network

Qimin Yang
Engineering Department, Harvey Mudd College
Claremont, California, USA
Qimin_yang@hmc.edu

***Abstract*-Optical interconnection networks provide solutions for high performance computing and communication networks which demand high throughput and low latency as well as high scalability. Data Vortex switching network has been studied previously for such purpose. In this paper, we focus on implementation issues, specifically on exploration of an alternative layout that allows the routing paths arrangement in such network to be implemented as multiple parallel planes. The original multiple cylinder three dimensional architecture is converted to planar structures. Since the new layout is designed to be functionally equivalent to the original Data Vortex, the routing performance in throughput and latency is shown to be very similar to that of the original network under same network operating conditions. The effect of injection at different angles is also investigated for optimized performance. The proposed planar architecture provides much more flexible configuration for multiple network levels, and the study validates that it provides comparable routing performance as in original design. Because of the flexibility of the new planar architecture, future work may explore further optimization in routing resources and performance.**

Keywords- Optical Interconnection Network; Data Vortex; Packet Switched; Routing; Planar Network.

I. INTRODUCTION

High performance multiprocessor supercomputers and multiple I/O communication systems require high throughput and low latency packet switched interconnection networks. As optical fiber technology matured with the optical communication industry, there are more research efforts recently dedicated to developing optically implemented interconnection networks for packet switched operation [1-2]. Considering the tremendous amount of routing decisions such networks have to make, recent researches in silicon photonics devices and platforms are important technology developments for seamless integration of the two domains [3-4]. Instead of converting existing electronic interconnection networks, the renewed interests in the area also bring in more innovative designs in network architectures that can efficiently utilize both electrical and optical domains. Photonic approaches are attractive solutions because they can not only easily achieve the bandwidth requirement, but also provide more promising potential in power consumption and scalability challenges that current electronic interconnection networks encounter [5-6]. A good network design should utilize the broad bandwidth of optical domain while avoiding extensive logic processing or optical buffering due to the lack of the mature technology.

Data Vortex network architecture is designed for localized high performance interconnection purpose, and it is scalable to a very large number of communication ports operated in packet switched mode [7-9]. With reasonable expense in routing resource redundancy, it is able to achieve very high traffic throughput while maintaining low latency and narrow latency distribution, both of which are extremely important for guarantee of packet’s signal quality at the physical layer [10-11]. Data Vortex optical interconnection network relies on a three dimensional cylindrical topology to efficiently move the data flow from input to output ports [12-14]. Considering the benefit of planar structure for physical implementation, this work focuses on converting the three dimensional topology to multiple planes of routing levels to facilitate construction and integration.

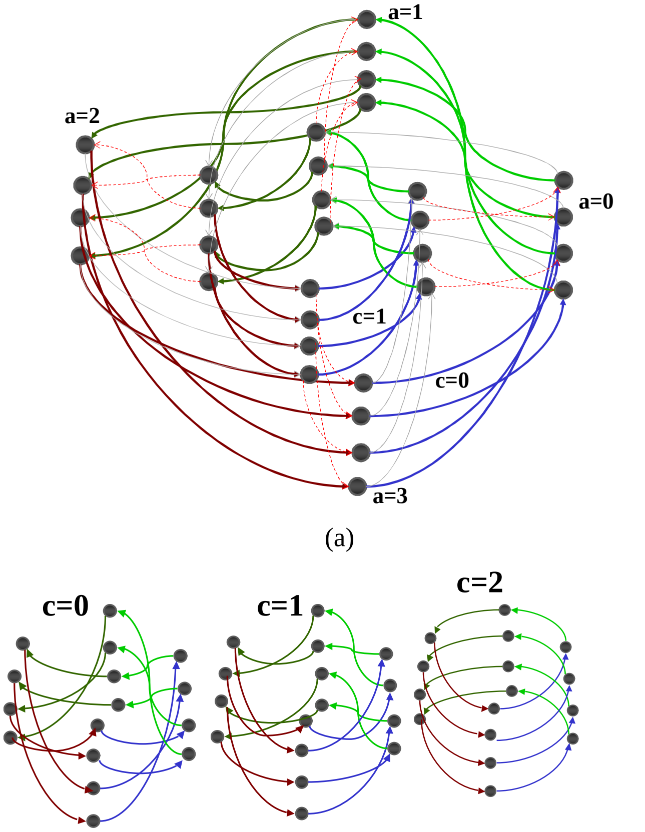
The rest of paper is organized as follows: Section II explains the original Data Vortex switching topology, and difficulty with large scale system construction in cylinders. Section III presents the proposed planar layout of each cylinder level which allows for an equivalent routing topology. Section IV presents the routing performance comparison with the original layout and confirms the feasibility of the proposed system, and Section V concludes the study and discuss future works relevant to the area.

II. ORIGINAL DATA VORTEX DESIGN

The original Data Vortex network uses a cylindrical layout with multiple cylinders of routing nodes along angle and height dimensions. As an example, Figure 1 (a) shows the two outer cylinders’ routing paths in a network of A=4 angles and H=4 in height. To allow for clear view, intra-cylinder paths patterns are also individually shown in Figure 1(b) for each of the three cylinders. The number of

ISBN: 978-1-61208-226-4

cylinders required is given by $\log_2 H + 1$ due to the binary decoding nature of the routing process. The additional last cylinder is typically added, as shown in cylinder c=2 (=C) in Figure 1 (b), and it maintains the height position; but, it allows for angular resolution if each angle connects to a different sets of I/O ports. In addition, the last cylinder provides optical buffering in addition to the electrical buffers situated at I/O ports.



(a)

(b)

Figure 1. (a) Two outer cylinders of Data Vortex topology for A=4, H=4. (b) Intra-cylinder routing paths for each of the cylinders.

The number of active angles A_{in} are connected to I/O ports, so that the ratio A_{in}/A controls the redundancy in network operation. The choice of A_{in} needs to balance between the support of I/O ports for the given network cost and the routing performance. The smaller A_{in}/A results in better routing performance, but also means supporting of smaller I/O ports or more expensive implementation as the required number of routing nodes and optical switches is proportional to the total number of angle A. The typical choice of A is small number around 5 because an ideal operation with $A_{in}/A=1/5$ results in the optimum routing performance [7]. A much larger angle introduces much longer delay due to the latency associated with the angular resolution at the last cylinder.

The routing process starts with packets injected at the outermost cylinder, and after through each of the cylinders exit to output ports at the innermost cylinder. As shown in Figure 1, at the specific cylinder, the semi-twisted routing path patterns repeat from angle to angle which forms a cylinder by connecting the last angle to the first angle, allowing the packet to switch between two groups of height where the corresponding binary bit of the height position flip back and forth between “1” and “0”. This not only allows the packets to quickly reach the correct height group, but also provide multiple open paths to reach their destination. Once the position group matches the desired group at the specific cylinder, the packet is forwarded to the inner cylinder by inter-cylinder paths (gray lines shown) that simply maintain the current height position. Such routing process continues until the packet reaches the innermost cylinder and exits the network. Another important design of the network is to guarantee single packet routing for each routing node through a traffic control mechanism. This greatly simplifies the node implementation and routing decisions, where electronic processing will not impose serious speed limitation. These traffic control signals are distributed throughout the network, and they are used between a pair of relevant nodes, sent from inner cylinder nodes to inform their outer cylinder neighbors its traffic condition. In case both have packets arriving, the inner node is always given higher priority and the outer cylinder traffic is deflected by staying at the outer cylinder instead. These control lines are shown as dash lines between the pair of nodes in Figure 1.

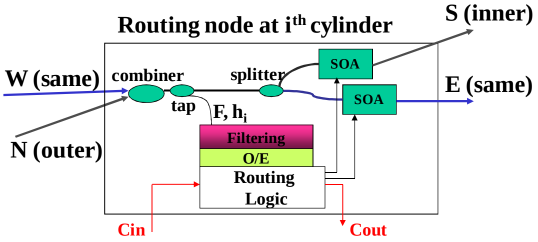


Figure 2. Routing node photonic implementation

For the physical implementation, the switches within the routing node are based on semiconductor optical amplifier (SOA). It not only allows for fast switching required for packet switching, but also provide broadband operation which allows for utilization of wavelength division multiplexing (WDM) techniques. Specifically, payload data is modulated onto WDM channels to keep short packet length while maintaining high data bandwidth, and header bits that contain the destination information are also modulated onto different wavelength channels for simple decoding within the routing nodes. In addition, any power loss in routing nodes can be easily compensated by the gain provided by semiconductor optical amplifier [8].

A detailed routing node implementation using optical components is shown in Figure 2. There are two input paths, where N (North) connects to the outer cylinder node, and W (West) connects to the same cylinder node. Since the control mechanism mentioned above guarantees that at most one packet enters the node, the input paths from N and W are combined before the header bit extraction and decoding. The binary header bits of the target address are encoded using distinct wavelength channels, therefore simple passive filters and low packet rate optoelectronic (O/E) detector are used to extract such information from the optical packet. The single packet is then split to two potential output paths, one to S (South) to inner cylinder or one to E (East) to the same cylinder. The routing decision not only examine the header bit and correct height group, but also depend on an input control signal C_{in} so that ensure it only enters the inner cylinder node when there is no potential conflict for single packet condition. Similarly, the routing logic generates a new control signal C_{out} for its outer cylinder node for the same purpose. As a result of control mechanism, packet deflection can happen in Data Vortex network without packet loss. Packets that are deflected to stay on the outer cylinder can take advantage of the multiple open paths, and this causes a rather small latency penalty in comparison to other existing networks. SOA switches are used for their fast sub-nanoseconds switching speeds and internal gain for power compensation occurring at taps and splitters. More details of signaling and path connections can be found in the references [7][9].

Previous researches on Data Vortex network have been mainly focused on network's routing performance and physical layer system performances. As more applications call for such optical interconnection networks, making them more flexible and easier for construction are of great importance. So far only a 12x12 small testbed has been built mainly because it heavily relies on discrete components. On one hand, recent researches in either SOA switching fabric or novel silicon photonic devices provide more potential for integration at the device and routing node level. On the other hand, network architectures may not have the same upgrades for easier implementation. For example, a small scale Data Vortex can be easily implemented using fiber waveguides and individual node modules in three dimensions. But, such arrangement could be very complex and cumbersome for a much larger network. The difficulties also include keeping every level aligned and synchronized necessary for the routing operation. For all paths physically the same length at all levels, inner levels must somehow wind up paths to occupy a smaller physical space than its outer level in cylindrical arrangement. In addition, due to close coupling of the electrical layer for routing logic and traffic control with the optical layer, a fully three dimensional fabrication is not compatible to integration solutions. One solution is if each of the routing levels or cylinders can be integrated as a subsystem on a plane as the electrical circuits naturally arrange on planes, it becomes a much more manageable overall system which simply interconnects the planes of subsystems. The complexity would not grow drastically as the size of the system. Because connections between the levels are parallel links, either fiber based on other type of waveguides in more integrated form can be used.

III. PLANAR LAYOUT DESIGN

To eliminate the incompatibility of integration of electrical layer in the cylindrical arrangement, and make Data Vortex easier to construct in large scale networks, this study explores an alternative layout of the Data Vortex architecture to allow for planar construction of the multiple routing levels.

In the logical level, we want to maintain the same principle of minimizing deflection probability by arranging the same semi-twisted routing patterns, while allowing for parallel planes for easy layered integration. To achieve this, instead of connecting the last angle of routing nodes to the first angle in the original Data Vortex, we added paths along the same angle at the first and the last angle (green paths as shown in Figure 3 and Figure 4), while changing half of the routing paths in the opposite traveling direction (red paths vs. blue paths). As a result, traveling on the same plane now forms a looping pattern similar to the ones by staying on the same cylinder in the original Data Vortex network. As examples, Figure 3 and Figure 4 show a network with H=4 at the first two routing levels (in comparison to the two outermost cylinders in the original network layout in Figure 1) for A=4 and A=5, respectively. As shown, the direction of the same angle green paths at the first and last angle depends on whether the angle A is even or odd, and the connection pattern follows the same pattern between the nodes as in between angles. The same cylinder paths' direction also depends on whether A is even or odd accordingly as shown.

The new layout now allows for parallel planes of different routing levels that correspond to the original cylinders. Between different planes, only parallel routing paths are needed as that in the original Data Vortex networks. The control signal paths are not shown for a clear view, but they should be set up similarly between the pair of nodes from inner cylinder to the outer cylinder whenever two nodes try to send packets to the same node. These control paths can be integrated with the optical routing path on another plane that would be perpendicular to the planes of routing levels. The control signals apply to the edge angle nodes in a similar fashion even though the output node could be located on the same angle. The detailed organization of nodal circuits on the plane is beyond this study. Either all nodes of the same plane are fabricated on a same platform/board which requires planar optical waveguide technology, or nodes of the same angle can be fabricated on a same board if angles are interconnected by more flexible fiber waveguides. Three dimensional arrangements are still open, but without the difficulty of occupying a different size space as in cylindrical networks.

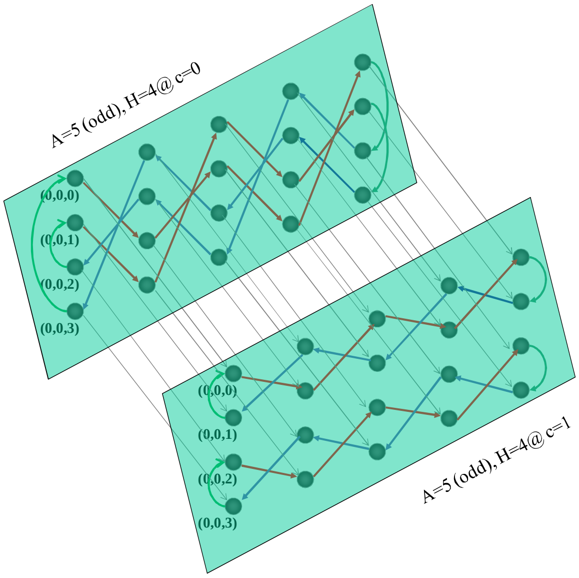


Figure 3. Routing paths in planar layout for A=5 (odd) and H=4 at the first two levels

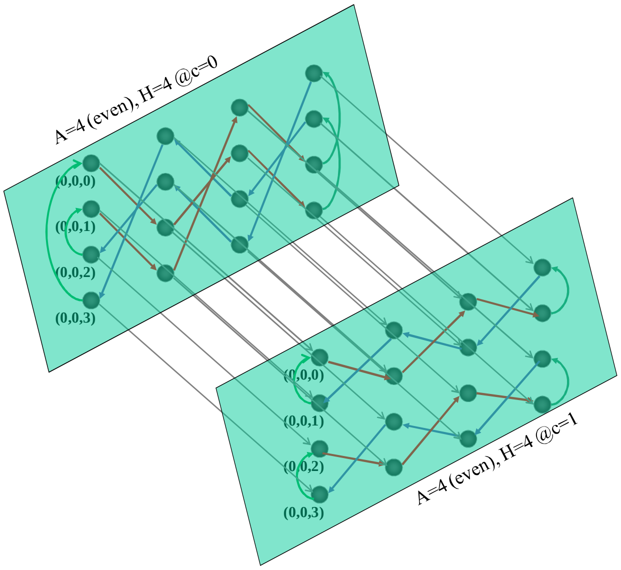


Figure 4. Routing paths in planar layout for A=4 (even) and H=4 at the first two levels

Logically, the new layout forms a very similar connection as that of the original Data Vortex network, and the rest designs such as parallel inter-level forwarding paths, control mechanism and additional last cylinder will all maintain the same. Therefore, we should expect very similar routing performance when comparing the new layout to the original Data Vortex cylindrical layout. On the other hand, the new same angle paths may affect the routing performance because nodes at different angles may carry slightly different traffic load and may result in different traffic distribution within the network. Packet injection at different angle should also be investigated with further details to make sure the planar layout can achieve the same routing performance as desired.

IV. PERFORMANCE EVALUATION

A custom written C/C++ event simulator is used to evaluate the new layout Data Vortex architecture. Same network size is chosen for two layouts while different traffic load and network redundancy are included in the study to ensure the performance comparison at various operation conditions. We choose the network angle to be A=5 as in earlier studies. A less redundant condition with increased $A_{in}>1$ for the same given A is also studied [7]. The delay performance examines the average number of hops packets experience in the network over a long period of simulation time after a steady state is reached. The throughput performance is presented by the successful injection rate at the input ports over the same simulation period. Because it is a non-blocking network, the successful injection rate reflects the overall data capacity that the network can handle. A more congested or saturated network essentially deflects more traffic at outer levels and creates traffic backpressure up to the injection ports. We always choose the same angle parameter for comparison to the original network layout, so angular resolution is not included for this study.

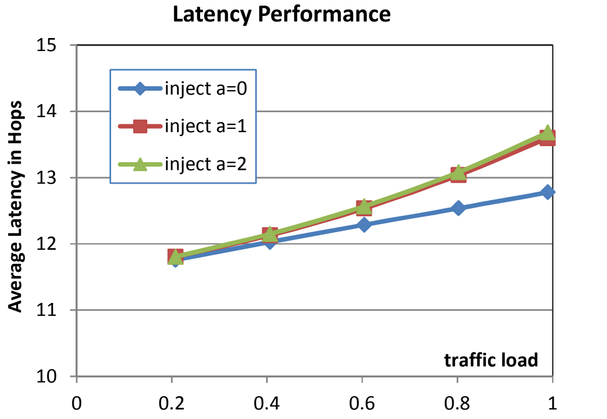


Figure 5. Latency Performance Comparison for injection at different angles for A=5, $A_{in}=1$, H=128

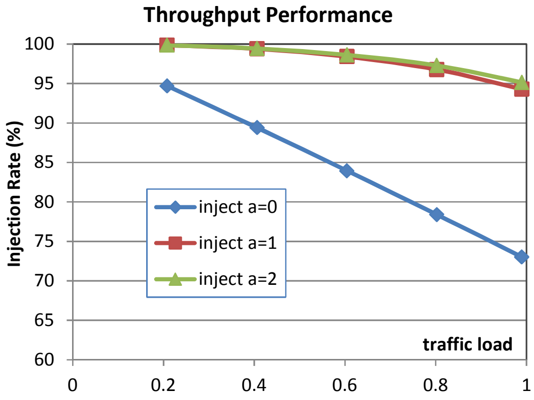


Figure 6. Throughput Performance Comparison for injection at different angles for A=5, Ain=1, H=128

First, we study the angular dependence of injection port in the new planar layout. Due to the symmetry of the layout, in a network of A=5, we only need to compare routing performance with injection occur at a=0 (end angle), a=1 and a=2 (middle angle) respectively. Figure 5 and Figure 6 present the latency and throughput performance for A=5 and H=128 with one injection angle A_{in}=1 at the specified angle a. The ratio of A_{in}/A is chosen for the optimized throughput performance as suggested in previous study. As shown, we should avoid inject at the end angles where we loop the packets back in the opposite direction on the same angle because the injection rate and throughput performance is shown to be significantly lower. The performance difference between the two middle angles however is shown to be negligible. More case studies with multiple injection angles also show similar results, which indicates that edge angles should be generally avoided if possible. The middle angle or angles close to the middle should be preferred to achieve the best overall performance in throughput and latency in the new layout. In case the network operates in much less redundant condition such as injecting at all A angles, then the edge angles have to be used as well.

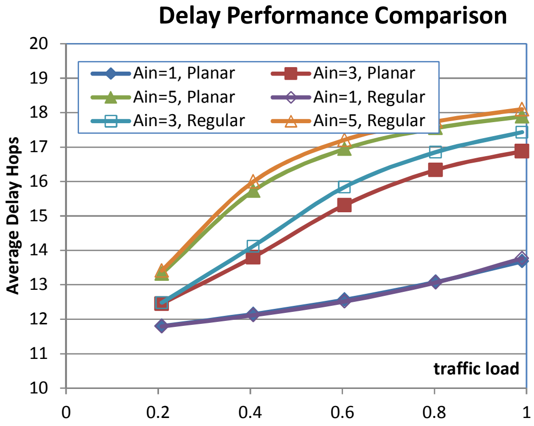


Figure 7. Delay performance comparison for two layouts for different A_{in} with A=5, H=128

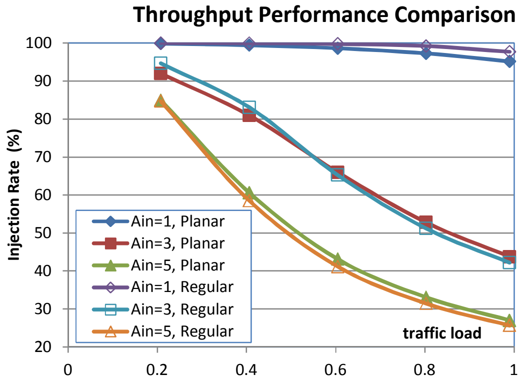


Figure 8. Throughput comparison for two layouts for different A_{in} with A=5, H=128

Next, we compare the planar layout performance with the original Data Vortex network as shown in Figure 7 and Figure 8, respectively. Here, all planar layouts use middle angles for injection angles to avoid unnecessary performance degradation, while A_{in}=5 case all angles including edge angles will be used for injection. The planar layout results are marked by solid shapes while regular layout results use hollow shapes for all three redundant operations. As shown, the routing performances in all cases are very close between the two layouts. We also notice that for Ain=1, the regular network achieves a little better throughput, while the planar layout achieves slight benefit over regular network in less redundant network conditions, such as Ain=3 and Ain=5. However, overall, there is very small difference between the two layouts as long as the injections are carried out through the middle angles.

Finally, to show the performance evaluation is valid for all network sizes, we also examine the comparison for different network heights. It has been confirmed that only small discrepancies have been observed between the two layouts and very similar trends are seen, as shown in Figure 7 and Figure 8 for different network sizes.

V. CONCLUSION

To summarize, an alternative planar layout of the Data Vortex architecture was proposed to facilitate the three dimensional integration of the network nodes. This is accomplished by dividing routing paths along the cylinder to two different directions, while adding same angle paths in the first and last angle to form similar routing loops. As a result, all nodes on the same routing level can be implemented on a plane, and overall architecture appears to be multiple planes interconnected. The optical paths and control links between routing levels along the same angle can further be put on a plane perpendicular to the routing level planes, and easily form a three dimensional structure that facilitates the modular design and synchronization of the overall network. The routing performance is specifically evaluated and in comparison to the original network performance under various network conditions and network sizes. The detailed analysis has shown that the new planar layout achieves very similar routing performance as that of original Data Vortex network, and, in general, edge angle should be avoided for packet injection for optimum routing performance.

REFERENCES

[1] A. Wonfor, H.Wang, R.V.Penty, and I.H. White, "Large Port Count High Speed Optical Switch Fabric for Use Within Datacenters", *Journal of Optical Communications and Networks*, Vol. 3, No. 8, pp. A32-39, 2011.

[2] Shinji Nishimura, Kazunori Shinoda, Yong Lee, Fumio Yuki, Takashi Takemoto, Hiroki Yamashita, Shinji Tsuji, Masaaki Nido, Masahiko Namiwaka, Taro Kaneko, Kazuhiko Kurata, Shigeyuki Yanagimachi, and Naoya Ikeda, "Optical Interconnection for High-speed Routers", *Optical Fiber Conference*, OThH2, 2011.

[3] T.-Y. Liow, K. W. Ang, Q. Fang, M. B. Yu, F. F. Ren, S. Y. Zhu, J. Zhang, J. W. Ng, J. F. Song, Y. Z. Xiong, G. Q. Lo, and Dim-Lee Kwong, "Silicon Photonics Technologies for Monolithic Electronic-Photonic Integrated Circuit Applications", *Optical Fiber Conference*, OThV1, 2011.

[4] Noam Ophir1, Kishore Padmaraju1, Aleksandr Biberman1, Long Chen2, Kyle Preston2, Michal Lipson2, and Keren Bergman, "First Demonstration of Error-Free Operation of a Full Silicon On-Chip Photonic Link", *Optical Fiber Conference*, OWZ3, 2011.

[5] R.S.Tucker, "Green optical communications-part II: energy limitations in networks", *IEEE Journal of Selected Topics in Quantum Electronics*, Vol. 17, No. 2, pp 261-274, 2011.

[6] Odile Liboiron-Ladouceur, Pier Giorgio Raponi, Nicola Andriolli, Isabella Cerutti, Mohammed Shafiqul Hai, and Piero Castoldi, "Scalable Space-Time Multi-plane Optical Interconnection Network Using Energy-Efficient Enabling Technologies", *Journal of Optical Communication and Networks*, Vol. 3, No. 8, pp. A1-A11, 2011.

[7] Q. Yang, K. Bergman, G. D. Hughes, and F. G. Johnson, "WDM packet routing for high-capacity data networks," *J. Lightw. Technol.*, vol. 19, pp. 1420–1426, Oct. 2001.

[8] C. Hawkins, B. A. Small, D. S. Wills, and K. Bergman, "The Data Vortex, an all optical path multicomputer interconnection network," *IEEE Transactions of Parallel and Distributed Systems*, vol. 18, No. 3, pp. 409-420, Mar 2007.

[9] Cory Hawkins, D. Scott Wills, Odile Liboriron-Ladouceur, and Keren Bergman, " Hiearchical clustering of the data vortex optical interconnection network", *Journal of Optical Networking*, Vol.6, No. 9, pp. 1179-1190, Sep. 2007.

[10] O.Liboiron-Ladouceur, B.A.Small and K.Bergman, "Physical Layer Scalability of WDM Optical Packet Interconnection Networks", *J. Lightwave Technology*, vol. 24, pp. 262-270, 2006.

[11] A. Shacham, B.A. Small, O. Liboiron-Ladouceur and K. Bergman, "A Fully Implemented 12x12 Data Vortex Optical Packet Switching Interconnection Network," *Journal of Lightwave Technology*, vol. 23, No. 10, pp. 3066-3075, 2005.

[12] Neha Sharma, D. Chadha, Vinod Chandra, "The augumented data vortex switch fabric: an all-optical packet switched interconnection network with enhanced fault tolerance", *Journal of Optical Switching and Networking*, Vol. 4, pp. 92-105, 2007.

[13] A. Shacham and K. Bergman, "Optimizing the performance of a data vortex interconnection network," *Journal. Optical Networking*, vol. 6, No. 4, pp. 369-374, April 2007.

[14] A. Shacham and K. Bergman, "On contention resolution in the data vortex optical interconnection network", *Journal of Optical Networking*, vol. 6, No. 6, pp. 777–788, June 2007.

[15] http://lightwave.ee.columbia.edu/downloads/acs_workshop/Columbia_GaTech_ACS_April16_2010_hw2.pdf (Last retrieved September 2012)

An Efficient Scheduling in Consideration of the Signaling Overhead for Relay Networks

Deokhui Lee and Jaewoo So
Department of Electronic Engineering
Sogang University
Seoul 121-742, Republic of Korea
Email: {akirain, jwso}@sogang.ac.kr

***Abstract*—An efficient scheduling for relay networks is proposed in consideration of the signaling overhead. A source node informs both relay and destination node about the resource assignments in every frame. The resource allocation process generates a substantial signaling overhead, which influences the system performance. However, the amount of the signaling overhead can be reduced by predetermining resource assignments of future frames. We develop a frame structure for scheduling and propose an efficient scheduling in consideration of the signaling overhead. The performance of the proposed scheduling is evaluated compared with that of the conventional scheduling in terms of the average capacity as both the number of relays increases and the average signal-to-noise ratio increases. Simulation results show that the average capacity of the proposed scheduling is greater than that of the conventional scheduling.**

Keywords-relay networks; persistent scheduling; signaling overhead.

I. Introduction

In the cellular networks, the relay node is one of the important techniques to improve the spectrum efficiency, link reliability, and coverage. Besides, an efficient scheduling algorithm is crucial for the efficient use of limited wireless resource. A scheduler in the relay networks determines a relay node which relays data from the source node to the destination node. Therefore, an optimal selection criterion of the relay node is dependent on the scheduling metric. A selection of the relay node is important to increase the system capacity. A signaling message usually broadcasts to users for every frame. The term signaling overhead is used to describe the information on the resource assignments and the path management. However, frequent route changes could cause high signaling overheads which influence the system performance [1]. For example, the amount of effective resources to transmit data traffic increases as the signaling overhead decreases. Hence, one of the important roles of the scheduler is to reduce the signaling overhead.

Many researchers have endeavored to develop an efficient scheduling algorithm to increase the system performance in the relay networks [2]-[5]. Lee and Hwang [2] proposed a scheduling algorithm to reduce the power, where a relay node is selected on the basis of channel conditions of relay nodes. Hence, the selected relay node meets the data rate requirement. In [3] and [4], authors propose a relay node selection algorithm in order to enhance the system performance. In [3], the optimal relay node selection strategies are proposed under fixed and variable transmit power. In [4], a centralized utility maximization frame work was introduced for relay networks. The physical layer transmission strategies are done efficiently by optimizing pricing variables as weighting factors. In [5], the relay selection strategies are proposed and evaluated in terms of spectral efficiency in the relay cellular network. To determine an optimal route for a data transmission, the scheduler takes into account the feature of multi-hop transmission.

However, the scheduling algorithms of [2]-[5] were not considered the signaling overhead that broadcasts information on resource assignments and path management. The resource allocation process generates a substantial signaling overhead, which influences on the amount of the available radio resources. Hence, the scheduler should consider the signaling overhead to increase the system capacity. Some studies have considered the signaling overhead in the scheduling metric for relay systems [6]-[7]. In [6] and [7], by taking the signaling overheads transmitted among relay nodes into consideration, the system capacity was evaluated. Bletas, Khisti, Reed, and Lippman [8] proposed a distributed relay selection algorithm for a two-hop amplified-and-forward system, where the selection criterion is to select the relay node which has the best instantaneous signal-to-noise ratio (SNR) across the two-hops. Although these studies consider the signaling overhead, the scheduling metric has no consideration to reduce the signaling overhead. The relay selection strategy was proposed in [9], which introduced a method to reduce the complexity and the signaling overhead of the relaying process. The feedback message for the channel state report is considered but the signaling message is not considered. In [10], the framework has been proposed for the relay selection in cellular networks while limiting the feedback overhead and complexity. However, the signaling message for data transmissions is not considered in the relay selection criterion.

This paper proposes an efficient scheduling in considera-

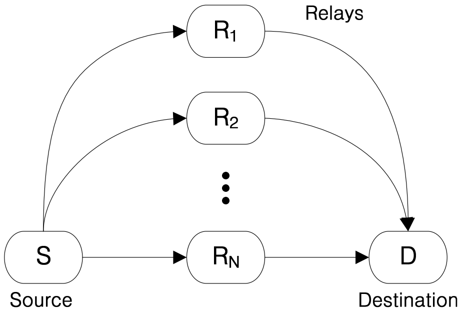


Figure 1. The multi-relay network

tion of signaling overhead for relay networks.

The remaining part of this paper is organized as follows: Section II contains the description of the system model. In Section III, we propose an efficient scheduling algorithm in consideration of the signaling overhead and analyze the system capacity of the proposed scheduling. We provide simulation results in Section IV and finally, Section V presents conclusions and future works.

II. System Model

We consider the downlink relay networks, consisting of a single source, N relays, and a single destination denoted by S, R, and D, respectivley. The system investigated in this paper is shown in Fig. 1. We assume that Rayleigh fading channels [11] between the source and relays, denoted by $S-R$ links, and the Rayleigh fading channels between the relays and the destination, denoted by $R-D$ links. Moreover, independent and identically distributed Rayleigh fading is assumed across all links. It is assumed that the entire links have the same average SNR. The source has no direct link with the destination and the data transmission is performed by one of N relays. The source broadcasts the signaling message in order to inform on resource assignments to both relays and the destination. We assume that the signaling message is only transmitted in S-R link and is not transmitted in R-D link. Therefore, the signaling message includes information on both S-R link and R-D link.

Let $\gamma_{i,j}$ denote the instantaneous SNR of i-j link, where $i = \{S, 1, \cdots, N\}$ and $j = \{1, \cdots, N, D\}$. For Rayleigh fading channels, $\gamma_{i,j}$ is an exponential random variable, and its cumulative density function (CDF), $F(\gamma_{i,j})$, and probability density function (PDF), $f(\gamma_{i,j})$, are given as

$$F(\gamma_{i,j}) = 1 - e^{-\frac{\gamma_{i,j}}{\bar{\gamma}}}, \quad (1)$$

$$f(\gamma_{i,j}) = \frac{1}{\bar{\gamma}} e^{-\frac{\gamma_{i,j}}{\bar{\gamma}}}, \quad (2)$$

where $\bar{\gamma}$ is an average SNR of i-j link.

The relay selection is performed as follows:

Step 1) For each relay link, determine a minimum SNR between $\gamma_{S,n}$ and $\gamma_{n,D}$:

$$\Omega_n = \min_{n=1,2,\cdots,N} \{\gamma_{S,n}, \gamma_{n,D}\}, \quad (3)$$

where Ω_n is the instantaneous SNR when data is transmitted by the relay n. The CDF, $F_{\Omega_n}(\Omega_n)$ and PDF, $f_{\Omega_n}(\Omega_n)$ can be formulated as

$$\begin{aligned} F_{\Omega_n}(\Omega_n) &= 1 - \Pr(\gamma_{S,n} > \Omega_n) \Pr(\gamma_{n,D} > \Omega_n) \\ &= 1 - e^{-\frac{\Omega_n}{\bar{\gamma}}} e^{-\frac{\Omega_n}{\bar{\gamma}}} \\ &= 1 - e^{-\frac{2\Omega_n}{\bar{\gamma}}}, \quad (4) \\ f_{\Omega_n}(\Omega_n) &= \frac{dF_{\Omega_n}(\Omega_n)}{d\Omega_n} = \frac{2}{\bar{\gamma}} e^{-\frac{2\Omega_n}{\bar{\gamma}}}. \quad (5) \end{aligned}$$

Step 2) The source node selects a relay for data transmissions by following as:

$$\gamma_{n^*} = \max_{n=1,2,\cdots,N} \{\Omega_n\}. \quad (6)$$

We define a minimum SNR of the selected relay as γ, i.e., $\gamma = \gamma_{n^*}$. The parameter, n^*, means a selected relay. The PDF, $f_\gamma(\gamma)$, can be derived by using a knowledge of order statistics [12], [13]. Using both (4) and (5), the PDF of γ can be formulated as

$$\begin{aligned} F_\gamma(\gamma) &= \prod_{n=1}^{N} \Pr(\Omega_n \leq \gamma) \\ &= \left(1 - e^{-\frac{2\gamma}{\bar{\gamma}}}\right)^N, \quad (7) \\ f_\gamma(\gamma) &= \frac{dF_\gamma(\gamma)}{d\gamma} = N \frac{2}{\bar{\gamma}} e^{-\frac{2\gamma}{\bar{\gamma}}} \left(1 - e^{-\frac{2\gamma}{\bar{\gamma}}}\right)^{N-1}. \quad (8) \end{aligned}$$

III. Performance Analysis

A. Conventional Scheduling

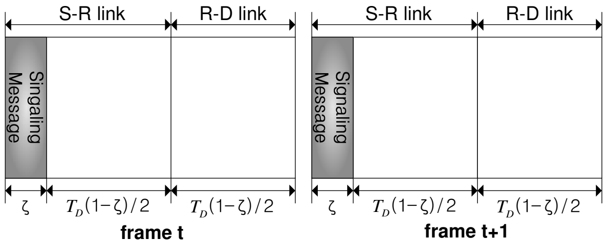


Figure 2. The frame structure of the conventional scheduling

In the conventional scheduling, the source broadcasts the signaling message for every frame to inform the allocations of radio resources in the downlink relay networks. The signaling message contains the information that indicates the path management and resource assignments. For conventional scheduling, we propose a modified frame structure, as shown in Fig. 2.

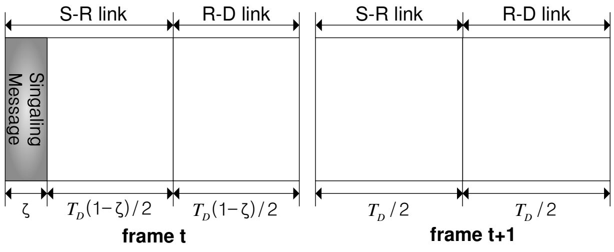


Figure 3. The frame structure of the proposed scheduling

Let ζ denote the ratio of the signaling message used to transmit information to both relays and the destination. The ratio, ζ, is defined as follows:

$$\zeta = \frac{\text{Resource used to transmit a signaling message}}{\text{Resource used to transmit data}}. \quad (9)$$

Let T_D be the frame duration. Then, $T_D(1-\zeta)/2$ denotes the duration for data transmission at S-R and R-D link, respectively. The duration of a subframe for S-R link, $\zeta + T_D(1-\zeta)/2$, is different from the duration of a subframe for R-D link, $T_D(1-\zeta)/2$. The frame structure is not symmetric between the S-R link and the R-D link. However, because the duration for data transmissions at the S-R link and at the R-D link is identical, the frame structure provides the equal opportunity for each link in view of the data transmission. The performance of conventional scheduling is evaluated in terms of the average capacity, where the capacity, C, is calculated by Shannon's capacity as follows:

$$C = \log_2(1+\gamma) \text{ [bps/Hz]}, \quad (10)$$

where γ is the SNR [14]. The average capacity of the conventional scheduling can then be rewritten by using the PDF of γ obtained from (8). The average capacity can be obtained as follows:

$$\begin{aligned}\bar{C} &= \frac{T_D(1-\zeta)/2}{T_D}\int_0^\infty \log_2(1+\gamma) f_\gamma(\gamma)d\gamma \\ &= \frac{1-\zeta}{2}\int_0^\infty \log_2(1+\gamma) f_\gamma(\gamma)d\gamma \text{ [bps/Hz]}, (11)\end{aligned}$$

where the factor of $1/2$ is used because the frame is divided into two subframes, and the factor of $(1-\zeta)$ is used because the signaling overhead is reduced.

B. Proposed Scheduling

To increase the system performance, the source performs an efficient scheduling taking the signaling message into consideration. In the conventional scheduling, the signaling message is transmitted in every frame. However, the resource allocation process generates a substantial signaling overhead, which influences the system performance.

In the fading channel environments, the instantaneous SNR is somewhat predictable. Hence, the scheduler reduces the signaling overhead by transmitting an initial assignment message, which is valid in a following of future frames. In the proposed scheduling, as shown in Fig. 3, the source broadcasts information on resource assignments in the signaling message only for frame t and does not broadcast the signaling message for frame $t+1$. The source allocates a persistent resource to both relay and destination when it first schedules the both relay and destination; and the allocated resource is valid in frame $t+1$. Hence, the signaling overhead decreases and the effective downlink resource increases. In the proposed scheduling, if the instantaneous SNR in frame $t+1$ is equal or greater than the instantaneous SNR in frame t, the proposed scheduling may result in some efficiency because the source transmits data without notification of signaling messages. However, if the signaling message which is predetermined in frame t is not suitable for frame $t+1$, the performance degradation will occur in frame $t+1$.

The instantaneous SNR may vary in every frame in accordance with the time-varying channel conditions. The probability that the highest SNR is greater at frame $t+1$ than at frame t, is defined as

$$P = \Pr(\gamma_1 \leq \gamma_2), \quad (12)$$

where γ_1 and γ_2 are the highest SNR at frame t and at frame $t+1$, respectively.

The average capacity of the proposed scheduling is twofold. First, when the source broadcasts the signaling message, the average capacity for frame t, which is denoted by C_1, can be written as the total capacity averaged over the PDF, $f_\gamma(\gamma)$, as follows:

$$\bar{C}_1 = (1-\zeta)\frac{1}{2}\int_0^\infty \log_2(1+\gamma) f_\gamma(\gamma)d\gamma \text{ [bps/Hz]}. \quad (13)$$

The average capacity, C_1, is equal to the average capacity of the conventional scheduling because the source broadcasts the signaling message to both relays and the destination.

Second, when the source transmits data without the notification of the signaling messages, the average capacity for frame $t+1$, which is denoted by C_2, can be written as the total capacity averaged over PDF, $f_\gamma(\gamma)$, as follows:

$$\begin{aligned}\bar{C}_P &= P\cdot\frac{1}{2}\int_0^\infty \log_2(1+\gamma) f_\gamma(\gamma)d\gamma, && (14)\\ \bar{C}_{1-P} &= (1-P)\cdot\frac{1}{2} \\ &\quad \times \int_0^\infty \log_2(1+(1-\alpha)\gamma) f_\gamma(\gamma)d\gamma, && (15)\\ \bar{C}_2 &= \bar{C}_P + \bar{C}_{1-P} \quad \text{[bps/Hz]}, && (16)\end{aligned}$$

where the $\bar{C}_P$ and $\bar{C}_{1-P}$ are the average capacity when $\gamma_1 \leq \gamma_2$ and $\gamma_1 > \gamma_2$, respectively. The SNR variation parameter, α, is a ratio of amount of a falling SNR value when $\gamma_1 > \gamma_2$. The average capacity for frame $t+1$ is determined by the sum of $\bar{C}_P$ and $\bar{C}_{1-P}$.

Table I
SIMULATION PARAMETERS

Parameter	Value	
	Default	Variation
The number of relays, N	5	1∼10
The average SNR of S-R link, γ^{S-R}	10 dB	0∼14 dB
The average SNR of R-D link, γ^{R-D}	10 dB	0∼14 dB
The signaling overhead ratio, ζ	20%	10%, 30%
The SNR decrease ratio, α	20%	-
The probability, P	80%	60%, 80%, 100%

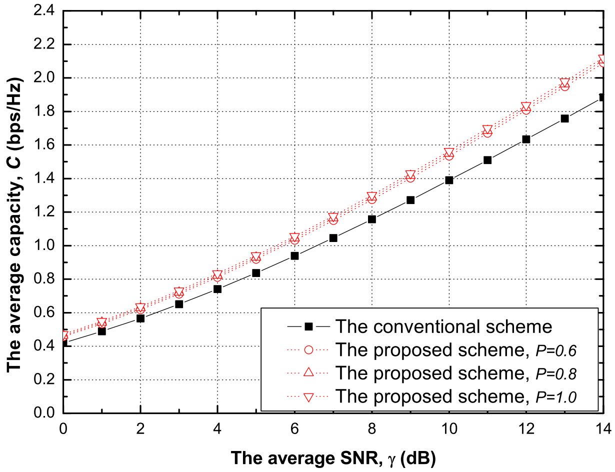


Figure 4. The average capacity versus the average SNR, γ (dB)

The average capacity of the proposed scheduling can be obtained by

$$\bar{C} = \frac{1}{2}\left(\bar{C}_1 + \bar{C}_2\right) \text{ [bps/Hz]}. \qquad (17)$$

IV. SIMULATION RESULTS

Simulation environments assume the downlink of a relay network with N active relays, in which all relay links have the same average SNR, $\bar{\gamma}$. Each link of channels has a Rayleigh channel environment of $f_{\gamma_{i,j}}(\gamma_{i,j}) = 1/\bar{\gamma}\exp(-\gamma_{i,j}/\bar{\gamma})$, where $\gamma_{i,j}$ is the instantaneous SNR from i node to j node where $i = \{S, 1, \cdots, J\}$ and $j = \{1, \cdots, J, D\}$. The ratio of the signaling overhead is assumed to be $\zeta = 0.2$. The probability of $\gamma_1 \leq \gamma_2$ is assumed to be $P = 0.8$. We perform the simulation according to the value of P because the channel environment can be experienced slow fading or fast fading. The other parameters used in our simulation are described in Table 1. The performance of the proposed scheduling is evaluated in terms of the average capacity.

Figure 4 shows the average capacity when $P = 0.6, 0.8$ and 1.0. The average capacity increases as the value of the average SNR increases. Because the conventional scheduling uses the signaling message to allocate resource in every frame, the system capacity decreases as the amount of the signaling overhead increases, regardless of the probability, P. However, the proposed scheduling predetermines the information on the resource assignments for frame $t+1$ in frame t. Therefore, the average capacity of the proposed scheduling is dependent on the probability, P. The proposed scheduling increases the average capacity by reducing the number of transmissions of the signaling messages. At $\gamma = 15$ dB, the average capacity of the proposed scheduling is about 12.5% greater than that of the conventional scheduling when $P = 1.0$ and about 10.96% greater than that of the conventional scheduling when $P = 0.6$.

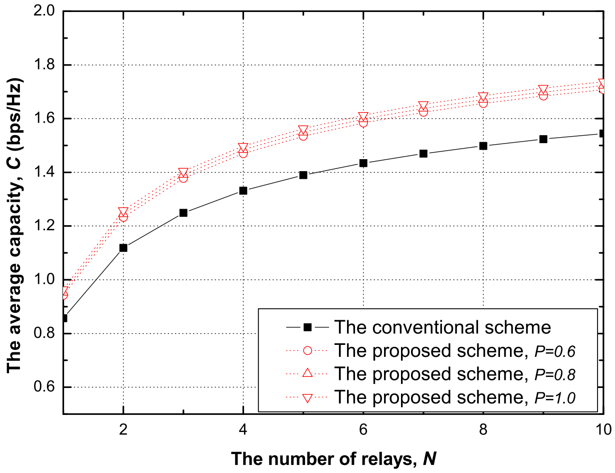


Figure 5. The average capacity versus the number of relays, N

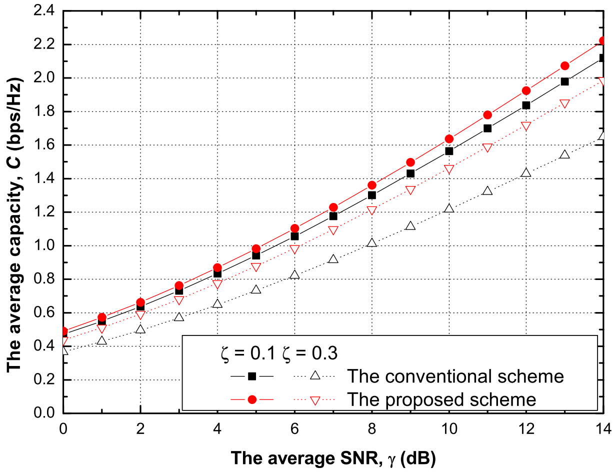


Figure 6. The average capacity versus the average SNR, γ (dB)

Figure 5 shows the average capacity as the number of relays increases when $P = 0.6, 0.8$ and 1.0. As expected, the average capacity increases as the number of relays increases. However, when the number of relays exceeds a certain value (about 10), the average capacity approaches an asymptotic limit. The average capacity of the conventional scheduling has a same performance regardless of the probability, P. At $\bar{\gamma} = 10$ dB, the average capacity of the proposed scheduling

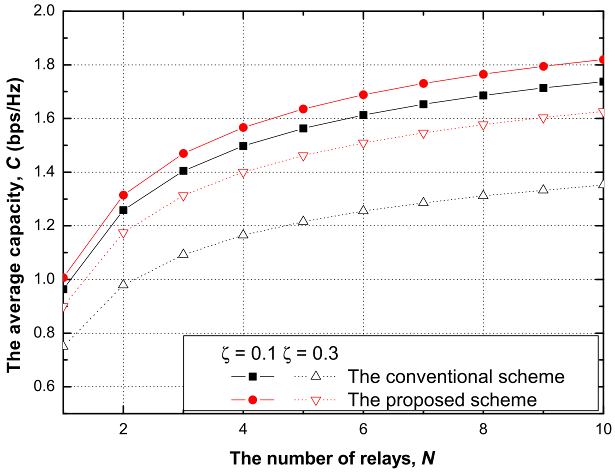


Figure 7. The average capacity versus the number of relays, N

increases by about 12.5% when $P = 1.0$ and by about 10.59% when $P = 0.6$.

Figure 6 shows the average capacity when $\zeta = 0.1$ and 0.3. The average capacity increases as the value of the average SNR increases. The proposed scheduling outperforms the conventional scheduling in an entire SNR region. The available resource for data transmissions in the proposed scheduling is greater than in the conventional scheduling because the number of transmissions of the signaling messages in the proposed scheduling is less than in the convention scheduling. At $\gamma = 15$ dB, the average capacity of the proposed scheduling is about 20.17% greater than that of the conventional scheduling when the ratio of signaling overhead, $\zeta = 0.3$.

Figure 7 shows the average capacity as the number of relays increases when the ratio of the signaling overhead is $\zeta = 0.1$ and 0.3. When the number of relays exceeds a certain value (about 10), the average capacity approaches an asymptotic limit. The asymptotic limit of the average capacity is proportional to the ratio of the signaling overhead. The proposed scheduling outperforms the conventional scheduling as the number of relays increases. The average capacity of the proposed scheduling increases by about 20.3% when $\zeta = 0.3$.

V. Conclusion and Future Work

The paper proposed an efficient scheduling in consideration of the signaling overhead for the downlink relay networks. Additionally, we developed a modified frame structure for scheduling. In the proposed scheduling, the source broadcasts a signaling message only for frame t and does not broadcast the signaling message for frame $t + 1$. Hence, the signaling overhead decreases and the system capacity increases. The simulation results show that the proposed scheduling outperforms the conventional scheduling in terms of the average capacity. When the ratio of signaling overhead is 30%, the average capacity of the proposed scheduling is roughly 20% higher than that of the conventional scheduling.

To develop the proposed scheduling, we have several directions for future work that can be envisioned. One is to consider a specific channel state transition model. Second is to consider a concrete model for the signaling overhead. Another is to determine an optimal the number of frames without the notification of the signaling messages.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (2012-0002089 and 2012012059). Correspondence should be addressed to Jaewoo So, jwso@sogang.ac.kr.

References

[1] H. Wei and R. D. Giltin "Two-hop-relay architecture for next-generation WWAN/WLAN integration," *IEEE Wireless Communications,* vol. 11, no. 2, pp. 24-30, Apr. 2004.

[2] C. Y. Lee and G. U. Hwang "Fair and minimal power allocation in a two-hop relay networks for QoS support," *IEEE Trans. Wireless Commun.,* vol. 10, no. 11, pp. 3864-3873, Nov. 2011.

[3] C. Yan, C. Peng, Q. Peiliang, and Z. Zhaoyang, "Optimal partner selection strategies in wireless cooperative network with fixed and variable transmit power," in Proc. *IEEE WCNC,* Mar. 2007, pp. 4080-4084

[4] T. C. Ng and W. Yu, "Joint optimization of relay strategies and resource allocations in cooperative cellular network," *IEEE J. Sel. Areas Commun.,* vol. 25, no. 2, pp. 328-339, Feb. 2007.

[5] Z. Ma, K. Zheng, W. Wang, and Y. Liu, "Route selection strategies in cellular networks with two-hop relaying," in Proc. *WiCom,* Sep. 2009, pp. 1-4.

[6] H.I. Cho and G.U. Hwang "Capacity optimal design of two-hop relay network," *IEEE Electronic Lett.,* vol. 46, no. 18, pp. 1295-1297, Sep. 2010.

[7] H. Viswanathan and S. Mukherjee, "Performance of cellular networks with relays and centralized scheduling," *IEEE Trans. Wireless Commun.,* vol. 4, no. 5, pp. 2318-2328, Sep. 2005.

[8] A. Bletsas, A. Khisti, D. P. Reed, and A. Lippman, "A simple cooperative diversity mothod based on network path selection," *IEEE J. Sel. Areas Commun.,* vol. 24, no. 3, pp. 659-672, Mar. 2006.

[9] I. Krikidis, J. Thompson, S. McLaughlin, and N. Goertz, "Amplify-and-Forward with Partial Relay Selection," *IEEE Commun. Lett.,* vol. 12, no. 4, pp. 235-237, Apr. 2008.

[10] A. Papadogiannis, A. Saadani, and E. Hardouin, "Exploiting dynamic relays with limited overhead in cellular systems," in Proc. *Globcom Workshops,* Dec. 2009, pp. 1-6.

[11] A. Adinoyi, Y. Fan, H. Yanikomeroglu, H. V. Poor, and F. Al-Shaalan, "Perforamcne of selection relaying and cooperative diversity," *IEEE Trans. Wireless Commun.,* vol. 8, no. 12, pp. 5790-5795, Dec. 2009.

[12] N. Balakrishnan and A. C. *Cohen, Order Statistics and Inference: Estimation Methods.* New York: Academic Press, 1991.

[13] L. Dai, B. Gui, and L. J. Cimini, "Selective relaying in OFDM multihop cooperative networks," in Proc. *IEEE WCNC,* Mar. 2007, pp. 964-969.

[14] C. E. Shannon, "A mathematical theory of communication," *Bell System Tech. J.,* pp. 379-423, 623-56, 1948.