



**Biblioteka Informatyki  
Szkół Wyższych**

# **Information Systems Architecture and Technology**

**Selected Aspects of Communication  
and Computational Systems**



## **Library of Informatics of University Level Schools**

Series of editions under the auspices  
**of the Ministry of Science and Higher Education**

The ISAT series is devoted to the publication of original research books in the areas of contemporary computer and management sciences. Its aim is to show research progress and efficiently disseminate current results in these fields in a commonly edited printed form. The topical scope of ISAT spans the wide spectrum of informatics and management systems problems from fundamental theoretical topics to the fresh and new coming issues and applications introducing future research and development challenges.

The Library is a sequel to the series of books including Multidisciplinary Digital Systems, Techniques and Methods of Distributed Data Processing, as well as Problems of Designing, Implementation and Exploitation of Data Bases from 1986 to 1990.

Wrocław University of Technology



# **Information Systems Architecture and Technology**

*Selected Aspects of Communication  
and Computational Systems*

Editors

*Adam Grzech*

*Leszek Borzemski*

*Jerzy Świątek*

*Zofia Wilimowska*

Wrocław 2014

**Publication partly supported by**  
Faculty of Computer Science and Management  
Wrocław University of Technology

*Project editor*  
Arkadiusz GÓRSKI

The book has been printed in the camera ready form

All rights reserved. No part of this book may be reproduced,  
stored in a retrieval system, or transmitted in any form or by any means,  
without the prior permission in writing of the Publisher.

© Copyright by Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2014

OFICyna WYDAWNICZA POLITECHNIKI WROCLAWSKIEJ  
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław  
<http://www.oficwyd.pwr.wroc.pl>;  
e-mail: [oficwyd@pwr.wroc.pl](mailto:oficwyd@pwr.wroc.pl)  
[zamawianie.książek@pwr.wroc.pl](mailto:zamawianie.książek@pwr.wroc.pl)

ISBN 978-83-7493-856-3

# CONTENTS

Introduction .....	5
--------------------	---

## PART 1. NETWORKS PLANNING, ANALYSIS AND EVALUATION

1. Sylwester KACZMAREK, Maciej SAC Call Processing Performance in a Multidomain IMS/NGN with Asymmetric Traffic .....	11
2. Marcin CZAJKOWSKI, Artur GORCZYCA, Sylwester KACZMAREK, Krzysztof SZALAJDA, Paweł KACZMAREK The Optical Transport Network Control Based on SDN Architecture .....	29
3. Mariusz GŁĄBOWSKI, Michał Dominik STASIAK Point-to-Point Blocking in Switching Networks with Overflow Links Servicing Selected Traffic Classes .....	41
4. Remigiusz RAJEWSKI Quality of Optical Connections in the $MBA(N, e, 2)$ Switching Networks .....	55
5. Marcin DZIUBA The New Parameters of Blocking Window Algorithm .....	65
6. Wojciech M. KEMPA On Transient Virtual Delay in a Finite-Buffer Queueing Model with Server Breakdowns .....	77
7. Jan KWIATKOWSKI, Andrzej GNATOWSKI Parallelization of Wireless Routers Placement Algorithms for the GPU .....	87
8. Krzysztof JUSZCZYSZYN A Mixed Approach to the Link Prediction Problem in Dynamic Complex Network .....	97

## PART 2. INTERNET OF THINGS USE CASES

9. Michał DYK, Andrzej NAJGEBAUER, Dariusz PIERZCHAŁA Augmented Perception Using Internet of Things .....	109
10. Mariusz CHMIELEWSKI, Wojciech KULAS, Marcin KUKIEŁKA, Damian FRĄSZCZAK, Dawid BUGAJEWSKI, Jakub KĘDZIOR, Damian RAINKO, Piotr STAPOR Development of Operational Picture in DSS Using Distributed SOA Based Environment, Tactical Networks and Handhelds .....	119
11. Haoxi ZHANG, Cesar SANIN, Edward SZCZERBICKI, Zhongyuan REN, Zhou FANG, Tailiang CHEN The Decisional DNA-Based Smart Bike for Internet of Things .....	133

### **PART 3. COMPRESSION, RECOGNITION AND SCHEDULING ALGORITHMS**

12. Aleksandr KATKOW	
Additive Algorithms for Accelerated Compression of Information .....	145
13. Jarosław KOSZELA, Karol WÓJCIK, Roman WANTOCH-REKOWSKI	
Gesture Recognition Interface Architecture for Virtual Simulation .....	157
14. Zbigniew BUCHALSKI	
An Heuristic Solution Procedure for Solving the Programs Scheduling Problem in Multiprocessing Computer System .....	167

### **PART 4. SECURITY AND PRIVACY**

15. Marek JASIŃSKI, Ryszard ANTKIEWICZ	
The Hidden PERT Model .....	179
16. Agnieszka DURAJ	
Preservation of Privacy in Association Rules .....	191

## INTRODUCTION

The overall gain of contemporary proposed, designed, deployed and used ICT (Information and Communication Technologies) applications is to explore and utilize new concepts, paradigms, attempts, methods, algorithms and architectures. There are two general reasons for the analysis of ICT technologies and ICT applications. The first is to discover new possible applications which are complementary to the already known. The second is to increase the effectiveness of business processes and to propose applications of high societal value through making use of reappraised distributed systems architectures, services and technologies in large-scale application context.

The book contains several chapters where Authors present original results of investigations devoted to study state-of-the-art as well as applicability of both known technologies properties and their possible applications using various methodologies, approaches, models and algorithms for distributed systems and its components evaluation. The mentioned chapter address various aspects of contemporary distributed information systems specified both by scope of used technology and possible area of ICT application.

Chapters, selected and presented in the book are devoted to discuss - on a very different level of generality – some selected communication technologies and address a number of issues important and representative both for available information and communication technologies as well as information system users requirements and applications. Submissions, delivered within distinguished chapters, are strongly connected with issues being important for contemporary information processing, communication and data communication system. Some of the chapters are to present of results obtained during the work on the practical implementation of information systems.

The book is divided into four complementary parts, which include sixteen chapters. The parts have been completed arbitrary from set of chapters where some extensively researched and recounted in the world literature, important and actual, issues of distributed information systems are discussed. The proposed decomposition of accepted set of chapters into parts is to compose units characterize by application area or how to use the methods and tools.

The first part – **NETWORKS PLANNING, ANALYSIS AND EVALUATION** – consists of chapters addressing various issues related to switching systems architectures

and properties, communication systems technologies, parallel processing architectures, analysis of switching, communication and services delivery systems as well as various methods and algorithms for performance evaluation of communication and processing distributed systems.

The second part – **INTERNET OF THINGS USE CASES** – contains chapters where some selected and promising aspects of utilizing Internet of Things concept as combined with various concepts of knowledge and data processing.

The next, third part – **COMPRESSION, RECOGNITION AND SCHEDULING ALGORITHMS** – is composed of chapters where algorithmic and computational aspects are the most important.

The modest number fourth part – **SECURITY AND PRIVACY** – contains chapters where both important and interesting practical security and privacy aspects are presented.

## **PART 1. NETWORKS PLANNING, ANALYSIS AND EVALUATION**

In the **Chapter 1** an analytical traffic model of a multi-domain Next Generation Network (NGN) is used to evaluate Call Set-up Delay (CSD) and Call Disengagement Delay (CDD), which are standardized call processing performance parameters. The outputs of the model are calculated for various types of intra- and inter-operator calls, taking into account network and load parameters. In this contribution asymmetric cases are investigated, in which selected parameters of two connected network's domains are different. Obtained quantitative results indicate the relations between parameters of one domain as well as mean CSD and mean CDD for calls originated in both domains.

The **Chapter 2** is devoted to discuss usability of the Software-Defined Networking (SDN) concept to control optical transport networks. For this purpose, results of testing of the developed easy-to-use connection scheduler, capable of controlling connections in optical transport networks, are presented.

In the **Chapter 3**, a new method, being an extension of the PPBMT (Point-to-Point Blocking for Multichannel Traffic) method, of the point-to-point blocking probability calculation in multiservice switching networks with overflow links is presented. To illustrate the method's efficiency, the results of the analytical availability calculations for 3-stage Clos switching networks are compared with the data of the discrete events simulations of the switching networks with chosen structures.

The **Chapter 4** purposes are to present and discuss details of algorithms for calculating the first-, the second-, and the third-order crosstalk stage-by-stage in a new  $MBA(N, e, 2)$  switching fabric. In the chapter the considered network are compared with a typical *baseline* and Beneš switching networks of the same capacity and functionality.

In this **Chapter 5** performance evaluation of Blocking Window Algorithm (BWA) with variable window size in blocking multicast  $\log_2(N, 0, p)$  switching networks is presented and studied. The chapter's gain is to show how different parameters of the

BWA influences quality of switching and how to select the best sets of these parameters values.

The **Chapter 6** gains is to present a finite-buffer queueing model of system with unreliable server. The discussed case is based on assumptions that customers occur according to a Poisson process and are being served during exponentially distributed processing time followed by generally distributed repair periods. The embedded Markov chain is used to model the considered system and to calculate queueing delay conditional distributions.

The aim of the **Chapter 7** is a quantitative analysis of using Graphics Processing Unit (GPU) to speed up the simulation and optimizations tasks as used for simulation of electromagnetic wave propagation and optimization of access points localizations in Wireless Local Area Network (WLAN). Results of experiments based on COST231 algorithm are presented and discussed.

In the **Chapter 8** a mixed approach to the link prediction problem in complex, dynamic social networks is proposed. It was shown that application of a weighted sum of three predictors based on totally different assumptions (time series analysis, global and local network topology) outperforms results returned by particular predictors used separately. The proposed approach is illustrated by examples.

## **PART 2. INTERNET OF THINGS USE CASES**

The **Chapter 9** is to propose a concept of *augmented perception* in which connected smart devices can support our senses. According to this approach, human is the central element of the Internet of Things (IoT), and can gather new information by describing his needs which should be satisfied by smart devices to satisfy. The concept is illustrated using described SenseSim simulator which is to build augmented perception virtually.

The **Chapter 10** is to present the development of efficient data fusion and integration methods implemented in SOA environment and delivered for specialized tactical handhelds. Developed and presented system serves as a server solution offering both an integration platform and a command and control portal.

In the **Chapter 11** a novel application of the Internet of Things (IoT), the Decisional DNA-based Smart Bike is presented. The Decisional DNA, being is a domain-independent, flexible and standard knowledge representation structure, combining with the sensor-equipped bicycle is able to learn its user's weight, riding habits, etc. It was shown that the Decisional DNA can be used on IoT applications enabling knowledge capturing, processing and reusing.

## **PART 3. COMPRESSION, RECOGNITION AND SCHEDULING ALGORITHMS**

The **Chapter 12** is to present a new additive algorithm for the calculation of Discrete Cosine Transform (DCT), which is widely used in the practice of converting graphics

and numerical solution of differential equations. The concept of accelerating calculation is illustrated by an example.

The aim of the **Chapter 13** is to present an approach how to combine gestures detections and the virtual simulation environment. To illustrate the proposed approach, an interface architecture and its implementation in a virtual Virtual Battlespace 2 (VBS2) simulation environment is presented.

The **Chapter 14** results of an heuristic algorithm for solving time-optimal programs scheduling and primary memory pages allocation in multiprocessing computer system task is presented. Some selected results are presented to show features of the proposed approach.

#### **PART 4. SECURITY AND PRIVACY**

The main problem considered in the **Chapter 15** is the detection of projects realized secretly. It is assumed that observations, generated by actions within not directly observable hidden project, could be collected and processed to identify current state of the project. The proposed approach is to combine Hidden Markov Model (HMM) and PERT network to model hidden project with actions performed in parallel.

The **Chapter 16** addresses the issues of assessing the impact of privacy protection methods on the detection of association rules. Presented results of tests show that the privacy protection algorithms have an impact on the detection of association rules.

Wroclaw, September 2014

*Adam Grzech*

**PART 1**

**NETWORKS PLANNING, ANALYSIS  
AND EVALUATION**



Sylwester KACZMAREK\*, Maciej SAC\*

## **CALL PROCESSING PERFORMANCE IN A MULTIDOMAIN IMS/NGN WITH ASYMMETRIC TRAFFIC**

In this paper we continue our research using the previously proposed analytical traffic model of a multidomain Next Generation Network (NGN), which is standardized for delivering multimedia services based on the IP Multimedia Subsystem (IMS). The aim of the model is to assess mean Call Set-up Delay (CSD) and mean Call Disengagement Delay (CDD), which are standardized call processing performance parameters. The output variables of the model are calculated for various types of intra- and inter-operator calls, taking into account a large set of network parameters large set and terminal registrations, which additionally load the network. To complete the previous research, in which the same parameters of the elements and traffic levels in both IMS/NGN domains (symmetric case) were assumed, in this paper asymmetric cases are investigated, in which selected parameters in one domain are different than in the other one. This allows to indicate the relations between parameters of one domain as well as mean CSD and mean CDD for calls originated in both domains.

### 1. INTRODUCTION

Our research concerns the Next Generation Network (NGN) [1], a standardized telecommunication network architecture proposed to fulfill current and future needs regarding distribution of various multimedia services with guaranteed quality. Delivering services in NGN is based on the IP Multimedia Subsystem (IMS) concept [2], and thus the names “IMS-based NGN” and “IMS/NGN” are very commonly used.

The paper continues our previous work [3–10] on a multidomain traffic model of IMS/NGN, which aim is to evaluate mean Call Set-up Delay (mean *CSD*,  $E(CSD)$ ) mean Call Disengagement Delay (mean *CDD*,  $E(CDD)$ ), a subset of call processing performance parameters [11, 12]. These parameters are standardized and closely relat-

---

\* Department of Teleinformation Networks, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, 11/12 Gabriela Narutowicza Street, 80-233 Gdańsk, Poland.

ed to Quality of Service (QoS). Guaranteeing their appropriate values is crucial to make the provided services satisfactory for users.

First stage of our previous work concerned an analytical [3] and simulation [3, 4] traffic model of a single domain of IMS/NGN. To achieve possibly the best conformity of the analytical model with experimental results provided by the simulation model (precisely implementing the operation of all network elements and call scenarios), we investigated different queuing models in the analytical model ([3, 5, 7, 8]). The goal of the first stage of the research was reached and we have recently started working on an analytical model of a multidomain IMS/NGN. In paper [10] we proposed a model of a multidomain IMS/NGN administered by two operators, which takes into consideration different variants of registration, intra-operator and inter-operator calls as well as a definable probability of transport resource unavailability. Using this model basic relations between network parameters and call processing performance parameters were examined assuming that parameters of the elements and traffic levels in both IMS/NGN domains are the same (symmetric case).

In this paper we accomplish that research by investigating the asymmetric cases, which are much more common in practice. During our investigations selected parameters of one domain are modified, while in the other one they remain constant. The aim of the paper is to determine the influence of these modifications on call processing performance parameters (mean *CSD* and mean *CDD*) for all types of calls originated in both domains. The rest of the text is organized as follows. Basic information about the used traffic model of a multidomain IMS/NGN is provided in section 2. The obtained call processing performance results are presented and discussed in section 3. Summary and future work regarding our model are described in section 4.

## 2. TRAFFIC MODEL OF IMS/NGN

The research presented in this paper is performed using the traffic model of a multidomain ITU-T NGN [13,14] (the most advanced of all NGN solutions [15,16]) proposed in [10]. In this section only the most important information about the model is provided. For details regarding the model and calculation of its output variables please refer to [10].

The elements of the model are presented in Fig. 1, which illustrates two IMS/NGN domains administered by two operators (in the paper the terms “operator” and “domain” are used interchangeably with similar meaning). Both operator 1 and 2 have their own core network (controlled by RACF C1 or RACF C2 element) and access network (controlled by RACF A1 or RACF A2 element) with negligible message loss probability. Access network of each operator consists of several access areas. Transport connection between different access areas require resources of the core

network. The network elements illustrated in Fig. 1 perform the following roles [10]:

- User Equipments (UEs): user terminals generating call set-up and disengagement requests, registering themselves in their domains; it is assumed that UEs perform standard voice calls (application servers are not used) and have compatible codec sets; no audio announcements are played during the calls,
- P-CSCF (Proxy – Call Session Control Function): the server responsible for receiving all messages from UEs and forwarding them to the S-CSCF server,
- S-CSCF (Serving – Call Session Control Function): the main server handling all calls in the domain,
- I-CSCF (Interrogating – Call Session Control Function): the server handling messages from other domains,
- SUP-FE/SAA-FE (Service User Profile Functional Entity/Service Authentication and Authorization Functional Entity): the database storing user profiles and location information, performs also authentication and authorization functions,
- RACF (Resource and Admission Control Functions): the unit of the transport stratum allocating resources for a new call and releasing resources during call disengagement; it is assumed that RACF elements control resources using push mode [17]; resources for a call may be unavailable with a defined probability.

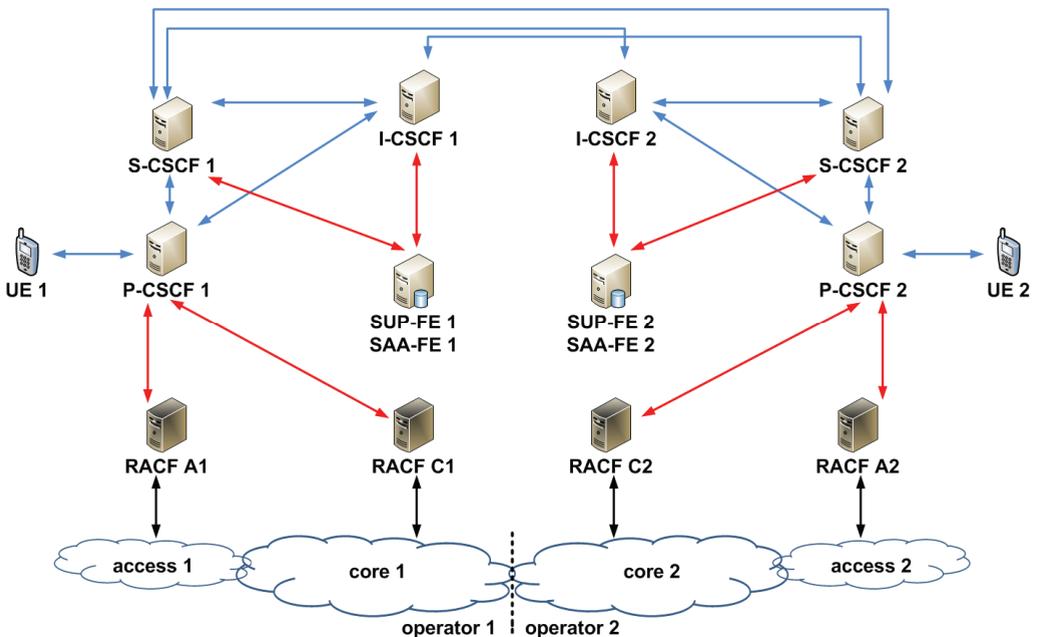


Fig. 1. Model of a multidomain IMS/NGN [10]; blue arrows – communication using SIP protocol [18]; red arrows – communication using Diameter protocol [19]; black arrows – resource control using dedicated protocols

To distinguish between the elements of different operators, numbers 1 and 2 are added to the above mentioned names. The elements presented in Fig. 1 participate in the following set of service scenarios [10, 13, 14, 17, 20–22]:

- a1 – registration of UE (domain 1),
- a2 – registration of UE (domain 2),
- b1 – intra-operator call (domain 1, the same access area, success),
- b2 – intra-operator call (domain 2, the same access area, success),
- c1 – intra-operator call (domain 1, the same access area, resources unavailable),
- c2 – intra-operator call (domain 2, the same access area, resources unavailable),
- d1 – intra-operator call (domain 1, different access areas, success),
- d2 – intra-operator call (domain 2, different access areas, success),
- e1 – intra-operator call (domain 1, different access areas, resources unavailable),
- e2 – intra-operator call (domain 2, different access areas, resources unavailable),
- f1 – inter-operator call (originated in domain 1, success),
- f2 – inter-operator call (originated in domain 2, success),
- g1 – inter-operator call (originated in domain 1, resources unavailable only in originating domain or in both domains),
- g2 – inter-operator call (originated in domain 2, resources unavailable only in originating domain or in both domains),
- h1 – inter-operator call (originated in domain 1, resources unavailable only in terminating domain),
- h2 – inter-operator call (originated in domain 2, resources unavailable only in terminating domain).

Due to lack of space service scenarios are not described in the paper. Full description of the most complicated successful inter-operator call scenario (f1, f2) can be found in [10]. The description of the other scenarios can be partially found in or derived from [13–15, 17, 20–22].

The structure of the traffic model (Fig. 3 in [10]) contains all elements of the network model presented in Fig. 1. CSCF servers contain message queues and Central Processing Units (CPUs), which process messages according to the service scenarios. Other elements (except links, which are described later) do not contain queues. SUP-FE/SAA-FE and RACF message processing times are modeled as random variables with any probability density. For UE 1 and UE 2 blocks (representing many user terminals of domain 1 and 2) message processing times are not included in calculations, according to ITU-T Y.1530 [11] and Y.1531 [12] standards.

All IMS/NGN elements are connected using links. Sending a message through a link involves: buffering the message in a queue before transmission if the link is busy, message transmission time (determined by message length and link bandwidth) and propagation time ( $5\mu\text{s}/\text{km}$  for optical links, which are assumed in the network).

The output variables of the traffic model are mean Call Set-up Delay and mean Call Disengagement Delay. Calculations of  $E(CSD)$  and  $E(CDD)$  are performed based

on the definition of *CSD* and *CDD* given by ITU-T [11,12]:

$$CSD = (t_2 - t_1) + (t_4 - t_3) + (t_6 - t_5) \quad (1)$$

$$CDD = (t_8 - t_7) + (t_{10} - t_9) \quad (2)$$

where:

- $t_1$  is the time of sending call set-up request (SIP INVITE message) by the originating UE,
- $t_2$  is the time of receiving SIP INVITE message by the destination UE,
- $t_3$  is the time of sending confirmation (SIP 200 OK (INVITE) message) by the destination UE,
- $t_4$  is the time of receiving SIP 200 OK (INVITE) message by the originating UE,
- $t_5$  is the time of sending confirmation (SIP ACK message) by the originating UE,
- $t_6$  is the time of receiving SIP ACK message by the destination UE,
- $t_7$  is the time of sending call disengagement request (SIP BYE message) by the originating UE,
- $t_8$  is the time of receiving SIP BYE message by the destination UE,
- $t_9$  is the time of sending confirmation (SIP 200 OK (BYE) message) by the destination UE,
- $t_{10}$  is the time of receiving SIP 200 OK (BYE) message by the originating UE.

$E(CSD)$  and  $E(CDD)$  are computed for all successful call scenarios (b1, b2, d1, d2, f1, f2). Indexes are added to indicate the scenario name. For example,  $E(CSD)_{f1}$  and  $E(CDD)_{f1}$  regard the f1 scenario. To calculate all output variables of the model, the following input variables are used [10]:

- $\lambda_{R1}, \lambda_{R2}$ : registration request (SIP REGISTER) intensity in domain 1 and 2 respectively,
- $\lambda_{11}, \lambda_{22}$ : intra-operator call set-up request (SIP INVITE) intensity (domain 1 and 2 respectively),
- $\lambda_{12}, \lambda_{21}$ : inter-operator call set-up request (SIP INVITE) intensity (requests originated in domain 1 and 2 respectively),
- $r_{C1}, r_{C2}$ : ratio of calls involving multiple access areas to all intra-operator calls (domain 1 and 2 respectively),
- $p_{A1}, p_{C1}, p_{A2}, p_{C2}$ : probability of transport resource unavailability in access 1, core 1, access 2 and core 2 (Fig. 1) respectively,
- $T_{INV\_P1}, T_{INV\_S1}, T_{INV\_I1}, T_{INV\_P2}, T_{INV\_S2}, T_{INV\_I2}$  times of processing SIP INVITE message by P-CSCF 1, S-CSCF 1, I-CSCF 1, P-CSCF 2, S-CSCF 2, I-CSCF 2 respectively,
- $a_k$  ( $k = 1, 2, \dots, 24$ ): the factors determining times of processing other SIP and Diameter messages by CSCF servers CPUs:

$$\begin{aligned}
T_{TR_*} &= a_1 \cdot T_{INV_*}, T_{RING_*} = a_2 \cdot T_{INV_*}, T_{OKINV_*} = a_3 \cdot T_{INV_*}, \\
T_{ACK_*} &= a_4 \cdot T_{INV_*}, T_{BYE_*} = a_5 \cdot T_{INV_*}, T_{OKBYE_*} = a_6 \cdot T_{INV_*}, \\
T_{AAA_*} &= a_7 \cdot T_{INV_*}, T_{STA_*} = a_8 \cdot T_{INV_*}, T_{PRA_*} = a_9 \cdot T_{INV_*}, \\
T_{UPD_*} &= a_{10} \cdot T_{INV_*}, T_{SP_*} = a_{11} \cdot T_{INV_*}, T_{OKUPD_*} = a_{12} \cdot T_{INV_*}, \\
T_{OKPRA_*} &= a_{13} \cdot T_{INV_*}, T_{LIA_*} = a_{14} \cdot T_{INV_*}, T_{SERUN_*} = a_{15} \cdot T_{INV_*}, \\
T_{CAN_*} &= a_{16} \cdot T_{INV_*}, T_{OKCAN_*} = a_{17} \cdot T_{INV_*}, T_{PRFAIL_*} = a_{18} \cdot T_{INV_*}, \\
T_{REG_*} &= a_{19} \cdot T_{INV_*}, T_{UAA_*} = a_{20} \cdot T_{INV_*}, T_{MAA_*} = a_{21} \cdot T_{INV_*}, \\
T_{UNAUTH_*} &= a_{22} \cdot T_{INV_*}, T_{SAA_*} = a_{23} \cdot T_{INV_*}, T_{OKREG_*} = a_{24} \cdot T_{INV_*},
\end{aligned} \tag{3}$$

where “\*” can be replaced by “P1”, “S1”, “I1”, “P2”, “S2”, “I2”, which represent P-CSCF 1, S-CSCF 1, I-CSCF 1, P-CSCF 2, S-CSCF 2, I-CSCF 2 respectively; the  $a_k$  factors concern the following messages: 100 Trying, 180 Ringing, 200 OK (INVITE), ACK, BYE, 200 OK (BYE), AAA, STA, PRACK, UPDATE, 183 Session Progress, 200 OK (UPDATE), 200 OK (PRACK), LIA, 503 Service Unavailable, CANCEL, 200 OK (CANCEL), 580 Precondition Failure, REGISTER, UAA, MAA, 401 Unauthorized, SAA, 200 OK (REGISTER)),

- $E(X_{A1}), E(X_{C1}), E(X_{A2}), E(X_{C2}), E(Y_1), E(Y_2)$ : mean message processing times by RACF A1, RACF C1, RACF A2, RACF C2, SUP-FE 1/SAA-FE 1, SUP-FE 2/SAA-FE 2 respectively,
- lengths and bandwidths of optical links, lengths of transmitted messages: values necessary to calculate communication times between network elements.

For calculations it is assumed that  $\lambda_{R1}, \lambda_{R2}, \lambda_{I1}, \lambda_{I2}, \lambda_{P1}, \lambda_{P2}$  input parameters represent aggregated registration and call set-up request intensities from many UEs. These requests are generated with exponential intervals.

During computations of mean *CSD* (1) and mean *CDD* (2) mean values of the delays grouped in parentheses in  $((t_2 - t_1), (t_4 - t_3), \dots)$  are decomposed into the following delays ([3, 9, 10, 15]):

- mean message waiting times in communication queues, which buffer messages at input of each outgoing link when it is currently busy (for calculation of these times queuing models are necessary),
- message transmission times (message lengths divided by links bandwidth),
- propagation times ( $5\mu\text{s}/\text{km}$  for assumed optical links),
- mean message waiting times in CSCF servers CPU queues, which store incoming messages when CSCF servers CPUs are busy (for calculation of these times queuing models are necessary),
- mean message processing times by CSCF servers CPUs as well as RACF and SUP-FE/SAA-FE elements (these times result from the  $T_{INV\_P1}, T_{INV\_S1}, T_{INV\_I1}, T_{INV\_P2}, T_{INV\_S2}, T_{INV\_I2}, a_k, E(X_{A1}), E(X_{C1}), E(X_{A2}), E(X_{C2}), E(Y_1), E(Y_2)$  input variables).

As already discussed in [10], in the first part of our research in a multidomain IMS/NGN we assume that operation of communication queues and CSCF servers CPU queues will be approximately represented using M/G/1 queuing models. Although message inter-arrival times in the network are generally not exponential [6], such approach allows easy computation of the output variables based only on the input variables (due to limited space, more details regarding these calculations are not provided and can be found in [3, 9, 10]). Moreover, the same approach was used in the analytical traffic model of a single domain of IMS/NGN [3]. This model was successfully verified by simulations with acceptable conformity excluding high load of servers or links. During our research we managed to find good queuing models for these areas (PH/PH/1 queuing models [7, 8], in which arrival and service distributions are represented by phase-type distributions [23, 24]). They, however, require experimental message inter-arrival histograms, which have to be obtained using an accurate simulation model [4, 6]. Therefore, PH/PH/1 queuing models will be applied in the analytical model of a multidomain IMS/NGN as soon as a proper simulation model is developed and the analytical model is verified. This work is already in progress.

### 3. RESULTS

In this section we present the obtained call processing performance results. The experiments are based on a default set of input variables, which was used in our previous research [10]:

- $\lambda_{R1} = \lambda_{R2} = 50/s$ ,
- $\lambda_{11} = \lambda_{22} = 50/s$ ,
- $\lambda_{12} = \lambda_{21} = 50/s$ ,
- $r_{C1} = r_{C2} = 0.5$ ,
- $p_{A1} = p_{C1} = p_{A2} = p_{C2} = 0$ ,
- $T_{INV\_P1} = T_{INV\_S1} = T_{INV\_I1} = T_{INV\_P2} = T_{INV\_S2} = T_{INV\_I2} = 0.5$  ms,
- $E(X_{A1}) = E(X_{C1}) = E(X_{A2}) = E(X_{C2}) = 10$  ms,
- $E(Y_1) = E(Y_2) = 10$  ms,
- lengths of all links are the same and equal to  $d = 200$  km,
- bandwidths of all links are the same and equal to  $b = 50$  Mb/s,
- $\{a_1, \dots, a_{24}\} = \{0.2, 0.2, 0.6, 0.3, 0.6, 0.3, 0.6, 0.6, 0.3, 0.5, 0.6, 0.5, 0.3, 0.5, 0.4, 0.6, 0.4, 0.4, 0.6, 0.5, 0.5, 0.3, 0.4, 0.3\}$ ,
- SIP message lengths in bytes [25]: *INVITE*: 930, *PRACK*: 450, *UPDATE*: 930, *ACK*: 630, *BYE*: 510, *CANCEL*: 550, *REGISTER*: 700, *100 Trying*: 450, *180 Ringing*: 450, *183 Session Progress*: 910, *200 OK (INVITE, UPDATE)*: 990, *200 OK (PRACK, BYE, REGISTER, CANCEL)*: 500, *401 Unauthorized*: 700, *503 Service Unavailable*: 500, *580 Precondition Failure*: 550,
- Diameter message length: 750 bytes.

Using the above mentioned set of input variables we achieve the same parameters of the elements and traffic levels in both IMS/NGN domains (symmetric case). This results in the same values of call processing performance parameters for calls originated in domain 1 and 2. Such a situation was the subject of the paper [10].

In this paper we investigate asymmetric cases in which value of one of the input variables in domain 2 is modified, while the corresponding value in domain 1 is default. The results of the investigations are presented in Figs. 2–9, which depict mean *CSD* and mean *CDD* for b1, b2, d1, d2, f1, f2 scenarios (section 2). In the research presented in this paper mean message processing times by RACF elements in particular domains are the same (but generally not the same in the whole network). Therefore the results for b1, d1 as well as b2, d2 scenarios are identical:

- $E(CSD)_{b1} = E(CSD)_{d1} = E(CSD)_{b1/d1}$ ,
- $E(CSD)_{b2} = E(CSD)_{d2} = E(CSD)_{b2/d2}$ ,
- $E(CDD)_{b1} = E(CDD)_{d1} = E(CDD)_{b1/d1}$ ,
- $E(CDD)_{b2} = E(CDD)_{d2} = E(CDD)_{b2/d2}$ .

In each subfigure (Figs. 2–9 contain two subfigures) there are two surfaces representing  $E(CSD)$  and  $E(CDD)$  respectively. Due to the fact that for all scenarios call set-up process is more complicated than call disengagement (more messages are sent and processed), the surfaces for mean *CSD* are always above these for mean *CDD*.

The relation between call processing performance parameters in both domains and normalized values of registration request intensity in domain 2 ( $m_1 = \lambda_{R2}/\lambda_{R1}$ ) as well as intra-operator call set-up request intensity in domain 2 ( $m_2 = \lambda_{22}/\lambda_{11}$ ) can be observed in Figs. 2–3. Both  $m_1$  and  $m_2$  influence the load of the elements of domain 2. Consequently, their higher values results in higher  $E(CSD)$  and  $E(CDD)$  for f1, f2, b2 and d2 scenarios, in which elements of domain 2 are used. As elements of domain 2 do not participate in intra-operator calls in domain 1,  $E(CSD)_{b1/d1}$  and  $E(CDD)_{b1/d1}$  are not affected by  $m_1$  and  $m_2$ . Comparing mean *CSD* and mean *CDD* values for f1 and f2 scenarios, we can notice that  $E(CSD)_{f2}$  and  $E(CDD)_{f2}$  are similarly influenced by  $m_1$  and  $m_2$  to  $E(CSD)_{f1}$  and  $E(CDD)_{f1}$ .

In another experiment we test the influence of normalized  $\lambda_{21}$  ( $m_1 = \lambda_{21}/\lambda_{12}$ ) as well as  $T_{INV\_*2}$  ( $m_2 = T_{INV\_*2}/T_{INV\_*1}$ ; \* can be here replaced with “P”, “S”, and “I”) on mean *CSD* and mean *CDD* (Figs. 4–5). The number of inter-operator calls originated in domain 2 in a unit time period ( $\lambda_{21}$ ) increases the load of both domains. Therefore, with  $m_1$  rise all analysed call processing performance parameters. Times of processing SIP INVITE messages by CSCF servers of domain 2 ( $T_{INV\_*2}$ ) increase only the load of this domain. Thus, higher  $m_2$  values result only in higher mean *CSD* and mean *CSD* for calls originated or terminated in this domain (f1, f2, b2 and d2 scenarios).

The next analyzed input variables are mean message processing times by RACF (in access and core,  $m_1 = E(X_{A2})/E(X_{A1}) = E(X_{C2})/E(X_{C1})$ ) and SUP-FE/SAA-FE

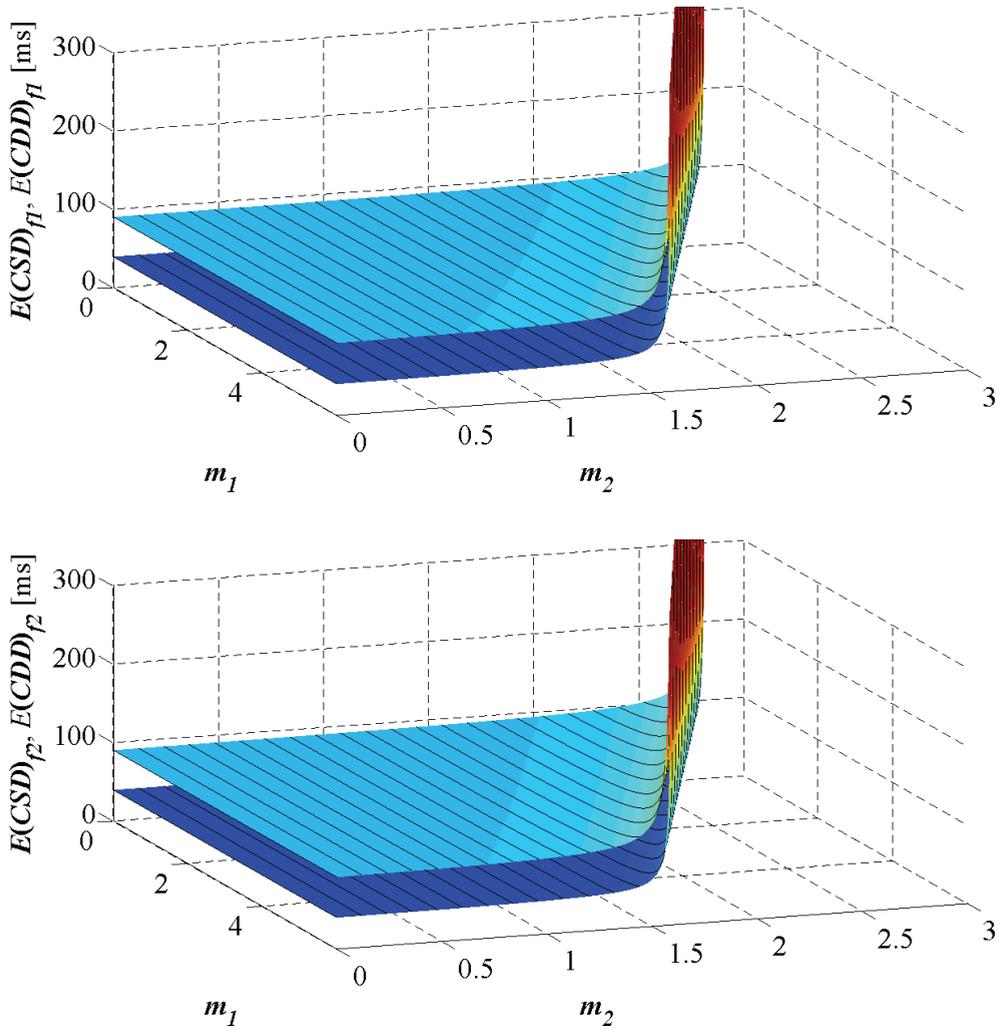


Fig. 2. Call processing performance results for inter-operator calls;  
 $\lambda_{R1} = 50/s$ ,  $m_1 = \lambda_{R2}/\lambda_{R1}$ ,  $\lambda_{11} = 50/s$ ,  $m_2 = \lambda_{22}/\lambda_{11}$ , other input variables are default

( $m_2 = E(Y_2)/E(Y_1)$ ) elements in domain 2 (Figs. 6–7). Higher  $m_1$  values result only in increasing call processing performance parameters for scenarios involving resource reservation in domain 2 (f1, f2, b2 and d2 scenarios). A more interesting situation takes place for the second analyzed parameter ( $m_2$ ). As SUP-FE 2/SAA-FE 2 is used only during set-up of inter-operator call scenario originated in domain 1 (for location of destination user S-CSCF server [10]),  $m_2$  only influences  $E(CSD)_{f1}$ . It has no impact on other analysed call processing performance parameters.

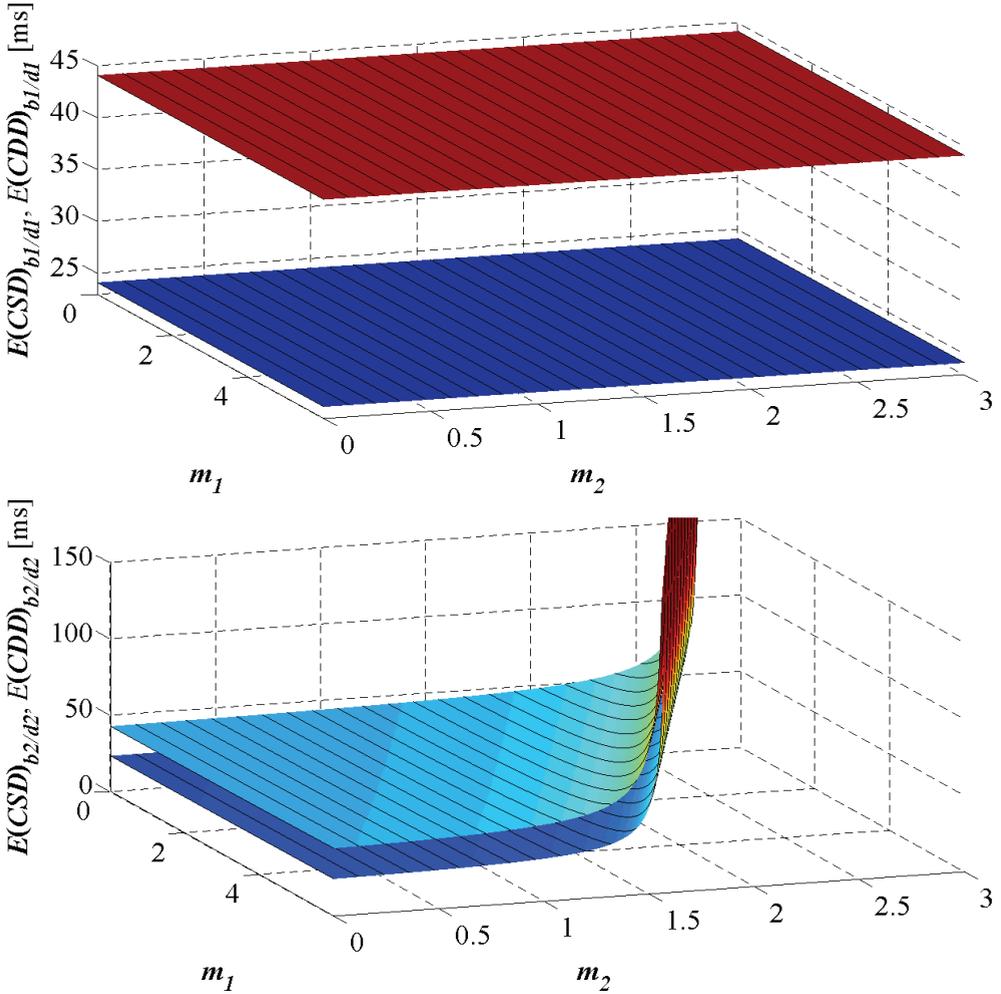


Fig. 3. Call processing performance results for intra-operator calls;  
 $\lambda_{R1} = 50/s$ ,  $m_1 = \lambda_{R2}/\lambda_{R1}$ ,  $\lambda_{I1} = 50/s$ ,  $m_2 = \lambda_{I2}/\lambda_{I1}$ , other input variables are default

Another case (Figs. 8–9) concerns normalized lengths ( $m_1 = d_2/d_1$ ) and bandwidths ( $m_2 = b_2/b_1$ ) of links in domain 2. Both  $m_1$  and  $m_2$  influence mean *CSD* and mean *CDD* for all scenarios in which messages are sent to and received from domain 2 (all scenarios except b1/d1). Lengths of the links in domain 2 ( $m_1$ ) increase call processing performance parameters for f1, f2, b2 and d2 scenarios proportionally, due to distant dependent propagation times. For higher  $m_1$  values propagation times can be a very important part of mean *CSD* and mean *CDD*. It is interesting that for inter-operator calls more messages are exchanged in the destination user domain [10]. Therefore, larger values of  $m_1$  affect  $E(CSD)_{f1}$  and  $E(CDD)_{f1}$  stronger than  $E(CSD)_{f2}$  and  $E(CDD)_{f2}$ . When it comes to link bandwidths in domain 2 ( $m_2$ ), they have to be large

enough to guarantee low message transmission times and, consequently, mean message waiting times in communication queues. For the investigated sets of network parameters, 25 Mb/s ( $m_2 = 0.5$ ) is a minimal acceptable value of link bandwidth in domain 2, which do not significantly affect call processing performance parameters.

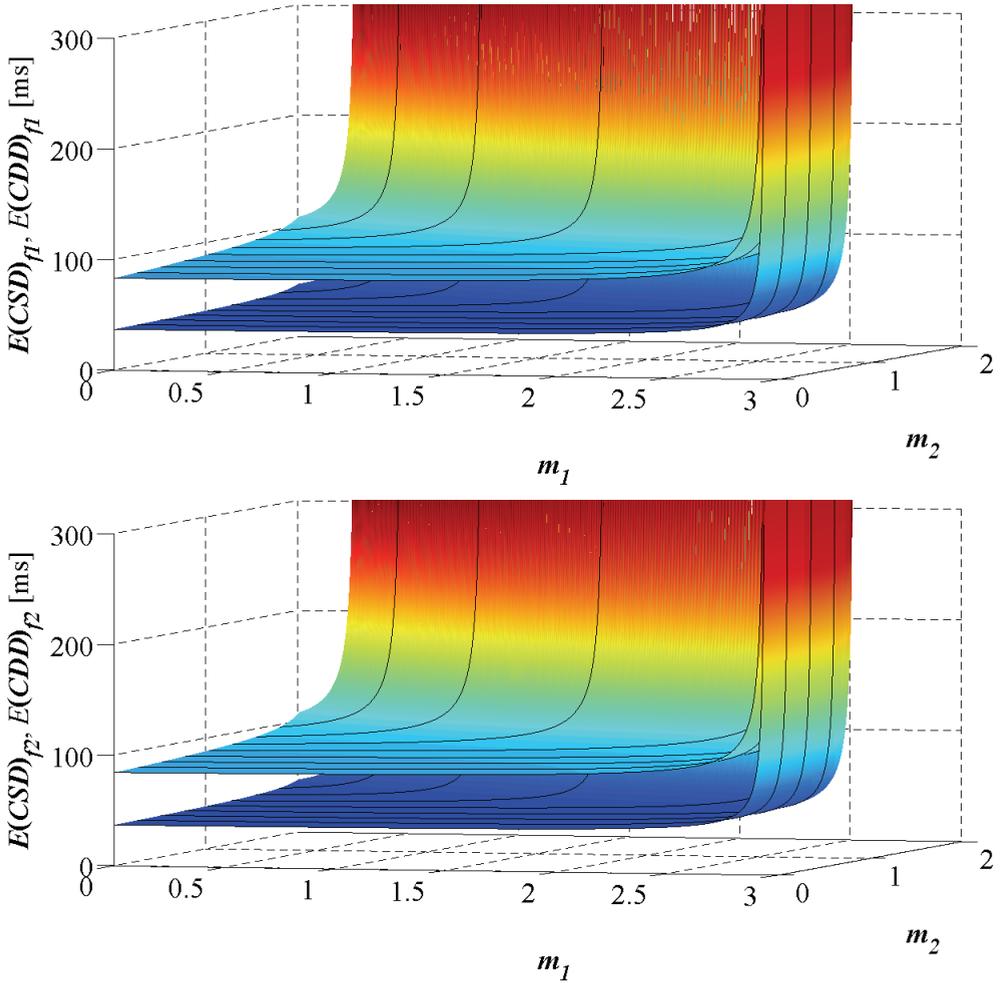


Fig. 4. Call processing performance results for inter-operator calls;  
 $\lambda_{12} = 50/s$ ,  $m_1 = \lambda_{21}/\lambda_{12}$ ,  $T_{INV\_*1} = 0.5$  ms,  $m_2 = T_{INV\_*2}/T_{INV\_*1}$ , other input variables are default

Apart from the cases illustrated in Figs. 2–9, we also investigated how call processing performance parameters are influenced by  $r_{C2}$ ,  $p_{A2}$  and  $p_{C2}$  input variables. This influence is very little and hard to notice visually. Therefore, no illustrations are presented. The ratio of calls involving multiple access areas to all intra-operator calls in domain 2 ( $r_{C2}$ ) very slightly increases the load of elements of domain 2 [10]. Conse-

quently, it affects mean *CSD* and mean *CDD* only for f1, f2, b2 and d2 scenarios (in b1 and d1 scenarios elements of domain 2 are not used). The probability of transport resource unavailability in domain 2 (for access and core, we assume that always  $p_{A2} = p_{C2}$ ) increases the number of unsuccessful call scenarios, which involve sending and processing less messages than for successful scenarios (this applies to both domains). As a result, the load of both domains as well as mean *CSD* and *CDD* for all call scenarios are decreased.

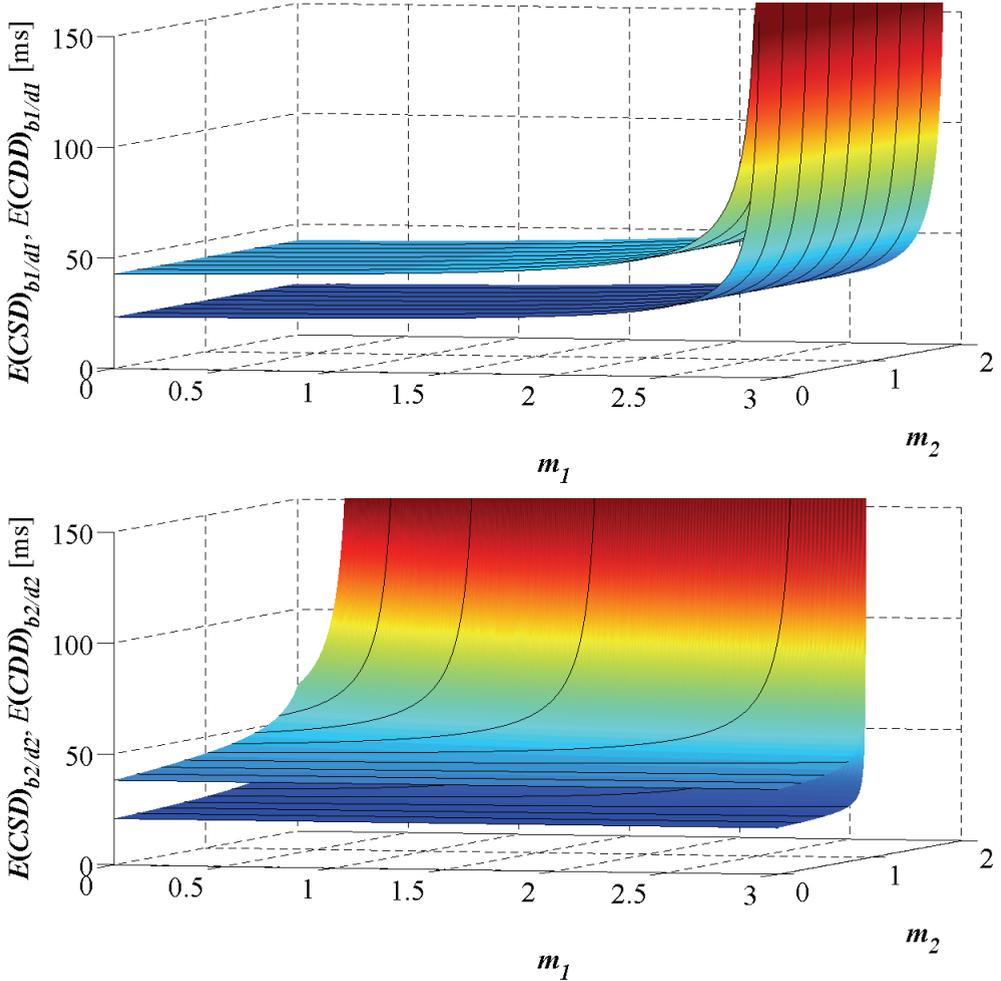


Fig. 5. Call processing performance results for intra-operator calls;  
 $\lambda_{12} = 50/s$ ,  $m_1 = \lambda_{21}/\lambda_{12}$ ,  $T_{INV\_*1} = 0.5$  ms,  $m_2 = T_{INV\_*2}/T_{INV\_*1}$ , other input variables are default

All the above mentioned observations are summarized in Tabs. 1–2, which present relations between parameters of domain 2 and call processing performance parameters

of both domains. In the assumed nomenclature these relations can be “positive” (the output variable increases with larger values of the input variable), “negative” (the output variable decreases with larger values of the input variable) or “none” (the output variable does not depend on the input variable). We can see that in many cases parameters of domain 2 increase mean Call Set-up Delay and mean Call Disengagement Delay for calls which are originated or terminated in this domain (f1, f2, b2 and d2 scenarios) and do not affect  $E(CSD)_{b1/d1}$  and  $E(CDD)_{b1/d1}$ . The influence of the input variables on mean CSD and mean CDD is generally similar for f1, f2, b2 and d2 scenarios. However, values of  $E(CSD)$  and  $E(CDD)$  are always smaller for intra-operator scenarios (b2, d2) than for inter-operator scenarios (f1, f2).

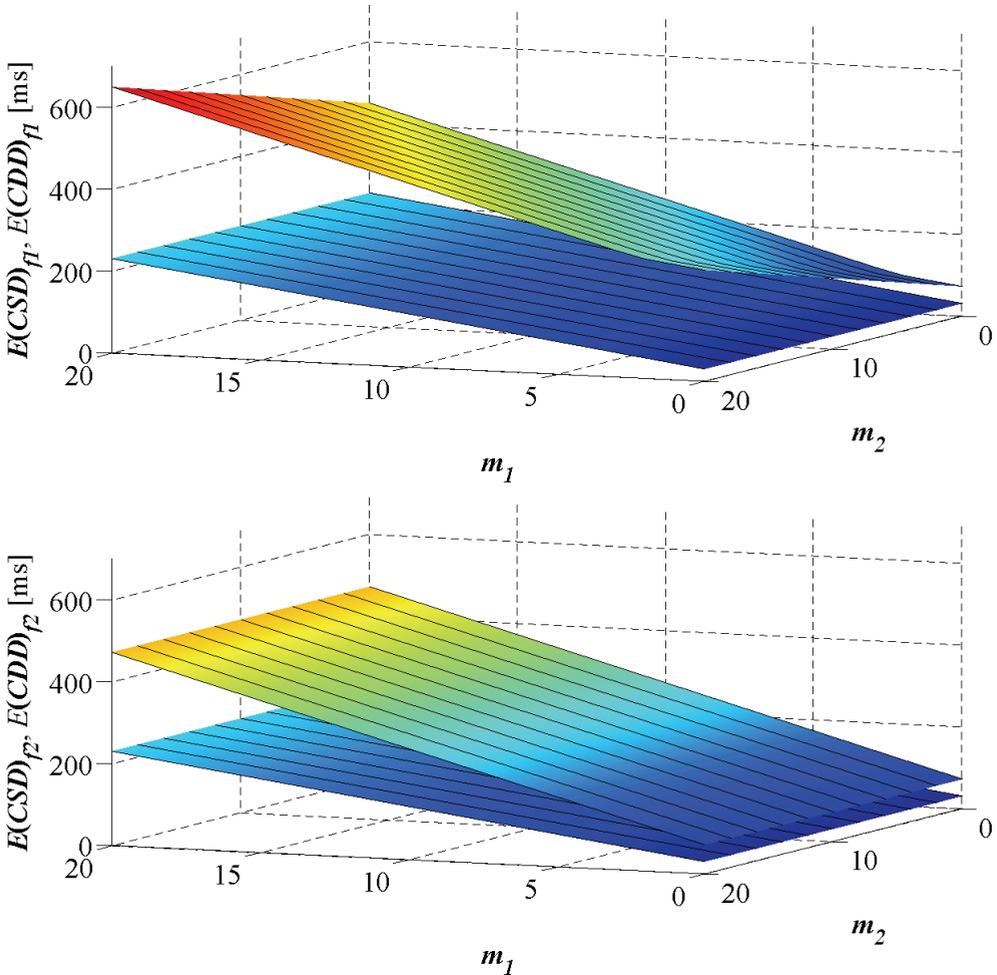


Fig. 6. Call processing performance results for inter-operator calls;  $E(X_{A1}) = E(X_{C1}) = 10$  ms,  $m_1 = E(X_{A2})/E(X_{A1}) = E(X_{C2})/E(X_{C1})$ ,  $E(Y_1) = 10$  ms,  $m_2 = E(Y_2)/E(Y_1)$ , other input variables are default

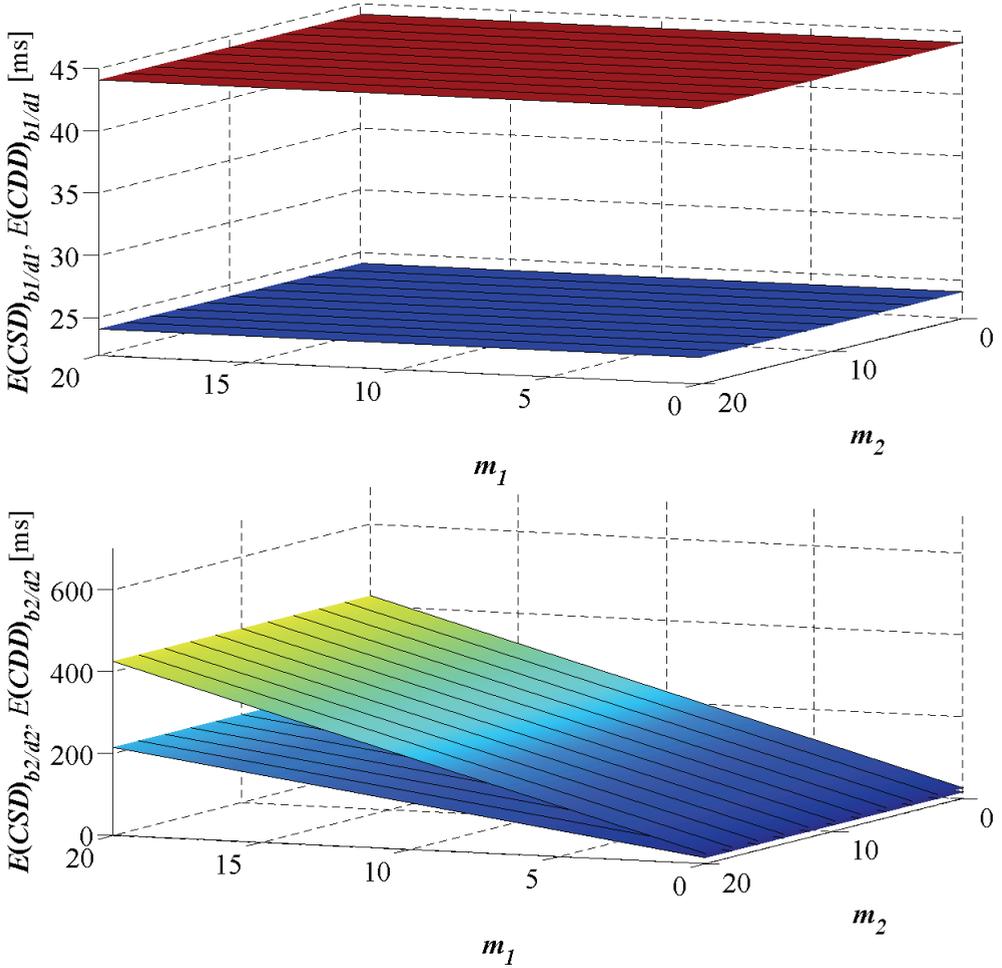


Fig. 7. Call processing performance results for intra-operator calls;  $E(X_{A1}) = E(X_{C1}) = 10$  ms,  $m_1 = E(X_{A2})/E(X_{A1}) = E(X_{C2})/E(X_{C1})$ ,  $E(Y_1) = 10$  ms,  $m_2 = E(Y_2)/E(Y_1)$ , other input variables are default

#### 4. CONCLUSIONS AND FUTURE WORK

The paper is a continuation of our research concerning an analytical traffic model of a multidomain IMS/NGN controlled by two network operators. The model takes into account an extensive set of network parameters and service scenarios (e.g. terminal registration as well as intra-operator and inter-operator calls, which can be successful or unsuccessful due to transport resource unavailability). The output variables of the model are mean Call Set-up Delay and mean Call Disengagement Delay (a subset

of standardized call processing performance parameters) for both intra-operator and inter-operator calls.

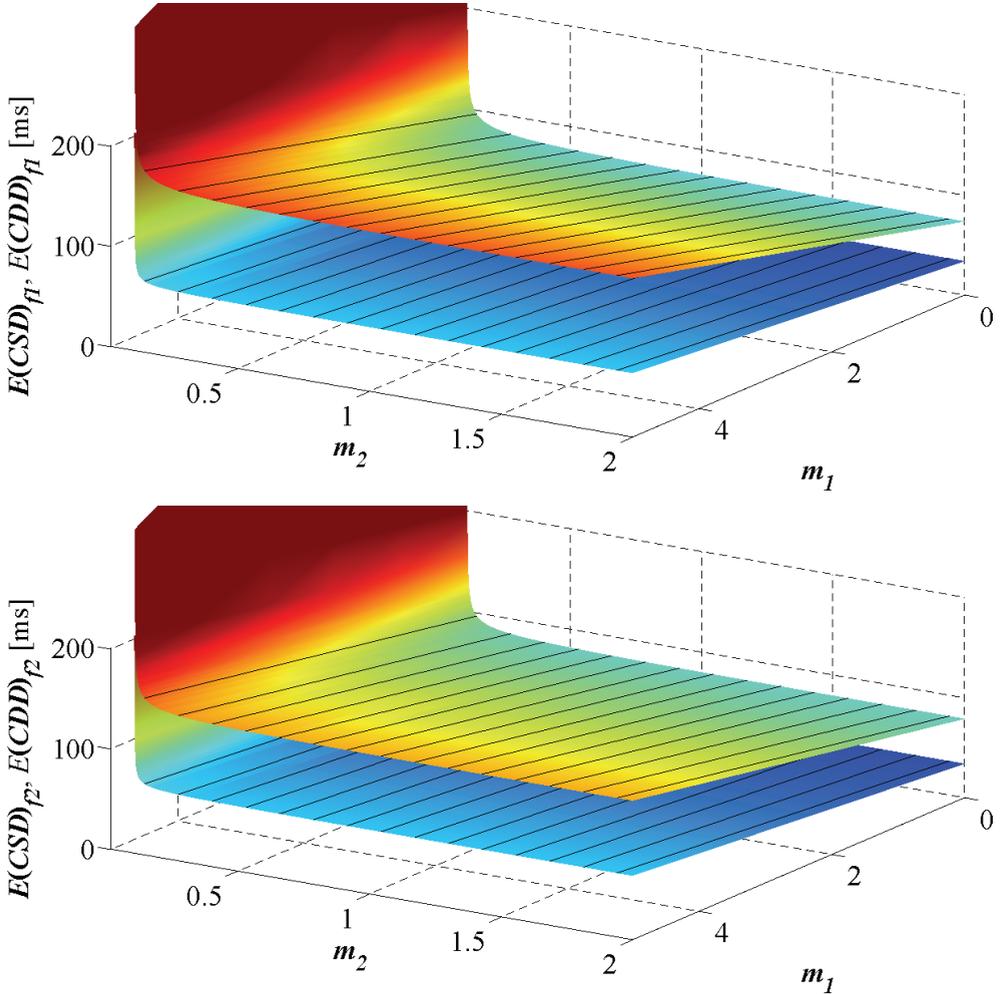


Fig. 8. Call processing performance results for inter-operator calls; link lengths and bandwidths in domain 1:  $d_1 = 200$  km,  $b_1 = 50$  Mb/s; link lengths and bandwidths in domain 2:  $d_2, b_2$ ;  $m_1 = d_2/d_1, m_2 = b_2/b_1$ , other input variables are default

In the first part of our experiments described in [10] we assumed that parameters of the elements and traffic levels in both IMS/NGN domains are the same (symmetric case). The research described in this paper is focused on the asymmetric cases in which value of one of the input variables in one of the domains is modified. This allows assessment of relations between parameters of one domain and call processing performance parameters for all types of calls originated in both domains.

The obtained results demonstrate that the above mentioned relations depend on the input variable, type of call processing performance parameter ( $E(CSD)$  or  $E(CDD)$ ), type of call (inter-operator or intra-operator) and its origination (the domain, in which the input variable is modified, or the other one). The details on this matter are presented and explained in the paper.

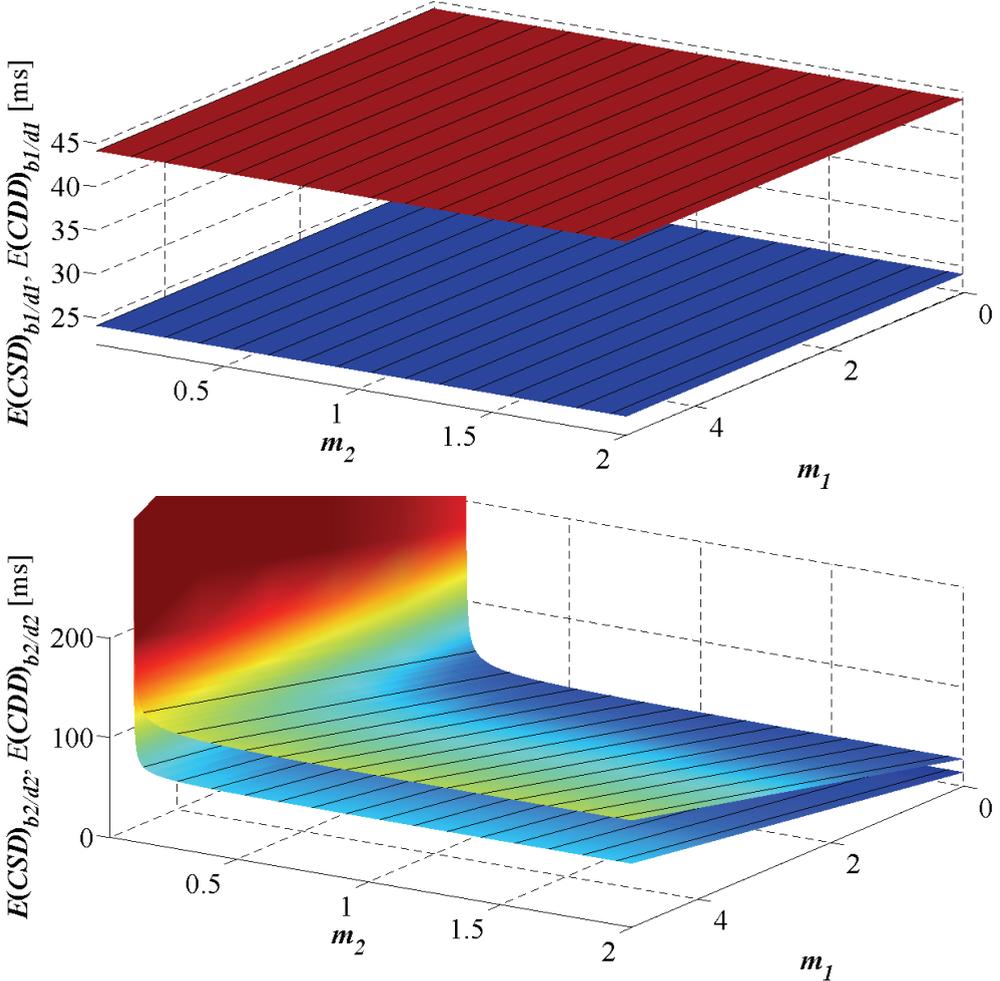


Fig. 9. Call processing performance results for intra-operator calls; link lengths and bandwidths in domain 1:  $d_1 = 200$  km,  $b_1 = 50$  Mb/s; link lengths and bandwidths in domain 2:  $d_2, b_2$ ;  $m_1 = d_2/d_1, m_2 = b_2/b_1$ , other input variables are default

Our future aim is to verify the described analytical model of a multidomain IMS/NGN using a proper simulation model, which is during development. The simulation model will also allow us to perform calculations using PH/PH/1 queuing mod-

els. These models offered good results in our previous research regarding a single domain of IMS/NGN [7,8], however, their application requires experimental message inter-arrival histograms obtained using simulations.

Table 1. Relations between parameters of domain 2 and call processing performance of both domains (inter-operator calls)

Input variable	Relation with			
	$E(CSD)_{f1}$	$E(CDD)_{f1}$	$E(CSD)_{f2}$	$E(CDD)_{f2}$
$\lambda_{R2}$	positive	positive	positive	positive
$\lambda_{22}$	positive	positive	positive	positive
$\lambda_{21}$	positive	positive	positive	positive
$T_{INV} * 2$	positive	positive	positive	positive
$E(X_{A2}) = E(X_{C2})$	positive	positive	positive	positive
$E(Y_2)$	positive	none	none	none
$d_2$	positive	positive	positive	positive
$b_2$	negative	negative	negative	negative
$r_{C2}$	positive (almost none)	positive (almost none)	positive (almost none)	positive (almost none)
$p_{A2} = p_{C2}$	negative (almost none)	negative (almost none)	negative (almost none)	negative (almost none)

Table 2. Relations between parameters of domain 2 and call processing performance of both domains (intra-operator calls)

Input variable	Relation with			
	$E(CSD)_{b1/d1}$	$E(CDD)_{b1/d1}$	$E(CSD)_{b2/d2}$	$E(CDD)_{b2/d2}$
$\lambda_{R2}$	none	none	positive	positive
$\lambda_{22}$	none	none	positive	positive
$\lambda_{21}$	positive	positive	positive	positive
$T_{INV} * 2$	none	none	positive	positive
$E(X_{A2}) = E(X_{C2})$	none	none	positive	positive
$E(Y_2)$	none	none	none	none
$d_2$	none	none	positive	positive
$b_2$	none	none	negative	negative
$r_{C2}$	none	none	positive (almost none)	positive (almost none)
$p_{A2} = p_{C2}$	negative (almost none)	negative (almost none)	negative (almost none)	negative (almost none)

## REFERENCES

- [1] *General overview of NGN*, ITU-T Recommendation Y.2001, December 2004.
- [2] *IP Multimedia Subsystem (IMS): Stage 2 (Release 12)*, 3GPP TS 23.228 v12.4.0, March 2014.
- [3] KACZMAREK S., SAC M., *Traffic Model for Evaluation of Call Processing Performance Parameters in IMS-based NGN*, In: Information Systems Architecture and Technology: Networks Design and Analysis, Grzech A., et al. (Eds.), Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 2012, 85–100.

- [4] KACZMAREK S., KASZUBA M., SAC M., *Simulation model of IMS/NGN call processing performance*, Gdańsk University of Technology Faculty of ETI Annals, Vol. 20, 2012, 25–36.
- [5] KACZMAREK S., SAC M., *Analysis of IMS/NGN call processing performance using G/G/1 queuing systems approximations*, Telecommunication Review and Telecommunication News (Przegląd Telekomunikacyjny i Wiadomości Telekomunikacyjne), No. 8–9, 2013, 702–710.
- [6] KACZMAREK S., SAC M., *Approximation of Message Inter-Arrival and Inter-Departure Time Distributions in IMS/NGN Architecture Using Phase-Type Distributions*, Journal of Telecommunications and Information Technology, No. 3, 2013, 9–18.
- [7] KACZMAREK S., SAC M., *Analysis of IMS/NGN call processing performance using phase-type distributions*, In: Information Systems Architecture and Technology: Network Architecture and Applications, Grzech A., et al. (Eds.), Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 2013, 23–39.
- [8] KACZMAREK S., SAC M., *Analysis of IMS/NGN Call Processing Performance Using Phase-Type Distributions Based on Experimental Histograms*, Bulletin of the Polish Academy of Sciences: Technical Sciences, submitted for publication, 2014.
- [9] KACZMAREK S., SAC M., WOŁONKIEWICZ M., *Call processing performance in multidomain IMS/NGN architecture*, In: Proc. ICT Young'2013, Gdańsk, Poland, pp. 87–94, 2013.
- [10] KACZMAREK S., SAC M., *Traffic model of a multidomain IMS/NGN*, Telecommunication Review and Telecommunication News (Przegląd Telekomunikacyjny i Wiadomości Telekomunikacyjne), accepted for publication, 2014.
- [11] *Call processing performance for voice service in hybrid IP networks*, ITU-T Recommendation Y.1530, November 2007.
- [12] *SIP-based call processing performance*, ITU-T Recommendation Y.1531, November 2007.
- [13] *Functional requirements and architecture of next generation networks*, ITU-T Recommendation Y.2012, April 2010.
- [14] *IMS for next generation networks*, ITU-T Recommendation Y.2021, September 2006.
- [15] KACZMAREK S., SAC M., *Traffic engineering aspects in IMS-based NGN networks (Zagadnienia inżynierii ruchu w sieciach NGN bazujących na IMS)*, In: Teleinformatics library, vol. 6. Internet 2011 (Biblioteka teleinformatyczna, t. 6. Internet 2011), Bem D. J., et al. (Eds.), Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 2012, 63–115 (in Polish).
- [16] KACZMAREK S., SAC M., *Traffic modeling in IMS-based NGN networks*, Gdańsk University of Technology Faculty of ETI Annals, Vol. 1, No. 9, 2011, 457–464.
- [17] *Resource and admission control functions in next generation networks*, ITU-T Recommendation Y.2111, November 2011.
- [18] ROSENBERG J., et al., *SIP: Session Initiation Protocol*, IETF RFC 3261, June 2002.
- [19] CALHOUN P., et al., *Diameter Base Protocol*, IETF RFC 3588, September 2003.
- [20] *Resource control protocol no. 1, version 3 – Protocol at the Rs interface between service control entities and the policy decision physical entity*, ITU-T Recommendation Q.3301.1, August 2013.
- [21] PIRHADI M., SAFAVI HEMAMI S. M., KHADEMZADEH A., *Resource and admission control architecture and QoS signaling scenarios in next generation networks*, World Applied Sciences Journal 7 (Special Issue of Computer & IT), 2009, 87–97.
- [22] CAMARILLO G., MARSHALL W., ROSENBERG J., *Integration of Resource Management and Session Initiation Protocol (SIP)*, IETF RFC 3312, October 2002.
- [23] OSOGAMI T., HARCHOL-BALTER M., *Closed form solutions for mapping general distributions to quasi-minimal PH distributions*, Performance Evaluation, Vol. 63, Iss. 6, 2006, 524–55.
- [24] BOBBIO A., HORVATH A., TELEK M., *Matching three moments with minimal acyclic phase type distributions*, Stochastic models, Vol. 21, Iss. 2–3, 2005, 303–326.
- [25] ABHAYAWARDHANA V. S., BABBAGE R., *A traffic model for the IP Multimedia Subsystem (IMS)*, In: Proc. IEEE 65th Vehicular Technology Conference, VTC2007, Dublin, Ireland, 2007.

Marcin CZAJKOWSKI\*, Artur GORCZYCA\*,  
Sylwester KACZMAREK\*, Krzysztof SZALAJDA\*,  
Paweł KACZMAREK\*\*

## **THE OPTICAL TRANSPORT NETWORK CONTROL BASED ON SDN ARCHITECTURE**

The aim of this publication is to present research results on the usability of the Software-Defined Networking concept to control transport networks. For this purpose, an easy-to-use connection scheduler was developed capable of controlling connections in optical transport networks. The authors would like to present this solution and details of constructed SDN architecture implemented for modern optical transport solutions based on ADVA Optical Networking's products. The results of testing the implemented applications and the created SDN architecture will also be presented.

### **1. INTRODUCTION**

Today's expectations of telecommunication networks are simplicity (configuration effort comparable with operating system user maintenance) as well as high performance. Increasingly, for economic and functional reasons solutions that integrate many services into a single network management application are preferred. This eliminates a problem of operations constraints related to different implementations of the same network management and control functions and enables unification of solutions. In some high level objectives, SDN (Software-Defined Networking) architecture attempts to address these problems [1]. SDN is a concept of building telecommunication networks which assumes the physical separation of the control plane from the data plane. One of the reasons for the introduction of the separation between the planes of

---

\* Gdańsk University of Technology, Faculty ETI, 80-233 Gdańsk, G. Narutowicza 11/12.

\*\* ADVA Optical Networking, 81-310 Gdynia, Śląska 35/37.

the network was to provide an administrator abstraction layer (additional software overlay) which hides the technical details used in the network.

This publication describes an adaptation of the SDN concept and OpenFlow protocol [2] [3] for a circuit switched network (lambda switched networks) control. Presented application "Scheduler" enables control of the physical layer of a communication network, using the OpenFlow protocol. Basic network configuration functions such as adding or removing connection, (mapped onto OpenFlow flow operation), are performed according to the schedule specified by the network administrator.

The rest of the publication is organized as follows. The second section presents the SDN concept, while the third section describes the extension of the controller - application "Scheduler" and its graphical user interface. The fourth section shows a solution which has been tested on a transport network realized on the basis of ADVA DWDM devices. The test results are presented in the fifth section, and the publication concludes with a summary.

## 2. SOFTWARE DEFINED NETWORKING CONCEPT

Software-defined networking (SDN) is an approach to computer networking which evolved from work done at UC Berkeley and Stanford University around 2008. The SDN network environment is controlled by the controller (Fig. 1), providing a centralized view of the entire network. The presented solution was based on the Floodlight controller [4] - written in Java, open source project implemented with the support of OpenFlow protocol, version 1.0. Through the controller, network administrators can quickly and easily make decisions on how switches, routers will handle the traffic. The most commonly used protocol in SDN networks that facilitates the communication between the controller (called the Southbound API) and the switches is currently OpenFlow.

One of the biggest challenges in the adaptation of SDN to the transport network control is the need to translate OpenFlow protocol messages (used to orchestrate Ethernet devices) onto existing control plane mechanisms used in the transport network (e. g. GMPLS). Therefore, two new objects are introduced – the central controller (PLC) that communicates with one or more abstract switches. The controller operates on an abstraction layer (implemented as a software component). In the presented research, control function is performed by issuing control messages to the respective abstract switch virtualizing transport domain using a dedicated protocol – OpenFlow.

An SDN environment also uses open, application programming interfaces (APIs) to support all the services and applications running over the network. These APIs, commonly called Northbound APIs [4], facilitate innovation and enable efficient service automation. As a result, SDN enables traffic shaping and deploys services to address

changing business needs, without having to touch each individual switch or router in the forwarding plane. In Floodlight controller the NBI is realized in the form of the REST API – a controller-specific interface that allows the user application to communicate with the controller.

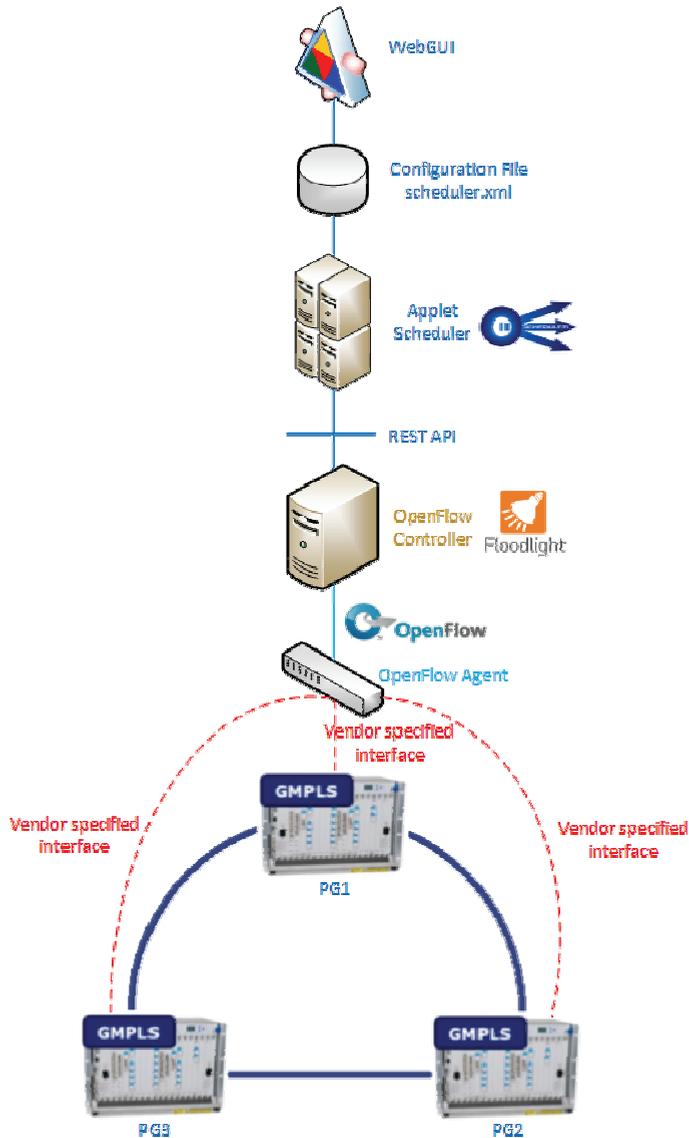


Fig. 1. Topology of the lab network

### 3. TRANSPORT NETWORK BASED ON WDM DEVICES

In the presented research, the transport network was based on ADVA Optical Networking's FSP 3000 R7 WDM devices (Fiber Service Platform). The network was built with three nodes (PG1, PG2, PG3) connected in a ring topology. Each of the FSP 3000 devices communicates with a prototype OpenFlow agent abstracting transport domain to provide basic control capabilities via SDN controller.

The following chapter contains a brief description of both ADVA WDM products and details of the built optical transport network.

#### 3.1. ADVA FSP 3000 R7 DWDM DEVICE

The ADVA FSP 3000 R7 [5] is a scalable WDM transport platform. It provides support for a variety of services, such as OTN, Ethernet (including 100 Gb/s), Fibre Channel, Infiniband and SDH. The FSP 3000 R7 provides a whole set of control and management plane technologies such as RayControl (GMPLS-based control plane), OSPF-based routing of management and control traffic, SNMP and TL1 as management protocols. The platform allows using up to 192 wavelengths per fiber pair with up to 50 GHz channel spacing. The platform modularity provides easy scalability and flexibility. One of the types of modules is the set of different channel modules which expose client services from the optical network. There are three main types of channel modules – core, access and enterprise. Moreover, the vendor also provides modules essential for proper fiber-optic system operation, such as ROADM multiplexers, optical amplifiers, optical filters, dispersion compensators, optical supervisory channel modules and protection modules.

The FSP 3000 R7 devices are widely used in modern high speed networks, in the core and access. ADVA Optical Networking also offers software for planning and managing networks built with FSP 3000 R7 nodes.

#### 3.2. ADVA OPENFLOW AGENT

This software component provides OpenFlow switch abstraction of the optical transport domain. The agent was created by ADVA Optical Networking as a result of research activity on the SDN concept within the OFELIA EU-funded project [6] and acts as an OpenFlow reference switch with all adaptations necessary for optical transport. Tributary ports of the optical devices are directly mapped onto the ports of an abstract switch. Some mechanisms, like physical topology discovery through LLDP packets wrapped into "ofpt\_packet\_in", "ofpt\_packet\_out", are not applicable due to the nature of optical transport domain and are hence disabled. The component provides an OpenFlow channel on top and vendor specific SBI towards the optical domain.

### 3.3. OPTICAL TRANSPORT NETWORK CONFIGURATION

The transport network, as mentioned, was built with three FSP 3000 R7 nodes named PG1, PG2 and PG3 connected in the ring topology. Each node was pre-configured with six configured transponder channel modules offering different types of services at 10 Gb/s rate (10GBE, OTU2, OTU2E, STM-64, 10GFC etc.). The modules were grouped into three and each connected to the one of 4-port DWDM channel splitter module. The output of filter is combined with optical supervisory channel (OSC) in optical supervisory channel filter (OSFM). The multiplexed signal from OSFM output is being sent to the others nodes via optical links (Fig. 2). The numbering of the ports shown in the figure below refers to the mapped virtual switch port. The ports 1 to 6 were assigned to PG1, the ports 7 to 12 to PG2 and, the ports 13 to 18 to PG3.

The communication between the FPS 3000 R7 and abstract virtual switch was realized by vendor specific interface. The connection has been established between NCU module (network element control unit) and abstract switch over internal IP network.

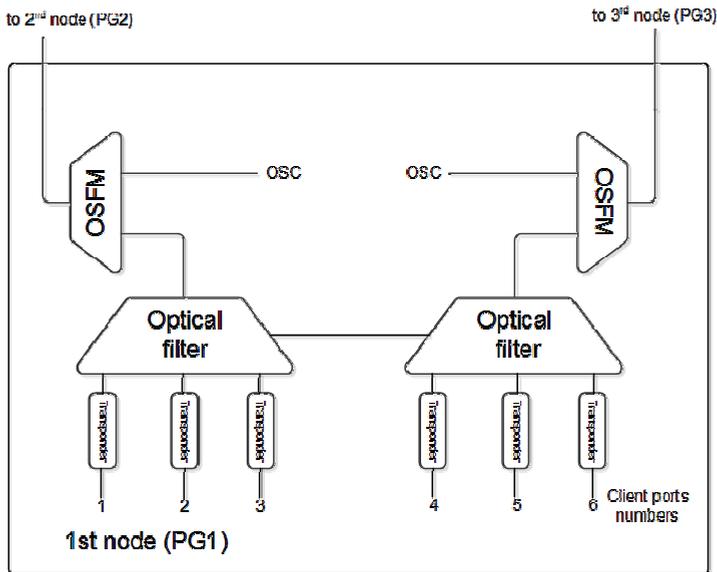


Fig. 2. Scheme of nodes modules configuration (for PG1 node)

## 4. SDN CONTROLLER EXTENSION – SCHEDULER

The “Scheduler” application, an extension of Floodlight controller, is a result of the research on the SDN concept. The development of the “Scheduler” allowed an introduction of connection schedule functionality into SDN architecture. This chapter in-

roduces the details of application design and the principle of its operation. A method of storing configuration and functionality of “Scheduler’s” graphical interface were also presented.

#### 4.1. “SCHEDULER” STRUCTURE

The “Scheduler” is composed of three functional blocks (Fig. 1): the core program written in Java, XML format configuration file and graphical web interface WebGUI written in PHP.

The “Scheduler’s” core is a standalone Java program packed to the *scheduler.jar* archive (runs independently of the controller). It has access to the configuration file and based on this configuration (details in sections 4.2 and 4.3) communicates with the SDN controller to push or delete flows. Communication takes place via the Floodlight API realized in REST technology [7]. It is an interface that allows obtaining information about state and parameters of controller and attached abstract switches. As two of its methods this API provides a push flow and a delete flow into the controller. REST API is an HTTP server listening on port 8080. The acceptable format of requests and responses is JSON.

In “Scheduler” XML and JSON support was implemented using external libraries *jdom-1.1.3.jar* [8] and *javax.json-1.0.4.jar* [9]. To implement HTTP protocol handling the *Apache HttpComponents* [10] libraries were used. As a part of research activity logging function was also implemented in “Scheduler”. Logs are stored in *scheduler\_log* folder.

The components of “Scheduler” application: executable file (*scheduler.jar*), configuration file (*scheduler.xml* by default, but it can be a loaded file with any other name) and logs folder (*scheduler\_log*) should be stored in the same directory.

An additional element created for the “Scheduler” is a graphical web interface WebGUI. It was created for easier editing and checking program configuration. WebGUI is a PHP based web service running on Apache 2 server. WebGUI makes changes to the configuration file, according to user commands.

#### 4.2. PRINCIPLE OF OPERATION

The main “Scheduler’s” task is to push flows at a specified time and delete these flows at another specified time. “Scheduler” is working on records from configuration files. Every record has three OpenFlow parameters (DPID – switch identifier, virtual switch input port, virtual switch output port) and four schedule parameters: start date, start time (exact hour and minute), end date and end time (exact hour and minute). There are three variants of schedule: *Single flow*, *Everyday flow* and *Selected day flow*. A similar approach was presented in [11].

In *Single flow* variant flow is pushed on a specified start date and at a specified start time and is deleted on a specified end date and at a specified end time. The entered end date cannot be earlier than the start date.

In *Everyday flow* variant flow is pushed and deleted daily. Start and end dates determine the period during which the flow will be pushed and deleted. The flow is pushed every day in this period at the start time and is deleted at the end time on the same day. In this variant, both the end date and time parameters cannot be earlier than the start date and start time.

*Selected days flow* variant acts as *Everyday flow* but in *Selected days flow* the user can select days of the week in which the flow will be pushed and deleted. Operation of this variant is possible due to the introduction of an additional parameter that stores the selected days of the week.

The mode of action above-mentioned variants is illustrated as a timeline in Fig. 3 for the following common parameters: start date – 12 May 2014, end date – 18 May 2014, start time – 6:00, end time - 18:00. For *Selected days flow* variant was chosen the following days: Tuesday, Saturday and Sunday. The grayed field indicates activation time of example flow.

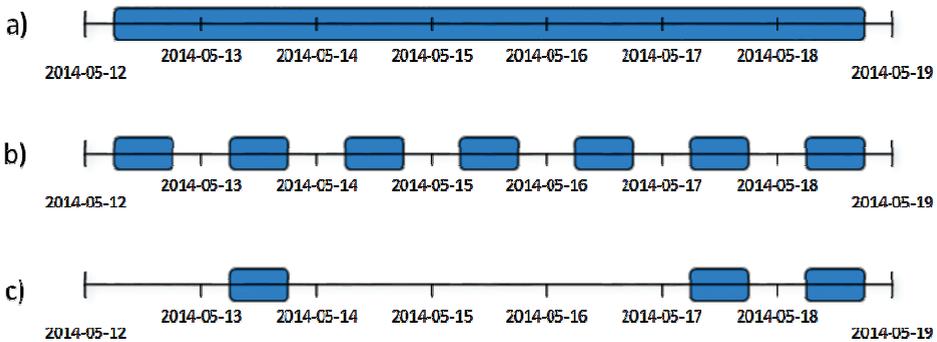


Fig. 3. Example modus operandi for: a) *Single flow*, b) *Everyday flow*, c) *Selected days flow*

Any more advanced schedule configurations require more than one record in the configuration file. Such records are stored in the configuration file *scheduler.xml*, which can be modified via the WebGUI or directly by the user. The configuration file is processed in the “Scheduler” every 10 seconds. After checking the correctness of the file syntax and parameters format, “Scheduler” makes decisions which flow in configuration file should be active or not at this moment. In the next step, “Scheduler” query Floodlight controller about the state of processing flows. If the flow should be active and it isn’t pushed, “Scheduler” makes the request to the controller to push this flow. If the flow duration expires and it is still active in the controller, the “Scheduler” makes this request to delete this flow. In other cases the requests are not sent. Moreo-

ver, records that have been ended (end date and end time passes) will be removed from the configuration file.

#### 4.3. STORAGE OF CONFIGURATION

Schedule records are stored in XML file. Default filename of this file is *scheduler.xml*. The file structure is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<connect>
  <flow>
    <date_start>20140201</date_start>
    <date_end>20140214</date_end>
    <hour_start>17</hour_start>
    <minutes_start>08</minutes_start>
    <hour_end>17</hour_end>
    <minutes_end>15</minutes_end>
    <when>1234567</when>
    <DPID>00:00:00:00:00:00:01</DPID>
    <name>f8e629cc4b6959de</name>
    <port_in>1</port_in>
    <port_out>2</port_out>
    <delete>0</delete>
  </flow>
  <flow>
    ...
  </flow>
  ...
</connect>
```

Meaning of presented tags is as follow:

- <connect> – XML root;
- <date\_start> – start date of record;
- <date\_end> – end date of record;
- <hour\_start> – start hour of record;
- <minutes\_start> – start minute of record;
- <hour\_end> – end hour of record;
- <minutes\_end> – end minute of record;
- <when> – digits that describes flow activation days: 0 – *Single flow* variant, 1 – Mondays, 2 – Tuesdays, and so on;
- <DPID> – virtual switch ID;
- <name> – unique flow ID;
- <port\_in> – flow's input port;
- <port\_out> – flow's output port;
- <delete> – deleting record on demand: 1 – delete, 0 – do not delete.

Each of the presented tags must appear in every record. Tags content format is strictly defined and is validated during configuration file processing by the “Scheduler”.

The list of attributes of the entry could be potentially extended with additional packets filtering policies, however packet based filtering has no applicability to transport technologies.

#### 4.4. WEBGUI – THE GRAPHICAL INTERFACE

In order to facilitate the operations with “Scheduler” and configuration file, it was decided to develop a webpage graphical user interface – WebGUI (Fig. 4). This is a PHP-based interface running on Apache 2 HTTP server.

WebGUI functions include adding record to schedule in all three variants, deleting records on user demand, records listing and preview log files. The interface also has a “Scheduler” run indicator.



Fig. 4. WebGUI layout

## 5. RESULTS OF TESTS

During the implementation phase and after its completion a number of tests were performed. The aim was to find all potential application errors. The tests were divided into two main stages: WebGUI and application tests.

### 5.1. WEBGUI TESTS

The tests verified all the functions of the WebGUI: adding flow (enter of: date start, date end, select days, time start, time end, DPID, port in and point out), delete flow, show flows (list of all flows in the configuration file), show logs (list of all application logs which is information about creating and deleting flows; there is information about errors of flows parameters too).

All these function are prior to approval check, that all information entered are proper before creating flow. If the data is incorrect WebGUI prevents the creation of a flow.

The tests showed a proper WebGUI reaction on the correct and false data input. For the correct data WebGUI carries out the users command. Set incorrect data or incorrect use WebGUI resulted in an error message and inability to approve changes for incorrect data.

### 5.2. APPLICATION TESTS

The application has been tested in a circuit switched network (lambda switched network). The network worked in a ring topology (Fig. 1).

At the start, the application checks the correctness of the input data and the structure of the configuration file. If the configuration file is correct, it is loaded into the application. Then it checks the status of the current connections and in the next step adds new flows. For the tests, multiple flows between the various ports of the abstract switch were introduced. They resulted in multiple services between different tributary ports of the optical nodes in the circuit switched network. The network behaved properly.

### 5.3. SUMMARY OF TESTS RESULTS

Tests confirmed the correct operation of the application and the WebGUI. Only proper connections with correct data input were created in the network. Creating a false connection was impossible. The application provided proper control of a circuit switched network.

## 6. SUMMARY

*Open Networking Foundation* contributes to improving and popularizing the conception of SDN and OpenFlow protocol. The publication presents implementation of a network control application using the OpenFlow protocol. It proved the applicability of control of the transport network in accordance with the SDN concept. The abstrac-

tion layer enables centralized control of any number of networks in which it is possible to implement OpenFlow protocol. The designed application uses the scheduled flows for the creation of services in the transport network. In order to eliminate errors of network administrators, the application and user interface has been equipped with an input validation control, in order to eliminate the creation of false flows.

The research was an adaptation attempt of the SDN conception to a circuit switched network control. The developed application and presented tests showed that SDN conception provides control simplicity and hides huge amounts of information about transport resources in the network, which may be important for carrier class solutions. On the other hand, this simplicity may be desired by some well-scoped data center oriented use cases

The presented solution was also tested on a virtual transport network based on Mininet simulator [12].

#### ACKNOWLEDGMENT

This work has been partially supported by Statutory Funds of Electronics, Telecommunications and Informatics Faculty, Gdansk University of Technology and partially by the European Regional Development Fund under the Innovative Economy Operational Programme, project POIG.01.04.00-22-084/12-00 „RINGO – stworzenie prototypu zintegrowanej platformy sieci wielowymiarowych”



**INNOVATIVE  
ECONOMY**  
NATIONAL COHESION STRATEGY

**EUROPEAN UNION**  
EUROPEAN REGIONAL  
DEVELOPMENT FUND



#### REFERENCES

- [1] Open Networking Foundation, *SDN Conception*, <https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf> (Accessed 14 May 2014).
- [2] Open Networking Foundation, *OpenFlow Whitepaper*, <http://archive.openflow.org/documents/openflow-wp-latest.pdf> (Accessed 14 May 2014).
- [3] Open Networking Foundation, *OpenFlow 1.0 Specifications*, <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.0.0.pdf> (Accessed 14 May 2014).
- [4] Project Floodlight, <http://www.projectfloodlight.org/floodlight/> (Accessed 14 May 2014)
- [5] ADVA Optical Networking, *FSP3000R7 Data Sheet*, [http://www.advaoptical.com/~media/Resources/Data%20Sheets/FSP\\_3000.ashx](http://www.advaoptical.com/~media/Resources/Data%20Sheets/FSP_3000.ashx)
- [6] <https://alpha.fp7-ofelia.eu/cms/assets/Publications-and-Presentations/2012-03-08-ONTC-Autenrieth-OFELIA-final.pdf> (Accessed on June 24, 2014)
- [7] WANG K-C., LAPPAS P., *Floodlight REST API*, <http://www.openflowhub.org/display/floodlightcontroller/Floodlight+REST+API> (Accessed May 13, 2014).

- [8] HUNTER J., *JDOM*, <http://www.jdom.org/> (Accessed May 13, 2014).
- [9] KOTAMRAJU J., *JSON Processing*, <https://jsonp.java.net/> (Accessed May 13, 2014).
- [10] Apache Software Foundation, *Apache HttpComponents*, <https://hc.apache.org/> (Accessed May 13, 2014).
- [11] FIGUEROLA S., CIULLI N., DE LEENHEER M., DEMCHENKO Y., ZIEGLER W., BINCZEWSKI A., *PHOSPHORUS: single-step on-demand services across multi-domain networks for e-science*, Proceedings of the SPIE, Volume 6784, Network Architectures, Management, and Applications V, article id. 67842X, November 19, 2007
- [12] CZAJKOWSKI M., GORCZYCA A., KACZMAREK S., SZALAJDA K., KACZMAREK P., *Realizacja aplikacji Scheduler dla sterowania siecią transportową*, XXX Krajowe Sympozjum Telekomunikacji Teleinformatyki (KSTiT 2014), September 3–5, 2014, Poznań (accepted paper).

Mariusz GŁĄBOWSKI\*, Michał Dominik STASIAK\*

## **POINT-TO-POINT BLOCKING IN SWITCHING NETWORKS WITH OVERFLOW LINKS SERVICING SELECTED TRAFFIC CLASSES**

In the chapter, a new method of the point-to-point blocking probability calculation in multiservice switching networks with overflow links has been presented. The proposed method is an extension of the PPBMT (Point-to-Point Blocking for Multichannel Traffic) method which is used for point-to-point blocking probability calculation in multiservice switching networks without overflow links. The proposed method belongs to the class of methods known as the effective availability methods. In the chapter, a method of the effective availability calculation for 3-stage Clos switching networks has been proposed. Special attention is paid to the method for determining the model of overflow link. The results of the analytical calculations are compared with the data of the discrete events simulations of the switching networks with chosen structures.

### 1. INTRODUCTION

Modern telecommunications and computer networks are designed in such a way as to accommodate traffic composed of a number of classes with differentiated demands and perform service appropriately. Call classes corresponding to real-time services, such as Voice over IP (VoIP) [1] and Dynamic Voice over IP (DVoIP) [2], are becoming steadily more and more important. For the proper execution of real-time services the network should satisfy the requirements of appropriate Quality of Service (QoS) parameters primarily related to acceptable delay. Even slight delays in

---

\* Chair of Communication and Computer Networks, Poznan University of Technology, Polanka 3, 60-965 Poznań, Poland.

transmission can have detrimental effect on the performance of the network and result in disruption in transmission perceivable for the end user.

The fundamental network devices responsible for the speed and quality of transmission are switching devices that are arranged in nodes, e.g. routers or switches. The main element in these devices that is responsible for the quality of switching is the switching network. There exist three basic methods for decreasing the internal blocking probability in switching networks [4]: rearrangement and repacking [5] as well as the application of overflow links [6]. Though rearrangement and repacking do not require any interference into the physical structure of the network, they significantly overload the controlling device, which in consequence prolongs the time needed for a connection to be set up. The application of overflow links, though requires interference into the network structure (introduction of additional inputs and outputs to switches), has no influence on the load of the controlling device. Their introduction is followed by a multiple decrease in blocking probability, both in single-service switching network [6] and multi-service switching network [7, 8, 9, 10].

The basis for modelling of switching networks with overflow links is provided by the so-called effective availability methods that were originally developed to analyze single-service networks [11–13]. Later, these methods were applied to modelling multi-service networks. Then, on the basis of the effective availability method, models of switching networks with overflow links have been also developed [8, 9, 10, 14–16]. The first analytical models assumed infinite capacity of overflow links. Such an approach made it possible to significantly simplify the algorithm for modeling switching networks with overflow links. The next stage of the study involved a development of a method that allowed the point-to-point blocking probability in a network with required capacity of overflow links to be determined [9]. Additionally, a model that allows us to determine the point-to-group blocking probability in the switching network with overflow links to which selected traffic classes are directed has been developed [10]. Switching networks with the point-to-point selection and overflow links with finite capacity have not been analyzed yet. This article presents an analytical model of the network with point-to-point selection and overflow links with some predefined capacity that services selected classes of call.

The chapter is structured as follows. Section 2 presents the structure of a three-stage multi-service Clos network with a system of overflow links for selected classes of call. Section 3 provides a description of the Point-to-Point Blocking for Multichannel Traffic (PPBMT) method for modeling multi-service networks without overflow links. Section 4 presents an analytical model of a network with overflow links that operates within the point-to-point selection mode. This model allows the capacity of overflow links to be determined and characteristics of traffic streams directed to these links to be designated. Section 5 provides a comparison of the analytical results with the results obtained during the simulation experiments for some selected structures of switching networks with overflow links. Section 6 sums up the results of the study presented in the article.

## 2. MULTI-SERVICE SWITCHING NETWORK WITH OVERFLOW LINKS

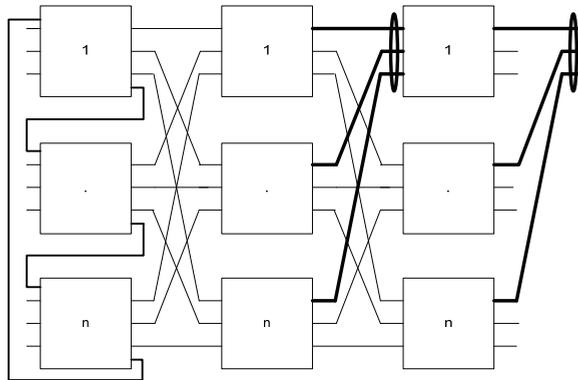


Fig. 1. The structure of the 3-stage switching network with overflow links

Our research object is a Clos switching network composed of symmetrical switches  $n \times n$  in each of the stages. The outputs of the switches of the last stage are grouped into directions according to the numbering of outputs of each of the switches, i.e. all outputs with the index  $i$  create the outgoing direction of the switching network. The assumption is that the capacity of all the links in the network, input, output and inter-stage links, is  $f$  Basic Bandwidth Units (BBU) [17]. The switching network can operate in either point-to-point selection or point-to-group mode. In the point-to-group mode the controlling device attempts to set up a connection between a given input of the input switch and any (randomly selected) output of a given direction. In the point-to-point selection mode the controlling device attempts to set up a connection between a given input of the input switch and the selected output in a given direction. If there is no connection path between the required input and the selected output of the switching network, then the call is rejected due to internal blocking. If all outputs of a given direction are occupied, then the call is lost due to external blocking.

Subsequently, let us consider a switching network with overflow links. Overflow links have been introduced to the first stage of the network. This means that switches with one additional output and input are used in the first stage. The additional output of each of the switches of the first stage is connected to the additional input of the next switch, while the additional output of the last switch of the first stage is connected with the additional input of the first switch in the first stage. Simulation studies have proved that such a structure of the overflow system is most favorable (optimum) as compared to other structures that use overflow links in the first stage and next stages [8]. This means that in the case of a required load of the switching network, the decrease in internal blocking in the presented configuration of overflow links is the highest. The model under scrutiny also assumes any capacity of the overflow links and that this capacity can be different from the capacity of the remaining links of the

network. The algorithm for setting up connections in the network with overflow links is as follows: first the control algorithm checks if there is a possibility of setting up a connection without the application of overflow links between the required, free input and the output. If there is not at least one free connecting path available (internal blocking), the algorithm checks whether the overflow link is free for a call of a given class. If so, the control algorithm attempts to set up a connection using a neighboring switch of the first stage. In the case where this connection cannot be set up from the neighboring switch of the first stage to a given (required) output, the call is rejected due to internal blocking. In design, the use of overflow links in the proposed algorithm for setting up connections is purposefully restricted to just one link. The simulation study has shown that an increase in the number of overflow links in setting up one call does not lead to any substantial decrease in the internal blocking probability [7].

### 3. MODEL OF THE NETWORK WITHOUT OVERFLOW LINKS – THE PPBMT METHOD

The PPBMT method (Point-to-Point Blocking for Multichannel Traffic) [18] has been developed for the purpose of analytical determination of the blocking probability in the multi-service switching network operating in the point-to-point selection mode. The basic assumption for the PPBMT method is the assumption adopted in [19] viz. that the point-to-point blocking probability in  $n$ -stage switching network is the same as the point-to-group blocking probability in  $(n-1)$  stage switching network. In the PPBMT method the model of the studied  $n$ -stage network is reduced in its essence to the  $(n-1)$ - stage network in which all inter-stage links that lead to one switch of  $n$ -stage are treated as a direction.

The point-to-point internal blocking probability for calls of class  $i$  in a 3-stage switching network will be then exactly the same as the point-to-group blocking probability in a 2-stage network. Therefore, in line with the PPBMT method, we can write:

$$E_{int}(i) = \sum_{s=0}^{v-d(i)} P(i, s \wedge 1) \left[ \binom{v-s}{d(i)} / \binom{v}{d(i)} \right], \quad (1)$$

where:

- $E_i(i)$  – internal point-to-point blocking probability for calls of class  $i$ ,
- $V$  – capacity of a given direction (the number of links leading to one switch of the third stage),
- $P(i, s \wedge 1)$  – distribution of available links in a switch,
- $d(i)$  – effective availability of the 2-stage switching network for calls of class  $i$ .

The distribution  $P(i, s \wedge 1)$  determines probability of an event in which  $s$  incoming links, and at the same time at least one outgoing link of a given switch, are available for a class  $i$  call. Distribution of available links in the last stage switch of the 2-stage network is approximated by the distribution of free links in the so-called limited availability group (LAG [20]):

$$P(i, s \wedge 1) = \frac{\sum_{x=0}^V P(i, s|x)[1-P(i, 0|x)][P_{V-x}]_V}{1 - \sum_{n=0}^k [\sum_{x=0}^V P(i, n|x)P(i, 0|x)[P_{V-x}]_V]}, \quad (2)$$

where  $[P_n]_V$  is the distribution of the occupancy in LAG, whereas  $P(i, s|x)$  is the conditional distribution of available output links, determined with the assumption that  $x$  BBUs in LAG are free:

$$P(i, s|x) = \frac{\binom{V}{s} \sum_{w=st_i}^w F(w, s, f, t_i) F(x-w, v-s, f, t_i-1.0)}{F(v, s, f, 0)}, \quad (3)$$

The limited availability group is a system composed of  $v$  separated links, each having the same capacity equal to  $f$  BBUs. The assumption is that a call of a given class can be serviced only when LAG has at least one link that can service this call. As a consequence, a given call cannot be shared between resources of several links. The LAG model precisely maps the method for servicing calls by a given direction in the real switching network. Distribution of occupied links in LAG can be approximated by the following recurrent formula:

$$n[P_n]_V = \sum_{i=1}^M A_i t_i \zeta_i(n - t_i) [P_{n-t_i}]_V, \quad (4)$$

where:

- $A_i$  – traffic intensity of class  $i$  offered to a given direction,
- $t_i$  – the number of BBUs required to set up a connection of class  $i$ ,
- $M$  – the number of traffic classes of multi-service traffic offered in LAG,
- $V$  – capacity LAG expressed in BBU, i.e.  $V = vf$
- $\zeta_i(n)$  – conditional transition probability between states  $n$  and  $(n + t_i)$  for a call stream of class  $i$ ,
- $[P_n]_V$  – occupancy probability of  $n$  BBU in LAG.

The conditional transition probability  $\zeta_i(n)$  is defined by the ratio between favorable arrangements of free resources in a given system that make service of a call of class  $i$  possible and all possible arrangements of free resources:

$$\zeta_i(n) = \frac{F(V-n, v, f, 0) - F(V-n, v, t_i-1, 0)}{F(V-n, v, f, 0)}, \quad (5)$$

where  $F(x, v, f, z)$  is the number of arrangements of  $x$  free BBUs in LAG, i.e. in  $v$  links with the capacity  $f$  BBUs each. The accompanying assumption is that in each link there are  $z$  free BBUs. The parameter  $F(x, v, f, z)$  can be determined in a combinatorial way using the following formula:

$$F(x, v, f, z) = \sum_{i=0}^{\lfloor \frac{x-vt}{f-z+1} \rfloor} (-1)^i \binom{v}{i} \binom{x - v(z-1) - 1 - i(f-z+1)}{v-1}. \quad (6)$$

The external blocking probability  $E_{ext}(i)$  in the switching network occurs when all links of the output direction are busy. Thus:

$$E_{ext}(i) = \sum_{n=v(f-t_i+1)}^V [P_n]_V \{1 - \zeta_i(n)\}. \quad (7)$$

The total blocking probability  $E(i)$  in the PPBMT method is determined by the formula:

$$E(i) = E_{ext}(i) + E_{int}(i)[1 - E_{ext}(i)]. \quad (8)$$

In order to estimate the internal blocking probability on the basis of (1) it is necessary to define the effective availability parameter for each traffic class offered in the switching network. In a given state of a blocking network, only part of output links is available for a given call. Effective availability is defined as the average value of available output links for a given input switch in the switching network. The effective availability parameter is determined on the basis of the so-called equivalent network [18]. The equivalent network for calls of class  $i$  is a single-service network with identical topological structure as that of the multi-service network. The assumption is that the capacity of each of the links of the equivalent network is equal to 1 BBU. Another assumption is that load of a single link in the equivalent network is equal to the blocking probability for calls of class  $i$  in a link of the real switching network. The load of the link in the equivalent network is called the fictitious load. Effective availability for a three-stage network in the point-to-point selection mode is equal to the effective availability of a two-stage network in the point-to-group selection mode and can be expressed by the formula [18, 26]:

$$d_e(i) = [1 - \pi(i)]v + \pi(i)b(i), \quad (9)$$

where:

$d_e(i)$  – effective availability for calls of class  $i$  in 2-stage equivalent network,

$\pi(i)$  – probability of direct non-availability for calls of class  $i$ ,

$v$  – the number of links in a given direction of the equivalent network (the number of links that lead to one switch of the third stage – Fig. 1)

$b(i)$  – fictitious load of the inter-stage link in the equivalent network.

The fictitious load parameter for the inter-stage link for calls of class  $i$  in the equivalent network is equal, according to the adopted definition of the equivalent network, to the blocking probability for calls of class  $i$  in one inter-stage link of the real network. This parameter can be determined on the basis of the Kaufman–Roberts distribution [22, 23]:

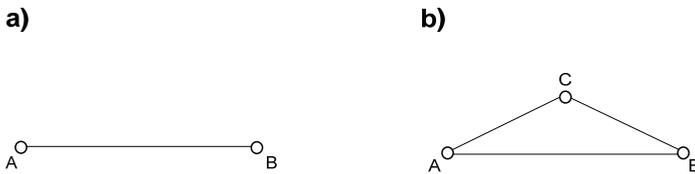
$$n[P_n]_f = \sum_{i=1}^M a_i t_i [P_{n-t_i}]_f, \tag{10}$$

$$b(i) = \sum_{n=f-t_i+1}^f P[n]_f, \tag{11}$$

where  $a_i$  is traffic offered to one (single) link of the multi-service network. In the considered 3-stage Clos switching network (Fig. 1), full symmetry of traffic is assumed. This means that traffic offered to each of the input switches is identical. In the network under scrutiny, the number of inter-stage links between successive stages is  $v^2$ . Hence, traffic offered to one (a single) inter-stage link is equal to:

$$a_i = \frac{vA_i}{v^2} = \frac{A_i}{v}, \tag{12}$$

The probability of direct non-availability for a given traffic class can be interpreted as the blocking probability for all possible connecting paths between switches of external stages in the equivalent network. This parameter is determined on the basis of the analysis of the probability graph using the Lee method [24, 25].



Rys. 2. Graph of the network without overflow links (a), with overflow links (b)

In the case of the PPBMT method, while considering a 3-stage multi-service network, we have to create an equivalent network for a 2-stage network. The probability graph of such a network is presented in Fig. 2a. Eventually, on the basis of the Lee method, the probability of direct non-availability in the two-stage network will be equal to:

$$\pi(i) = b(i). \tag{13}$$

At this point, on the basis of (9)–(13), it is possible to determine the effective availability in the switching network without overflow links, and then, on the basis of Formulas (1,7,8), the internal, external and total blocking probabilities can be determined.

#### 4. MODEL OF A NETWORK WITH OVERFLOW LINKS – A MODIFIED PPBMT METHOD

The introduction of overflow links changes the structure of the multi-service switching network and its corresponding equivalent network. Traffic that would be rejected in the network without overflow links due to internal blocking is directed to overflow links. Thus changed structure of the switching network leads to a change in the probability graph, in which edges that correspond to overflow links appear. The fictitious load of such an edge in the equivalent network is equal, on the basis of the adopted definition, to the blocking probability of calls of a given class in a real overflow link of the multi-service network. Because traffic offered to overflow links has a peaked traffic pattern, then in order to determine the blocking probability for a given class of calls (the fictitious load of the edge of the probability graph that corresponds to the overflow link), it is necessary to develop an appropriate model of the overflow link.

##### 4.1. MODEL OF THE OVERFLOW LINK

In the considered network model we assume the total symmetry of traffic. We can then assume that the value of traffic of class  $i$  offered to one direction of the two-stage multiservice network (Fig. 1) is  $A_i$ , whereas the value of traffic of class  $i$  offered to all directions in the network is  $vA_i$ . Knowing that traffic in overflow links results from the internal blocking of the network without overflow link, the average value of overflow traffic offered to one overflow link can be determined on the basis of the following formula:

$$R_i = v \frac{A_i}{v} E_{int}^0(i) = A_i E_{int}^0(i), \quad (14)$$

where  $E_{int}^0(i)$  is the internal blocking probability of calls of class  $i$  in the switching network without overflow links. To analyze the overflow link we can use Hayward's model [30]. According to this model, first the average value and the overflow traffic variance are determined. The average value is defined by Formula (14). We can assume that traffic of class  $i$  overflows to overflow links from a fictitious group  $V^*$

[27] that services exclusively traffic of this particular class. The parameter  $V_i^*$  is determined with the assumption that traffic of class  $i$  that overflows to an overflow link has the same average value and variance as traffic of class  $i$  blocked by internal blocking in the switching network without overflow links:

$$E_{V_i^*}(A_i) = E_{int}^0(i), \quad (15)$$

where  $E_{V_i^*}(A_i)$  is the blocking probability in the full-availability group with the capacity  $V_i^*$  and offered traffic  $A_i$  (Erlang model [28]). Having determined the capacity of the fictitious group  $V_i^*$ , it is possible to determine, on the basis of Riordan formulas, the variance of overflow traffic of class  $i$  [29], [30]:

$$\sigma_i^2 = R_i \left[ \frac{A_i}{V_i^* + 1 - A_i + R_i} + 1 - R_i \right], \quad (16)$$

The peakedness factor  $Z$  is defined by the formula:

$$Z_i = \sigma_i^2 / R_i. \quad (17)$$

The parameters  $(R_i, Z_i)$  are the basis for the evaluation of the blocking probability for calls of class  $i$  in the overflow link. This probability is determined on the basis of the generalized Hayward method [31]. In this method, first the so-called aggregated degeneration coefficient  $Z$  is to be determined. This coefficient is a weighted sum of degeneration coefficients for individual classes in which weights are directly proportional to the participation of traffic of a given class in multi-service traffic:

$$Z = \sum_{i=1}^M Z_i \frac{R_i t_i}{\sum_{j=1}^M R_j t_j}, \quad (18)$$

To determine the blocking probability in an overflow link, the generalization of Hayward's formula for the link servicing overflow multi-service traffic is used:

$$n[P_n]_{f_0/Z} = \sum_{i=1}^M \frac{R_i}{Z_i} t_i [P_{n-t_i}]_{f_0/Z}, \quad (19)$$

$$p(i) = \sum_{n=\binom{f_0}{Z}-t_i+1}^{f_0/Z} P[n]_{f_0/Z}. \quad (20)$$

where  $f_0$  is the capacity of the overflow link and  $p(i)$  is the blocking probability of calls of class  $i$  in the overflow link.

#### 4.2. AVAILABILITY IN THE NETWORK WITH OVERFLOW LINKS

The  $\pi(i)$  parameter, i.e. the probability of the direct non-availability for calls of class  $i$  in the network with overflow link, is determined on the basis of the probability graph shown in Fig. 2b. Therefore, we can write:

$$\pi(i) = b(i)^{2-p(i)}. \quad (21)$$

At this point, on the basis of (9) we are in position to determine the effective availability parameter in the switching network with overflow links, and then the internal blocking probability (Formula (1)), external blocking probability (Formula (2)) and the total blocking probability (Formula (3)).

### 5. A STUDY OF MULTI-SERVICE POINT-TO-POINT SWITCHING NETWORKS WITH OVERFLOW LINKS

Since the proposed model of the switching network is an approximated model, in order to examine its accuracy the results of the analytical calculations for blocking probabilities in the network with overflow links were compared with the results of the simulation experiments. A specially-designed software developed by the authors makes it possible to perform analytical calculations and simulation experiments for different structural parameters of the network and different parameters of the offered mixture of traffic.

A three-stage network with overflow Clos links that were composed of  $5 \times 5$  switches in the first stage and  $4 \times 4$  in the second and third stages was selected for the study. Input, output, inter-stage and overflow links had the same capacity equal to 30 BBU. The network operated in the point-to-point selection mode. Offered traffic was composed of 3 classes of Erlang calls that demanded respectively 1, 2 and 6 BBUs to set up a connection. The assumption was that the proportion of offered traffic was 1:1:1.

Figure 3 shows the analytical results of the internal point-to-point blocking probability in the studied switching network with and without overflow links for three classes of calls. The graphs indicate that introduction of overflow links in the switching network leads to the significant decrease in the internal blocking probability for particular traffic classes. This decrease can even reach one order of magnitude.

Figure 4 shows the analytical and simulation results for the internal point-to-point blocking probability in the studied switching network for three classes of calls. We assumed in the considered scenario that all the three classes of calls could make use of overflow links. Figure 5 presents the results for the scenario in which only one class of

calls (class 1) can be serviced by overflow links. Comparing Figure 4 and 5 it can be stated that the scenario with one class of calls serviced by overflow links leads to significant decrease in the internal blocking probability of this class of calls. Simultaneously, the blocking probability of others classes, that are not serviced by overflow links, increases.

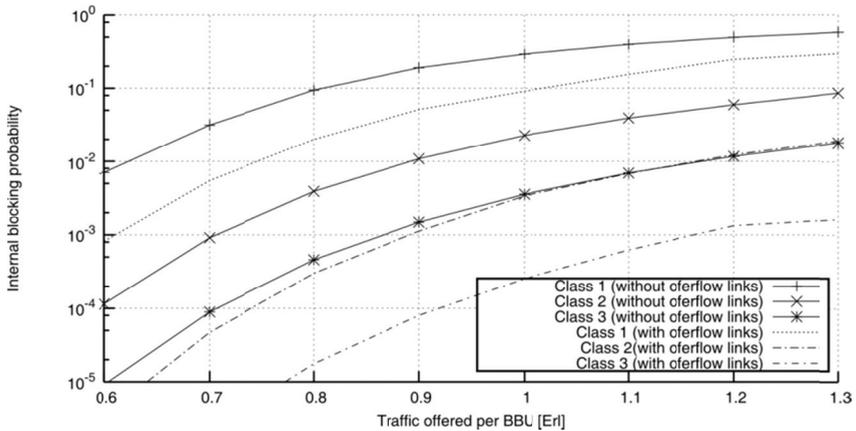


Fig. 3. Internal blocking in 3-stage network with and without overflow links

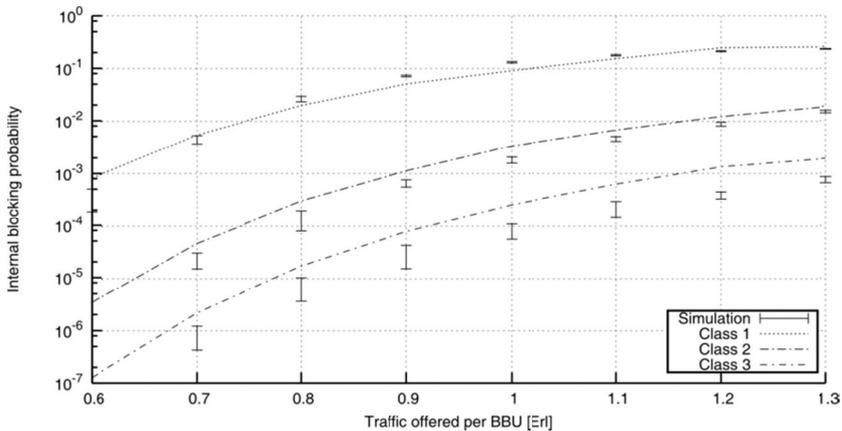


Fig. 4. Internal blocking in 3-stage network with overflow links for three classes

To ensure high credibility of presented results each result of the simulation experiment amounts to 10 series, 100,000 calls each. The obtained results take into account 99% confidence interval calculated on the basis of the Student distribution.

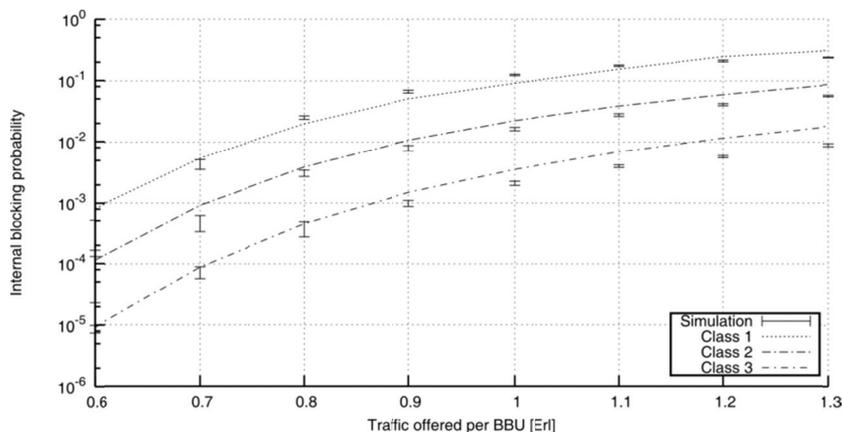


Fig. 5. Internal blocking in 3-stage network with overflow links limited to the first class

## 6. CONCLUSIONS

This article presents a modification of the PPBMT method that allows blocking in multiservice Clos networks with overflow links to be evaluated. This method includes Hayward's model that allows characteristics of overflow links to be investigated. The model presented in the article makes it also possible to evaluate traffic parameters of the network in the case where traffic directed to overflow links to selected call classes is restricted. The results of the simulation study indicate high accuracy of the proposed method.

The study carried out by the authors indicate the relevance and applicability of overflow links in the switching network that operates in the point-to-point selection mode. The presented results also show that a restriction of overflow links to traffic of a given class leads to the maximum decrease in the internal blocking probability for this call class. It should be pointed out at this point, however, that such a solution has an adverse effect on internal blocking probabilities for traffic of other call classes. Nonetheless, in considerations of services in real time this approach can be justified and effective though and can eventually lead to significant improvements in the operation of the network as perceived by the end user for whom a possible slight deterioration of the service in other traffic classes will hardly be noticeable.

## REFERENCES

- [1] KHASNABISH B.: *Implementing Voice over IP*, Wiley, 2003.
- [2] SUN, L., MKWAWA, I., JAMMEH, E., IFEACHOR, E.: *Guide to Voice and Video over IP*, Springer, 2013

- [3] CLOS C. *A study of non-blocking switching networks*, Bell System Technical Journal, Vol. 32, No. 2, 1953, pp. 406–424.
- [4] JAJSZCZYK A., *Wstęp do telekomunikacji*, WNT, Warszawa, 1998.
- [5] KABACIŃSKI W.: *Nonblocking Electronic and Photonic Switching Fabrics*, Springer, 2005.
- [6] FORTET R. (Ed.): *Calcul d'orange, Systeme Pentaconta*, L.M.T., Paris, 1961.
- [7] STASIAK M. D., ZWIERZYKOWSKI P., *Performance study in multi-rate switching networks with additional inter-stage links*, Proc. of The Seventh Advanced International Conference on Telecommunications (AICT 2011), St. Marteen, Holland, 2011, pp. 141–146.
- [8] STASIAK M. D., ZWIERZYKOWSKI P., *Multi-service switching networks with overflow links*, Image Processing and Communications, Vol. 15, No. 2, 2010, pp. 61–71.
- [9] GŁĄBOWSKI M., STASIAK M. D.: *Multi-stage Switching Networks with Overflow Links for a Single Call Class* [in]: Advances in Intelligent Systems and Computing, Vol. 233 Image Processing and Communications Challenges 5, Springer, pp. 335–344, 2013.
- [10] GŁĄBOWSKI M., STASIAK M. D.: *Modelling of multiservice switching networks with overflow links for any traffic classes*, Circuid Divisions Systems, 2014 (accepted for publication)
- [11] BININDA N., WENDT W.: *Die effektive Erreichbarkeit für Abnehmerbündel hinter Zwischenleitungsanlagen*, Nachrichtentechnische Zeitschrift, Heft 11, No. 12, 1959, pp. 579–585.
- [12] CHARKIEWICZ A.D.: *An approximate method for calculating the number of junctions in a crossbar system exchange*, Elektrosvyaz, No. 2, 1959, pp. 55–63.
- [13] LOTZE A.: *Bericht über verkehrstheoretische untersuchungen* CIRB, Inst. für nachrichtenvermittlung und datenverarbeitung der technischen hochschule, Univ. of Stuttgart, 1963, pp. 1–42.
- [14] GŁĄBOWSKI M., STASIAK M. D., *Internal Blocking Probability Calculation in Switching Networks with Additional Inter-Stage Links and Engset Traffic*, Proc. of 8th IEEE, IET Int. Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), Poznań, Poland, 2012.
- [15] GŁĄBOWSKI M., STASIAK M. D., *Internal Blocking Probability Calculation in Switching Networks with Additional Inter-Stage Links* [in]: Information Systems Architecture and Technology, Vol. Service Oriented Networked Systems, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2011, pp. 279–288.
- [16] GŁĄBOWSKI M., STASIAK M. D., *Internal Blocking Probability Calculation in Switching Networks with Additional Inter-stage Links and mixture of Erlang and Engset Traffic* Image Processing and Communications, Vol. 16, No.1, pp. 61–73, 2012.
- [17] ROBERTS J., MOCCI V., VIRTAMO I. (Eds.), *Broadband Network Teletraffic*, Final Report of Action COST 242, Berlin, Commission of the European Communities, Springer, 1996.
- [18] STASIAK M., *Combinatorial considerations for switching systems carrying multichannel traffic streams*, Annals of Telecommunications, Vol. 51, No. 11–12, pp 611–625, 1996.
- [19] LOTZE A., RODER A., THIERER G., PPL – a reliable method for the calculation of point-to-point loss in link systems, Proc. 8th Int. Teletraffic Congress, Melbourne, 1976, pp. 547/1–44.
- [20] STASIAK M., *Blocking probability in a limited-availability group carrying mixture of different multi-channel traffic streams*. Ann. Telecommun., Vol. 48, No. 1–2, 1993, pp. 71–76.
- [21] STASIAK M., GŁĄBOWSKI M., WIŚNIEWSKI A., and ZWIERZYKOWSKI P. *Modeling and Dimensioning of Mobile Networks*. Wiley, 2011.
- [22] ROBERTS J.: *A service system with heterogeneous user requirements*, in Performance of data communications systems and their applications, Ed. Pujolle G., North Holland Pub Co, Amsterdam, 1981, pp. 423–431.
- [23] KAUFMANN J.S., *Blocking in a shared resource environment*, IEEE Trans. Commun., Vol. COM-29, No. 10, 1981, pp. 1474–1481.

- [24] LEE C.Y., *Analysis of switching networks*, Bell Syst. Techn. J., Vol. 34, No. 6, 1955, pp. 1287–1315.
- [25] STASIAK M.: *Blockage interne point a point dans les reseaux de connexion*, Ann. Telecommun., Vol. 43, No. 9–10, 1988, pp. 561–575.
- [26] GŁĄBOWSKI M., *Modelowanie systemów Muti-rate ze strumieniami zgłoszeń BPP*, Wydawnictwo Politechniki Poznańskiej 2009
- [27] HUANG Q., IVERSEN V., *Approximation of loss calculation for hierarchical networks with multiservice overflows*, IEEE Transactions on Communications, Vol. 56, no. 3, 2008, pp. 466–473.
- [28] ERLANG A., *Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges*, Elektrotechniker, Vol. 13, 1917.
- [29] WILKINSON R.: *Theories of toll traffic engineering in the USA*, Bell System Technical Journal, Vol. 35, No. 2, 1956, pp. 421–514.
- [30] FREDERICKS A., *Congestion in blocking systems – a simple approximation technique*, Bell System Technical Journal, Vol. 59, No. 6, 1980, pp. 805–827.
- [31] GŁĄBOWSKI M., KUBASIK K., STASIAK M., *Modeling of systems with overflow multi-rate traffic*, Telecommunication Systems, Vol. 37, No. 1–3, Springer 2008, pp. 85–96.

Remigiusz RAJEWSKI\*

## **QUALITY OF OPTICAL CONNECTIONS IN THE MBA(N, e, 2) SWITCHING NETWORKS**

In this article, it is described how to calculate the first-, the second-, and the third-order crosstalk stage-by-stage in a new MBA(N, e, 2) switching fabric. All results for this network are compared with a typical *baseline* (one of the best known from the topic of this subject) and Beneš switching networks. The new architecture gives in many cases better optical signal-to-crosstalk ratio than other ones. What is more, it is also described how the optical signal, representing some connection in the switching network, goes through a whole structure – what kind and what number of passive and active optical elements are used. These numbers are compared also with results given by typical switching networks of the same capacity and functionality. The MBA(N, e, 2) fabric has, in many cases, fewer number of such elements what allows to decrease a cost of the whole architecture.

### 1. INTRODUCTION

In the switching theory, besides Clos architecture [1], very often *banyan* switching network [2] is used. The *banyan* network is also called *baseline* switching network. It is because both, *banyan* and *baseline* networks, are topologically equivalent to each other [3]. Another name for the *baseline* structure also used in a subject of switching theory is the  $\log_2 N$  switching network [4], where  $N$  denotes the capacity of such an architecture. The capacity of switching network corresponds strictly to the number of inputs and outputs in such a structure.

The  $\log_2 N$  switching network is build from symmetrical optical switching elements of size  $2 \times 2$ . In general, optical switching element can has size  $d \times d$ . Then, this structure is called the  $\log_d N$  switching network [5]. Such an architecture was

---

\* Faculty of Electronic and Telecommunications, Poznan University of Technology, 3 Polanka St., 60-965 Poznan, Poland, (e-mail: remigiusz.rajewski@put.poznan.pl).

later extended to another switching network of different functionalities [6], [7], and [8]. More architectures were described in details in [3], [9], and [10].

The optical switching fabric build from asymmetrical and symmetrical optical switching elements was proposed in [11], [12], and it is called the  $\log_2 N - 1$  switching network. This structure constitutes a particular case of the  $MBA(N, e, 2)$  switching fabric [13]. These two structures constitutes better alternative (cheaper solution, where the cost is a metric of comparison different structures) for almost all range of capacity of the switching fabric than the *baseline* architecture of the same functionality and capacity. Detailed description how the  $\log_2 N - 1$  and the  $MBA(N, e, 2)$  networks are build and how to extend them to structures of greater capacities are included in [11], [12], and [13], respectively. A simple example of  $MBA(14, 4, 2)$  switching fabric is shown in Figure 1.

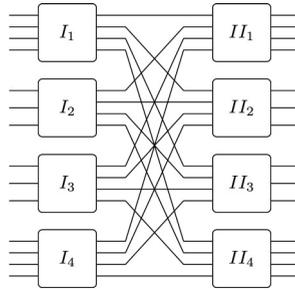


Fig. 1. The  $MBA(14, 4, 2)$  optical switching fabric

To compare different structures very often the cost of the switching fabrics is used. As a cost it could be used the number of optical switching elements. However, to be more accurate, the number of passive and active optical elements, from the optical switching elements are build, are used to exact a cost of the whole switching fabric. Because different structures are build from different numbers of these elements, this architecture, which is build from fewer number of such elements, is cheaper/better solution.

As a passive optical elements there are used splitters (Ss) and combiners (Cs) at the input and at the output side of the optical switching element, respectively. In turn, as an active optical elements the semiconductor optical amplifiers (SOAs) [14], [15] are used. In Figure 2(a) and Figure 2(b) optical switching elements of size  $2 \times 2$  and  $3 \times 3$  are shown, respectively. The first one is build from 4 SOAs, 2 optical splitters of size  $1 \times 2$ , and 2 optical combiners of size  $2 \times 1$ . The second one is build from 9 SOAs, 9 splitters of size  $1 \times 2$ , and 9 combiners of size  $2 \times 1$ . It should be noted that three splitters and three combiners in Figure 2(b) are not fully connected to SOAs – there used only one output and one input, respectively, and the second one is free. In Figure 2 splitters are denoted as red triangles directed in the left side (they are placed at the input side of the optical switching element), combiners are denoted as red trian-

gles directed in the right side (they are placed at the output side of the optical switching element), and semiconductor optical amplifiers are denoted as green rectangles.

The new  $MBA(N, e, 2)$  structure, described in details in [13], is very competitive to the *baseline* network. Therefore, the goal of this paper is to check what kind of quality is offered for connections in the new switching fabric and how they look in comparison to the *baseline* ones.

This paper is organized as follows. In section 2 the architecture is described. In the next section quality of connection is defined and calculated for both the *baseline* and the  $MBA(N, e, 2)$  switching fabrics. In this section comparison of these networks is also given. In the last section conclusions are given.

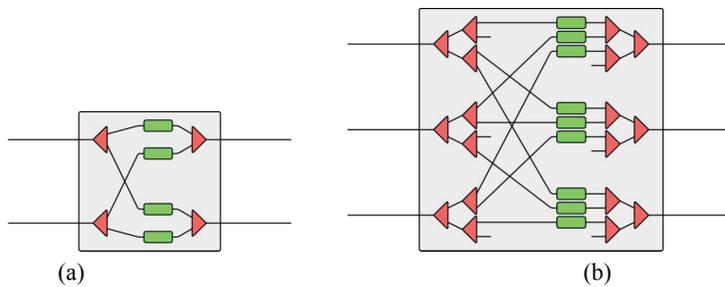


Fig. 2. The optical switching element of size: (a)  $2 \times 2$ , (b)  $3 \times 3$

## 2. ARCHITECTURE OF THE SWITCHING FABRIC

The optical network can be build using different technology. In this article each architecture is build using SOAs, however, each of them can be produced in another technology. What is important here is: the combinatorial properties, the number of optical elements which are used to build some architecture, and the number of such elements through which the optical signal goes. Decreasing the number of optical elements for the same capacity and leaving the same functionality allows to achieve cheaper solution. In turn, decreasing the number of optical elements (passive or active) through which the optical signal goes allows to increase the quality of this signal when it appears at the output of the switching network. It should be noted, that using this same technology in different architectures allows to compare them between each other and it also allows to put a main focus on the structure itself.

The strength of switching fabric build from SOAs is such an active elements have a quite good bandwidth [14], [15]. Each SOA can be tuned up separately to compensate all losses which occurs in one optical switching element (OSE). Such losses can came strictly from SOA and also from the passive element – splitter in this case which di-

vide optical power of the useful signal. Thanks to SOAs all optical signals occurred at the output of the optical switching element have the same optical power.

Different switching structures are build from different number of OSEs. These OSEs are connected together and the proper switching network is then created. Optical switching elements, used to achieve some switching fabric, can be identical – all have the same capacity and they are symmetrical. OSEs can have also different sizes and capacities and they can be asymmetrical and symmetrical, too. Thus, some structures are cheaper another are more expensive, where as a cost the number of passive and active optical elements is used. More information about cost and comparison of different structures between each other can be found in [12] and [13].

### 3. QUALITY OF CONNECTIONS

When some signal appears at input side of the switching fabric it means that some optical signal appears in OSE in the first stage. This signal has optical power  $P_{in}$ . When optical signal leaves OSE the optical power is then  $P_{out}$ . Regarding to the previous section, optical signal is amplified inside OSE by SOA and then  $P_{out} \approx P_{in}$ . This is true for the useful optical signal.

To be a little bit more precisely, it can be said that each SOA works like a typical ON-OFF switch. When this switch is ON, the optical signal goes through it (this signal is in fact amplified), when it is in OFF state, the optical signal does not go through this switch (it is attenuated very strong). However, the attenuation is not perfect and some part of the optical signal goes through such a switch. Such a signal is then regarded as not useful optical signal or just simpler as a noise. Therefore, it could be said that optical signal has then power  $P_{noise} = mP_{in}$ , where  $m = 0.01$  means interstice between  $P_{in}$  and  $P_{noise}$  (it is true only for one connection established in range of one OSE). To measure useful signal and noise the optical signal-to-crosstalk ratio (OSXR) is used [12], [16], [17], and [18]. The general expression for OSXR is as follows:

$$OSXR(type\_of\_network) = 10 \log_{10} \frac{P_{out}}{P_{noise}} [dB]. \quad (1)$$

Because  $P_{out} \approx P_{in}$  and  $P_{noise} = mP_{in}$  then equation (1) can be modified to the following one:

$$OSXR(type\_of\_network) = 10 \log_{10} \frac{P_{in}}{mP_{in}} = 10 \log_{10} \frac{1}{m} = X [dB]. \quad (2)$$

Applying  $m = 0.01$  to equation (2) it gives  $X = 20 \text{ dB}$ .

Table 1. Stage-by-stage crosstalk in the  $\log_2 8$  optical switching network

OSE	Output	After stage $s_1$	After stage $s_2$	After stage $s_3$
1	1	$P_1 + mP_2$	$P_1 + m(P_2+P_3) + m^2P_4$	$P_1 + m(P_2+P_3+P_5) + m^2(P_4+P_6+P_7) + m^3P_8$
	2	$P_2 + mP_1$	$P_3 + m(P_1+P_4) + m^2P_2$	$P_5 + m(P_1+P_6+P_7) + m^2(P_2+P_3+P_8) + m^3P_4$
2	1	$P_3 + mP_4$	$P_5 + m(P_6+P_7) + m^2P_8$	$P_3 + m(P_1+P_4+P_7) + m^2(P_2+P_5+P_8) + m^3P_6$
	2	$P_4 + mP_3$	$P_7 + m(P_5+P_8) + m^2P_6$	$P_7 + m(P_3+P_5+P_8) + m^2(P_1+P_4+P_6) + m^3P_2$
3	1	$P_5 + mP_6$	$P_2 + m(P_1+P_4) + m^2P_3$	$P_2 + m(P_1+P_4+P_6) + m^2(P_3+P_5+P_8) + m^3P_7$
	2	$P_6 + mP_5$	$P_4 + m(P_2+P_3) + m^2P_1$	$P_6 + m(P_2+P_5+P_8) + m^2(P_1+P_4+P_7) + m^3P_3$
4	1	$P_7 + mP_8$	$P_6 + m(P_5+P_8) + m^2P_7$	$P_4 + m(P_2+P_3+P_8) + m^2(P_1+P_6+P_7) + m^3P_5$
	2	$P_8 + mP_7$	$P_8 + m(P_6+P_7) + m^2P_5$	$P_8 + m(P_4+P_6+P_7) + m^2(P_2+P_3+P_5) + m^3P_1$

In case when there are more connections set up in one optical switching element the noise generated by them influences to other connections. The worst case for the *baseline* switching network of capacity  $N = 8$  (build from OSEs of size  $2 \times 2$ ) is presented in Table 1. As a worst case it should be understand the maximal number of possible set up connections in one OSE for each section  $s_j$ , where  $j = 1, 2, \dots, n$ , and  $n = \log_d N$  denotes the number of sections in the *baseline* network. More information about *baseline* network can be found in [4], and [5]. Therefore, for mentioned  $\log_2 8$  structure, there are set up following connections:  $\langle 1, 1 \rangle$ ,  $\langle 2, 5 \rangle$ ,  $\langle 3, 3 \rangle$ ,  $\langle 4, 7 \rangle$ ,  $\langle 5, 2 \rangle$ ,  $\langle 6, 6 \rangle$ ,  $\langle 7, 4 \rangle$ , and  $\langle 8, 8 \rangle$ . Of course, there is possibility of different set of connections, however, the number of such connections will be exactly the same as in the mentioned set of established connections. In turn, in Table 2 it is presented the worst case for the MBA(14, 4, 2) switching fabric for the following set of connections:  $\langle 1, 1 \rangle$ ,  $\langle 2, 5 \rangle$ ,  $\langle 3, 8 \rangle$ ,  $\langle 4, 11 \rangle$ ,  $\langle 5, 2 \rangle$ ,  $\langle 6, 6 \rangle$ ,  $\langle 7, 12 \rangle$ ,  $\langle 8, 3 \rangle$ ,  $\langle 9, 9 \rangle$ ,  $\langle 10, 13 \rangle$ ,  $\langle 11, 4 \rangle$ ,  $\langle 12, 7 \rangle$ ,  $\langle 13, 10 \rangle$ , and  $\langle 14, 14 \rangle$ . For both examples presented in Table 1 and Table 2 the  $P_1$  denotes the power of optical signal from input 1,  $P_2$  denotes the power of optical signal from input 2,  $P_3$  denotes the power of optical signal from input 3, and so on.

Generalizing and assuming that the power of the optical signal at any input of the  $\log_2 16$  switching network is  $P_I$ , the power of the optical signal at any output of this structure is equal to:

$$P_O = P_I(1 + 4m + 6m^2 + 4m^3 + m^4). \quad (3)$$

For  $\log_4 16$  the power of any input optical signal is also  $P_I$ , however, the power of optical signal at any output is:

$$P_O = P_I(1 + 6m + 9m^2). \quad (4)$$

In turn, for MBA(14, 4, 2) the power of any input optical signal is  $P_I$  and  $P_O$  is:

$$P_O = P_I(1 + 6m + 7m^2), \quad (5)$$

for MBA(28, 4, 2)  $P_O$  is:

$$P_O = P_I(1 + 5m + 7m^2 + 3m^3), \quad (6)$$

for MBA(56, 4, 2)  $P_O$  is equal to:

$$P_O = P_I(1 + 6m + 12m^2 + 10m^3 + 3m^4), \quad (7)$$

and for MBA(112, 4, 2) is equal to:

$$P_O = P_I(1 + 7m + 18m^2 + 22m^3 + 13m^4 + 3m^5). \quad (8)$$

In brackets there are components which corresponds to: useful optical signal (1), the first-order crosstalk ( $m$ ), the second-order crosstalk ( $m^2$ ), the third-order crosstalk ( $m^3$ ), the fourth-order crosstalk ( $m^4$ ), etc. It should be also noted that interstice between  $P_{in}$  and  $P_{noise}$  is quite small ( $m = 0.01$ ) therefore the second-, the third-, and so on, order crosstalk can be just omitted (because it is order of magnitude smaller than the first-order crosstalk). Thus, the final optical signal-to-crosstalk ratio for different structures can be found as follows.

Based on equations (3), (4) and on information included in [12], [16], [17] the optical signal-to-crosstalk ratio for *baseline* architecture of capacity  $N$  can be found using the following formula:

$$OSXR(\log_2 N) = X - 10 \log_{10}((d - 1)\log_d N). \quad (9)$$

For Beneš network OSXR is equal to [16], [17]:

$$OSXR(Beneš) = X - 10 \log_{10}(2\log_2 N - 1). \quad (10)$$

Based on equations (5), (6), (7), and (8) it can be found a general formula for OSXR in the MBA( $N, e, 2$ ) switching network:

$$OSXR(MBA(N, e, 2)) = \begin{cases} X - 10 \log_{10}(2e) & \text{for } N \leq N_0 \\ X - 10 \log_{10}(n' + e - 2) & \text{for } N > N_0 \end{cases}, \quad (11)$$

where  $n' = \left\lceil \log_2 \left[ \frac{N}{N_0} \right] + 2 \right\rceil$ , and  $N_0 = e^2 - e + 2$ .  $N_0$  denotes the basic capacity of the MBA( $N, e, 2$ ) switching fabric. More details of this capacity are included in [13].

Cases for the OSXR in the MBA( $N, e, 2$ ) switching fabric in expression (11) reflect strictly from architecture of this network [13].

Table 2. Stage-by-stage crosstalk in the  $MBA(14, 4, 2)$  optical switching network

OSE	Output	After stage $s_1$	After stage $s_2$
1	1	$P_1 + m(P_2+P_3+P_4)$	$P_1 + m(P_2+P_3+P_4+P_5+P_8+P_{11}) + m^2(P_6+P_7+P_9+P_{10}+P_{12}+P_{13}+P_{14})$
	2	$P_2 + m(P_1+P_3+P_4)$	$P_5 + m(P_1+P_6+P_7+P_8+P_{11}) + m^2(P_2+P_3+P_4+P_9+P_{10}+P_{12}+P_{13}+P_{14})$
	3	$P_3 + m(P_1+P_2+P_4)$	$P_8 + m(P_1+P_5+P_9+P_{10}+P_{11}) + m^2(P_2+P_3+P_4+P_6+P_7+P_{12}+P_{13}+P_{14})$
	4	$P_4 + m(P_1+P_2+P_3)$	$P_{11} + m(P_1+P_5+P_8+P_{12}+P_{13}+P_{14}) + m^2(P_2+P_3+P_4+P_6+P_7+P_9+P_{10})$
2	1	$P_5 + m(P_6+P_7)$	$P_2 + m(P_1+P_3+P_4+P_6+P_{12}) + m^2(P_5+P_7+P_{11}+P_{13}+P_{14})$
	2	$P_6 + m(P_5+P_7)$	$P_6 + m(P_2+P_5+P_7+P_{12}) + m^2(P_1+P_3+P_4+P_{11}+P_{13}+P_{14})$
	3	<i>not used</i>	$P_{12} + m(P_2+P_6+P_{11}+P_{13}+P_{14}) + m^2(P_1+P_3+P_4+P_5+P_7)$
	4	$P_7 + m(P_5+P_6)$	<i>not present</i>
3	1	$P_8 + m(P_9+P_{10})$	$P_3 + m(P_1+P_2+P_4+P_9+P_{13}) + m^2(P_8+P_{10}+P_{11}+P_{12}+P_{14})$
	2	<i>not used</i>	$P_9 + m(P_3+P_8+P_{10}+P_{13}) + m^2(P_1+P_2+P_4+P_{11}+P_{12}+P_{14})$
	3	$P_9 + m(P_8+P_{10})$	$P_{13} + m(P_3+P_9+P_{11}+P_{12}+P_{14}) + m^2(P_1+P_2+P_4+P_8+P_{10})$
	4	$P_{10} + m(P_8+P_9)$	<i>not present</i>
4	1	$P_{11} + m(P_{12}+P_{13}+P_{14})$	$P_4 + m(P_1+P_2+P_3+P_7+P_{10}+P_{14}) + m^2(P_5+P_6+P_8+P_9+P_{11}+P_{12}+P_{13})$
	2	$P_{12} + m(P_{11}+P_{13}+P_{14})$	$P_7 + m(P_4+P_5+P_6+P_{10}+P_{14}) + m^2(P_1+P_2+P_3+P_8+P_9+P_{11}+P_{12}+P_{13})$
	3	$P_{13} + m(P_{11}+P_{12}+P_{14})$	$P_{10} + m(P_4+P_7+P_8+P_9+P_{14}) + m^2(P_1+P_2+P_3+P_5+P_6+P_{11}+P_{12}+P_{13})$
	4	$P_{14} + m(P_{11}+P_{12}+P_{13})$	$P_{14} + m(P_4+P_7+P_{10}+P_{11}+P_{12}+P_{13}) + m^2(P_1+P_2+P_3+P_5+P_6+P_8+P_9)$

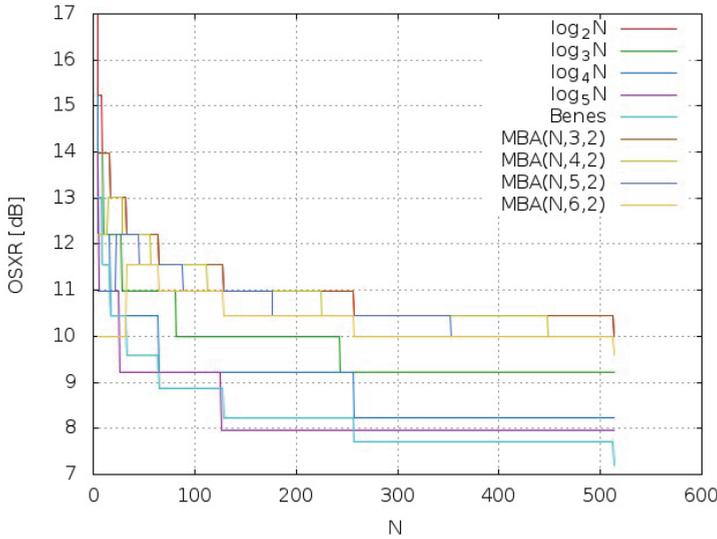


Fig. 3. The optical signal-to-crosstalk ratio (OSXR) for different kind of structures

In Figure 3 comparison of OSXR's for Beneš, *baseline* and  $MBA(N, e, 2)$  switching networks is presented. Results for the  $\log_2 N$  and the  $MBA(N, 3, 2)$  switching fabrics are identical therefore both are denoted by the red color line (dark and bright, respec-

tively). For other networks it can be seen that the  $MBA(N, e, 2)$  architecture has better OSXR in comparison to Beneš or *baseline* network builds from different size of OSE ( $d > 2$ ). What is more, if the size of OSE used to build the  $MBA(N, e, 2)$  grows ( $e > 3$ ), OSXR has greater value, too. It should be also noted that the  $MBA(N, e, 2)$  switching fabric has his “basic capacity”  $N_0$  (it was described in details in [13]) and for such a size of a whole architecture result can be worst than for *baseline* or Beneš architectures. Detailed information about OSXR, the number of stages as well as what number of active (semiconductor optical amplifiers) and passive (splitters and combiners) optical elements are on the route of any connection in the Beneš,  $\log_2 N$ ,  $MBA(N, 4, 2)$ ,  $MBA(N, 5, 2)$ , and  $MBA(N, 6, 2)$  type of switching networks are included in Table 3, Table 4, Table 5, and Table 6, respectively.

What is worth to mention, and what is reflecting from properties of the architecture of the  $MBA(N, e, 2)$  switching fabric [13], is that optical signal, which goes through this network, goes through one stage less. In result, one less SOA is needed on the way of the optical signal which represents some connection. It is very important in optical implementation of switching architecture, especially when active optical elements are more expensive than passive ones. Cost of switching networks is not a topic of this paper, however, it can be found in [11], [12], and [13].

Table 3. The OSXR, the number of stages, passive and active optical elements through which goes connection in the worst case in the Beneš and the  $\log_2 N$  switching networks

$N$	Kind of network	Number of stages	Splitters $1 \times 2$	Combiners $2 \times 1$	SOAs	OSXR [dB]
2–8	Beneš	5	5	5	5	13.01
	$\log_2 N$	3	3	3	3	15.23
9–16	Beneš	7	7	7	7	11.55
	$\log_2 N$	4	4	4	4	13.98
17–32	Beneš	9	9	9	9	10.46
	$\log_2 N$	5	5	5	5	13.01
33–64	Beneš	11	11	11	11	9.59
	$\log_2 N$	6	6	6	6	12.21
65–128	Beneš	13	13	13	13	8.86
	$\log_2 N$	7	7	7	7	11.55
129–256	Beneš	15	15	15	15	8.24
	$\log_2 N$	8	8	8	8	10.97
257–512	Beneš	17	17	17	17	7.70
	$\log_2 N$	9	9	9	9	10.46

Table 4. The OSXR, the number of stages, passive and active optical elements through which goes connection in the worst case in the  $MBA(N, 4, 2)$  switching network

$N$	Number of stages	Splitters $1 \times 2$ s	Combiners $2 \times 1$	SOAs	OSXR [dB]
4–14	2	4	4	2	12.22
15–28	3	5	5	3	13.01
29–56	4	6	6	4	12.22
57–112	5	7	7	5	11.55
113–224	6	8	8	6	10.97
225–448	7	9	9	7	10.46
449–512	8	10	10	8	10.00

Table 5. The OSXR, the number of stages, passive and active optical elements through which goes connection in the worst case in the  $MBA(N, 5, 2)$  switching network

$N$	Number of stages	Splitters $1 \times 2$ s	Combiners $2 \times 1$	SOAs	OSXR [dB]
5–22	2	6	6	2	10.97
23–44	3	7	7	3	12.22
45–88	4	8	8	4	11.55
89–176	5	9	9	5	10.97
177–352	6	10	10	6	10.46
353–512	7	11	11	7	10.00

Table 6. The OSXR, the number of stages, passive and active optical elements through which goes connection in the worst case in the  $MBA(N, 6, 2)$  switching network

$N$	Number of stages	Splitters $1 \times 2$ s	Combiners $2 \times 1$	SOAs	OSXR [dB]
6–32	2	6	6	2	10.00
33–64	3	7	7	3	11.55
65–128	4	8	8	4	10.97
129–256	5	9	9	5	10.46
257–512	6	10	10	6	10.00

#### 4. CONCLUSIONS

In this paper, the quality of connections in the  $MBA(N, e, 2)$  switching fabric (proposed in [13]) is described. Quality is defined here as the optical signal-to-crosstalk ratio (OSXR). Such an approach allows to compare a new switching network with already known from literature (Beneš and *baseline*). It is also given step-by-step what crosstalk appears in each stage for the worst case in compared structures. The new  $MBA(N, e, 2)$  architecture is better than already known structures giving greater OSXR, except “basic capacity” where it gives worst OSXR than *baseline* switching network of this same capac-

ity. What is more, and what reflects from architecture of  $MBA(N, e, 2)$  structure, an optical signal goes through one less stage than in *baseline* or Beneš network of this same capacity and functionality. That is why a new switching network is very attractive.

#### ACKNOWLEDGMENTS

The work described in this paper was financed from funds of Ministry of Science and Higher Education for year 2014 (grant no. 08/82/DSMK/8209).

#### REFERENCES

- [1] CLOS C., *A Study of Non-Blocking Switching Networks*, Bell System Technical Journal, Vol. 32, No. 2, 1953, 406–424.
- [2] GOKE L. R., LIPOVSKI G. J., *Banyan Networks for Partitioning Multiprocessor System*, In: Proceedings ISCA, 1973, 21–28.
- [3] PATTAVINA A., *Switching Theory: Architecture and Performance in Broadband ATM Networks*, John Wiley & Sons, 1998.
- [4] LEA C.-T., *Multi- $\log_2 N$  Networks and Their Applications in High-Speed Electronic and Photonic Switching Systems*, IEEE Transactions on Communications, Vol. 38, 1990, 1740–1749.
- [5] LEA C.-T., *Buffered or Unbuffered: A Case Study Based on  $\log_2(N, e, p)$  Networks*, IEEE Transactions on Communications, Vol. 44, No. 1, 1996, 105–113.
- [6] MELEN R., TURNER J. S., *Nonblocking Multirate Networks*, SIAM Journal on Computing, Vol. 18, No. 2, 1989, 301–313.
- [7] MELEN R., TURNER J. S., *Nonblocking Multirate Distribution Networks*, In: Proceedings IEEE INFOCOM, Vol. 3, 1990, 1234–1241.
- [8] CHEUNG S.-P., ROSS K. W., *On Nonblocking Multirate Interconnection Networks*, SIAM Journal on Computing, Vol. 20, No. 4, 1991, 726–736.
- [9] HWANG F. K., *The Mathematical Theory of Nonblocking Switching Networks*, 2<sup>nd</sup> edition, Singapore: World Scientific, 2004.
- [10] KABACINSKI W., *Nonblocking Electronic and Photonic Switching Fabrics*, Boston/London: Kluwer Academic Publisher, 2005.
- [11] DANILEWICZ G., KABACINSKI W., RAJEWSKI R., *The New Banyan-Based Switching Fabric Architecture Composed of Asymmetrical Optical Switching Elements*, In: Proceedings IEEE GLOBECOM, 2009, 1–6.
- [12] DANILEWICZ G., KABACINSKI W., RAJEWSKI R., *The  $\log_2 N - 1$  Optical Switching Fabrics*, IEEE Transactions on Communications, Vol. 59, No. 1, 2011, 213–225.
- [13] DANILEWICZ G., RAJEWSKI R., *The Architecture and Strict-Sense Nonblocking Conditions of a New Baseline-Based Optical Switching Network Composed of Symmetrical and Asymmetrical Switching Elements*, IEEE Transactions on Communications, Vol. 62, No. 3, 2014, 1058–1069.
- [14] CONNELLY M. J., *Semiconductor Optical Amplifiers*, Kluwer Academic Publisher, 2004.
- [15] EL-BAWAB T. S., *Optical Switching*, Springer, 2006.
- [16] LU C.-C., THOMPSON R. A., *The Double-Layer Network Architecture for Photonic Switching*, IEEE Transactions on Lightwave Technology, Vol. 12, No. 8, 1994, 1482–1489.
- [17] KABACINSKI W., *Modified Dilated Benes Networks for Photonic Switching*, IEEE Transactions on Communications, Vol. 47, No. 8, 1999, 1253–1259.
- [18] DANILEWICZ G., KABACINSKI W., ZAL M., *Reduced Banyan-Type Multiplane Rearrangeable Switching Networks*, IEEE Communications Letters, Vol. 12, No. 12, 2008, 900–902.

Marcin DZIUBA\*

## **NEW PARAMETERS OF BLOCKING WINDOW ALGORITHM**

In this paper, new parameters for the Blocking Window Algorithm (BWA) are considered. BWA with fixed-size windows was first introduced by Lee in [1]. In further research, Danilewicz and Kabaciński extended BWA to the variable window size variant in paper [2]. In both papers, the authors used BWA in multicast banyan type switching networks. Some researchers used Danilewicz and Kabaciński's approach in different architectures of nonblocking multicast switching networks [3], [4]. This paper presents the evaluation of BWA performance in blocking multicast  $\log_2(N, 0, p)$  switching networks. Different parameters of BWA are designated. It has been shown that for different parameters, BWA achieves different results. The best sets of these parameters were found.

### 1. INTRODUCTION

Multicast connections are becoming increasingly popular because sophisticated applications expect connections between one source and many destinations [5], [6], [7], [8]. Due to this fact, modern communication nodes must provide capabilities to handle also multicast connections. In means that a new generation of nodes has implemented switching networks that are able to establish a connection from one input to more than one or even all outputs. This kind of connections is well known in the literature as multicast or broadcast connections [9], [10]. One of the switching network architectures where multicast connections can be set up is multi- $\log_2 N$ . This architecture can also be called  $\log_2(N, 0, p)$  or multiplane banyan type switching network [1], [11], [12], [13], [14]. The nonblocking characteristics of the  $\log_2(N, 0, p)$  architecture was studied by authors (Lee, Danilewicz and Kabaciński, Tscha and Lee) in [1], [2],

---

\* Chair of Communication and Computer Networks, Faculty of Electronics and Telecommunications, Poznan University of Technology, Polanka 3, 60-965 Poznań, Poland.

[15]. All researchers tried to determine strict-sense nonblocking (SSNB) and wide-sense nonblocking (WSNB) conditions. Switching network is SSNB for a multicast connection where there is always the opportunity to find a connecting path between an open input and a collection of free outputs, regardless which of the searching path algorithms is used [2]. Switching network is WSNB for multicast connection, when there is always opportunities to find connecting path between open input and collection of free outputs. To find connection paths, suitable algorithm is used [2].

In the presented paper, the blocking window algorithm (BWA) is taken into account [1], [2]. This algorithm is based on a concept called variable-size blocking windows first described in [2]. New parameters of BWA are described in this paper. The presented algorithm has the same assumptions given by Danilewicz and Kabaciński in [2]. The results are presented because the variable-size blocking window algorithm was never considered in blocking multicast switching networks. In this paper, the best set of BWA parameters is found. For these parameters, BWA produces the lowest blocking probability.

The paper is organized as follows. In section 2, the research model description is given.  $\log_2(N, 0, p)$  switching network and multicast connection are described in more detail. Section 3 is dedicated to the blocking window algorithm and its parameters. The main concept of subconnections and planes selection in BWA is presented. In section 4, simulation parameters and graph results are discussed. In the last section, conclusions are drawn.

## 2. RESEARCH MODEL

### 2.1. THE $\log_2(N, 0, p)$ SWITCHING NETWORK ARCHITECTURE

In this paper, a multiplane multicast banyan type switching network is considered. One switching network plane used in our research is called  $\log_2 N$ . Such network consists of switches with 2 inputs and 3 outputs. In general, the number of inputs/outputs in one switch can be described with variable  $d$ . [2], [3]. All switches are arranged in  $n$  columns called stages. The number of switching network inputs and outputs can be calculated according to the equation:  $N = 2^n$ . Inter-stage links are used to connect all switches together. In our considerations, we have used a baseline inter-stage link pattern. Multiplane banyan type switching networks were first proposed and described by Lee in [12]. This kind of a network can be called  $\log_2(N, 0, p)$  or multi- $\log_2 N$  and it consists of  $p$  copies of  $\log_2 N$  switching network (also called plane) connected in parallel [12]. An example of an  $8 \times 8$  switching network is shown in Fig. 1.

In Fig. 1, it can be observed that the number of inputs and outputs is equal. The inputs/outputs are numbered  $0, 1, \dots, N-1$  from top to bottom. Stages are numbered

1, 2, ...,  $n$  from left to right, respectively. The main elements of a  $\log_2(N, 0, p)$  switching network were described in this section. More details about multi- $\log_2 N$  switching network architecture and its graphic representation can be found in papers [2], [3].

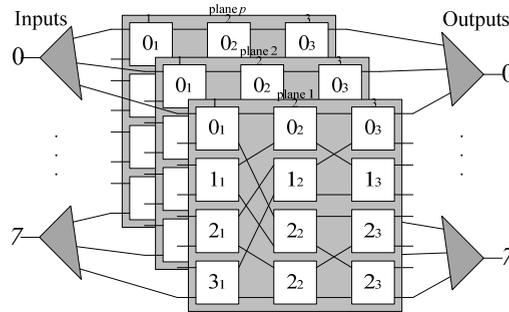


Fig. 1. An example of  $p$  planes of a  $\log_2(N, 0, p)$  switching network with  $N = 8$ , where  $n = 3$

## 2.2. MULTICAST CONNECTIONS

A multicast connection between input  $x$  and group of outputs  $y_0, y_1 \dots y_i, 0 \leq i \leq N - 1$  is denoted by  $\langle x, Y \rangle$ , where  $Y = \{y_0, y_1 \dots y_i\}$ . Thus, a connection is called multicast when  $|Y| > 1$ , where  $|Y|$  is a cardinality of set  $Y$ . To establish one multicast connection, the input and outputs should be selected from a suitable range  $0 < x, y_0, y_1 \dots y_i < N - 1$ . In the presented  $\log_2 N$  switching network, point-to-point ( $|Y| = 1$ ), broadcast ( $|Y| = N$ ) connections can be established. For example,  $\langle 4, 10 \rangle$  is a point-to-point connection, while  $\langle 4, \{1, 2, 5, 6\} \rangle$  is called a multicast connection.

Different connections can have their paths in the same plane. When two different connections (point-to-point or multicast) want to use the same path, the same input or the same output, one of these connections must be set up in a different plane of a switching network. To solve this problem, different algorithms are used. In one algorithm strategy, all paths belonging to the same multicast connection have to be set up in the same plane. In another approach, a multicast connection can be divided into separate point-to-point connections and each of them can be set up in a different plane of a switching network. Some details about algorithm strategy can be found in [1], [10], [15].

## 3. MULTICAST ALGORITHM

### 3.1. BLOCKING WINDOW DEFINITION

Using an algorithm to set up connections can influence the blocking probability in switching network blocking. In our consideration, we took into account multicast connection. Connections of this kind are set up by the blocking window algorithm. This

algorithm is based on the variable-size blocking window concept. More details about this algorithm can be found in [1], [2], [3], [15]. Blocking windows definition was presented in [2], [10]. In general, a blocking window is defined as follows:

**Definition 1:** Outputs  $O = \{0, 1, \dots, N - 1\}$  can be divided into  $N/K$  subsets  $BW_i = \{K \times i, K \times i + 1, \dots, K \times i + (K - 1)\}$ , where  $K = 2^t$ ,  $i = 0, 1, \dots, (N/K) - 1$  and  $1 \leq t \leq n$ . Each  $BW_i$  subset is called a blocking window.

According to Definition 1, a multicast connection can be divided into subconnections [2], [10]. For example, a new multicast connection can be described as  $\langle x, Y \rangle$ . The set of outputs  $Y$  can be divided into subsets  $Y_i$ , where  $Y_i = \{y: y \in BW_i, i = 0, 1, \dots, (N/K) - 1\}$ . In this approach, connection  $\langle x, Y_i \rangle$  is called the subconnection of multicast connection  $\langle x, Y \rangle$ .

### 3.2. BLOCKING WINDOW ALGORITHM

Algorithm, which based on blocking window concept, can be described as follows [1], [2], [10]:

**Algorithm 1:**

- Step 1: A new multicast connection  $\langle x, Y \rangle$  is divided into subconnections  $\langle x, Y_i \rangle$ ,
- Step 2: One of the subconnections is chosen,
- Step 3: One plane is chosen to set up this subconnection,
- Step 4: The next subconnection is chosen. Steps 3–4 are repeated until all subconnections of multicast connection  $\langle x, Y \rangle$  are set up.

The algorithm is responsible for choosing the subconnection that will be set up and the plane that will be used to set up this subconnection. The rules for choosing the subconnections and planes to set up the whole multicast connection are presented in the next subsection.

### 3.3. PARAMETERS OF THE BLOCKING WINDOW ALGORITHM

Three parameters of BWA are determined. These parameters specify the rules for choosing the subconnections and planes to set up a multicast connection in a suitable switching network. BWA parameters are also responsible for minimizing the blocking probability and this fact is presented in the next part of this paper. In the first step (according to Algorithm 1), a multicast connection is divided into subconnections using BWA. The number of the subconnections depends on the blocking window size. In the second step, one of these subconnections must be selected to be set up through

a switching network. The first parameter of BWA determines the rule of selecting the first subconnection. It was assumed that this selection will be performed by one of two algorithms:

- a) Sequential Algorithm (S) – selection of the first subconnection starts from the first blocking window where at least one output is required by the multicast connection,
- b) Random Algorithm (R) – selection of the first subconnection starts from a random blocking window where at least one output is required by the multicast connection

When the first subconnection is set up, the BWA chooses the next subconnections. The second parameter of BWA determines the rule of choosing these subconnections. It was assumed that this selection is made by one of two algorithms:

- a) Sequential Algorithm (S) – selection of the next subconnections starts according to an index of blocking windows where at least one output is required by the multicast connection,
- b) Random Algorithm (R) – selection of the next subconnections starts according to a random index of blocking windows where at least one output is required by the multicast connection.

The last parameter of BWA is responsible for choosing the plane where each subconnection of the multicast connection is set up. This selection is made by the following algorithms:

- a) Sequential Algorithm (S) – selection starts according to the plane index,
- b) Random Algorithm (R) – selection starts according to a random plane index,
- c) Quasi-Random Algorithm (RR) – the first selection is starts according to a random plane index and the next planes are selected according to a specified plane index, starting from the first selected plane,
- d) Beneš's Algorithm (B) – selection starts from the plane where the highest number of connections are already set up,
- e) Inverted Beneš's Algorithm (RB) – selection starts from the plane where the lowest number of connections are already set up.

According to the presented algorithms as BWA parameters, it was assumed that the blocking window algorithm will be marked as: Alg(sub\_1, sub\_succ, plane), where:

- Sub\_1 – first parameter of BWA, which is responsible for choosing the first subconnection of the multicast connection,
- Sub\_succ – second parameter of BWA, which is responsible for choosing the next subconnections of the multicast connection,
- Plane – third parameter which is responsible for choosing the plane where the subconnection is set up.

For example, Alg(S, R, RB) means that the first subconnection will be chosen by a sequential algorithm (S). The next subconnections will be set up by a random algo-

rithm (R). The plane where these subconnections will be set up will be selected according to the inverted Beneš's algorithm. Considering all the combinations of the blocking window algorithm parameters, twenty variants can be given: Alg(S, S, S), Alg(S, S, R), Alg(S, S, RR), Alg(S, S, B), Alg(S, S, RB), Alg(S, R, S), Alg(S, R, R), Alg(S, R, RR), Alg(S, R, B), Alg(S, R, RB), Alg(R, S, S), Alg(R, S, R), Alg(R, S, RR), Alg(R, S, B), Alg(R, S, RB), Alg(R, R, S), Alg(R, R, R), Alg(R, R, RR), Alg(R, R, B), Alg(R, R, RB).

## 4. SIMULATION RESULTS

### 4.1. SIMULATION PARAMETERS

In this paper, multi- $\log_2 N$  switching networks are considered. During the simulation process, the input and outputs required by one multicast connection are selected randomly. The number of outputs is a random variable with uniform distribution. Each multicast connection has an exponential holding time that can be calculated according to  $1/\mu$ . When the holding time of the multicast connection is finished, a suitable connection is disconnected in the switching network. All connections are incoming to a switching network according to the Poisson process. In the simulation, the work load was defined. This parameter can be calculated according to equation  $p = (\lambda/\mu)$ , where  $\lambda$  is the arrival rate [16]. The simulation process can be described as follows:

#### Simulation process 1:

Set blocking window size

counter\_1 = 0

blocked\_connections = 0

set\_connections = 0

**while** counter\_1  $\leq$   $5 * 10^5$  **do**

    Generate a new multicast connection (according to  $\lambda$ )

    Set holding time  $\mu$

    Multicast connection is divided into subconnections, according to the window size

**for** Alg(*sub-1*, *sub-succ*, *plane*) **do**

        choose the first subconnection, according to *sub-1*

**if** there is a possibility to set up the subconnection according to the algorithm *plane*

**then**

            set up the subconnection through a plane, according to the plane selected by the algorithm *plane*

**while** the next subconnection **do**

            choose the next subconnection according to algorithm *sub-succ*

**if** there is a possibility to set up the subconnection according to the algorithm *plane* **then**

                set up the subconnection through a plane, according to the plane selected by the algorithm *plane*

**else if then**

```

        blocked_connections = blocked_connections + 1
        break
    end if
end while
set_connections = set_connections + 1
else
    blocked_connections = blocked_connections + 1
    break
end if
end for
Connections where holding time is equal to 0 are disconnected
end while
Calculate and return simulation results (blocking probability)

```

The simulation procedure was the same for all sets of parameters of BWA. This procedure was performed for different switching network sizes ( $8 \times 8$ ,  $16 \times 16$ ,  $32 \times 32$ ,  $64 \times 64$ ,  $128 \times 128$ ). The simulation procedure and its parameters followed SSNB and WSNB conditions. These conditions (SSNB, WSNB) were presented by Danilewicz and Kabacinski in [2]. In this paper, the authors defined the minimum number of planes for a multi- $\log_2 N$  switching network to be strict-sense and wide-sense nonblocking. The simulation procedure confirmed that for the number of planes equal to WSNB conditions, the blocking probability for BWA is equal to 0. One simulation run has about  $5 * 10^5$  connection requests. The simulation program was implemented in the C programming language.

#### 4.2. BLOCKING PROBABILITY

WSNB conditions for a variable-size blocking windows algorithm were presented in [2]. These conditions determine the number of planes ( $p$ ) that are necessary to make a switching network nonblocking. WSNB conditions also define the optimum blocking window size, calculated according to  $2^t$  ( $1 \leq t \leq n$ ). In this paper, blocking switching networks are considered. In this kind of networks, fewer planes are used than determined in WSNB conditions. The point of simulations was to find the best set of parameters of Alg(sub\_1, sub\_succ, plane) that offers give the best results for a given number of planes. To find the best parameters, the blocking probability ( $BP$ ) was defined as follows:

$$BP = \frac{\text{blocked acceptable connection requests}}{\text{all acceptable connection requests}} \quad (1)$$

The numerator denotes blocked connection requests. This variable increases when at least one subconnection of a multicast connection is blocked in a switching network. The denominator denotes all acceptable connection requests. Better results are achieved by the algorithm which has a lower blocking probability.

## 4.3. GRAPH RESULTS

All graphs are prepared with 95% confidence intervals. The graphs are prepared as BP (y axis) versus traffic load (x axis). The results are presented for different numbers of planes ( $p$ ) and for optimal blocking window sizes (derived from WSNB conditions) for a given number of switching network stages ( $n$ ).

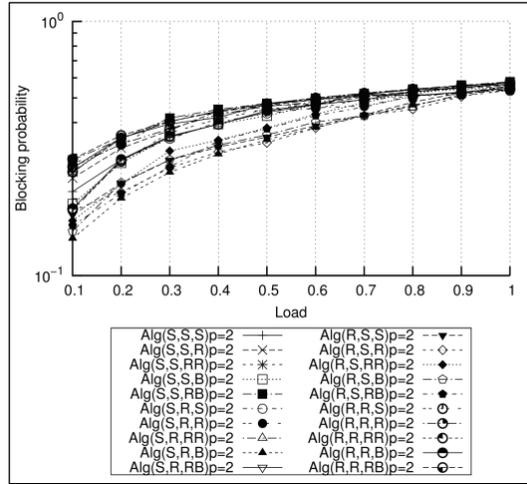


Fig. 2. The blocking probability for  $\log_2(16,0,2)$  switching network

Fig. 2 presents a  $\log_2(16,0,2)$  switching network where  $p = 2$  and  $t = 2$ . This graph presents the blocking probability for different BWA parameters. The confidence of interval marks is not included in this picture due to result line density. It can be observed that different parameters of BWA give different values of BP. The same trend can be seen for other values of  $N$  and  $p$ . In most cases, the best results are achieved by Alg(S, R, B). It means that the lowest blocking probability is achieved by the BWA where first the subconnection to be set up is selected sequentially (S), while the next subconnection is chosen randomly (R). Each subconnection of multicast connection is set up in the plane where the biggest number of connections is already set up (B - Beneš's Algorithm).

The other graphs (Fig. 3, Fig. 4, Fig 5) present the blocking probability for a different  $\log_2(16,0,2)$  switching network. Only the results for the set of parameters Alg(S, R, B) are presented. This set of parameters gives the best results among twenty sets of parameters (see Fig. 2). In each  $\log_2(N, 0, p)$  switching network, the number of planes changes from 1 to  $p_{WSNB} - 1$ , where  $p_{WSNB}$  is the number of planes defined in WSNB conditions for which a switching network is wide-sense nonblocking. In Figures 3-5, it can be seen that the blocking probability increases as the number of planes

decreases. The important conclusion is that the BP increases for the same number of planes when the number of inputs/outputs ( $N$ ) of the switching network also increases.

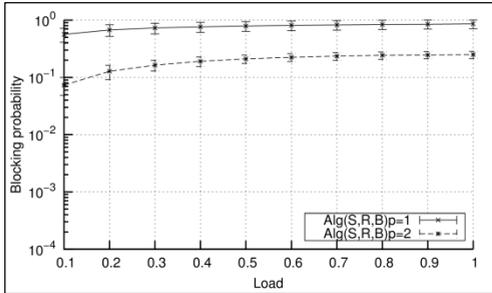


Fig. 3. The blocking probability for  $\log_2(8,0,p)$  switching network

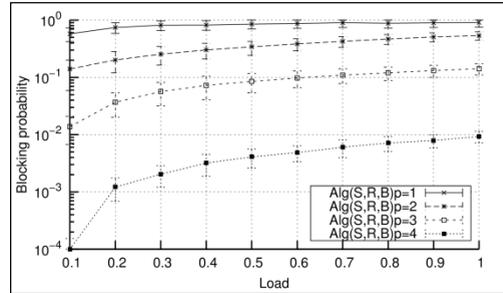


Fig. 4. The blocking probability for  $\log_2(16,0,p)$  switching network

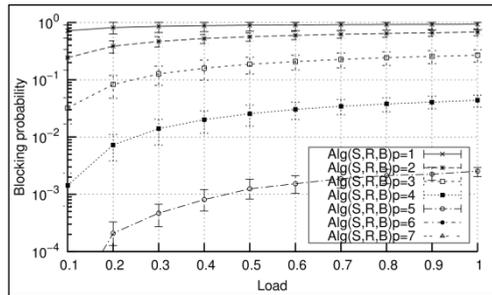


Fig. 5. The blocking probability for  $\log_2(32,0,p)$  switching network

## 5. CONCLUSION

In this paper, multiplane banyan-type switching network architecture is considered. This architecture has the capability to set up multicast connections. This kind of network can be called a  $\log_2 N$  or multi- $\log_2 N$  switching network. Banyan-type networks were considered in [2]. The authors presented strict-sense nonblocking and wide sense-nonblocking conditions for BWA based on the variable-size blocking window concept. In this paper, the parameters for BWA were defined. In the presented considerations, three parameters should be provided in BWA to set up multicast connections. The first parameter is connected with the selection of the first subconnection of the multicast connection. The second one is responsible for choosing the next subconnection of this multicast connection. The last parameter determines which algorithm will be used to find the plane for setting up subconnections through a switching network. The presented simulation results confirm that for different combinations of param-

ters, the blocking window algorithm achieves different blocking probabilities. In most cases, the set of parameters: Alg(S, R, B), BWA gives the lowest blocking probability. In this configuration, the first subconnection of the multicast connection is selected by a sequential algorithm (S). The next subconnections are chosen by a random algorithm (R). Each subconnection is set up through a plane where the biggest number of connections is already set up (B - Beneš's Algorithm). The results presented in this paper are the first step to further, greater work. The blocking window algorithm with the best set of parameters will be used to compare BWA with other algorithms well-known in the literature. In the future work, new parameters of BWA, which were investigated in this paper, will be used to set up multicast connections in multicast multi-plane banyan-type switching networks.

#### ACKNOWLEDGMENT

The work described in this paper was financed from funds of Ministry of Science and Higher Education for year 2014.

#### REFERENCES

- [1] TSCHA Y., LEE K. H., *Yet another result on multi- $\log_2 N$  networks*, IEEE Trans. Commun., Vol. 47, pp. 1425–1431, Sept. 1999.
- [2] KABACIŃSKI W., DANILEWICZ G., *Wide-sense and strict-sense non-blocking operation of multicast multi- $\log_2 N$  Switching Networks*, IEEE Trans. Commun., Vol. 50, pp. 1025–1036, June 2002.
- [3] DANILEWICZ G., *Wide-Sense Nonblocking Multicast  $\log_d(N, m, p)$  Switching Networks Under The Window Algorithm*, IEEE Trans. Commun., Vol. 57, pp. 3724–3731, December 2009.
- [4] HWANG F. K., *A Unifying Approach to Determine the Necessary and Sufficient Conditions for Nonblocking Multicast 3-Stage Clos Networks*, IEEE Trans. Commun., Vol. 53, pp. 1581–1586, Sept. 2005.
- [5] HWANG F. K., *The Mathematical Theory of Nonblocking Switching Networks*, Singapore, World Scientific, 2004.
- [6] KABACIŃSKI W., *Nonblocking Electronic and Photonic Switching Fabrics*, Kluwer Academic Publishers, 2005.
- [7] WITTMANN R., ZITTERBART M., *Multicast communication: protocols and applications*, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc., 1999.
- [8] LISTANI M., ROVERI A., *Switching structures for ATM*, Computer Commun., Vol. 12, pp. 349–358, Dec. 1989.
- [9] LEE T. T., *Nonblocking copy networks for multicast packet switching*, IEEE J. Select. Areas Commun., Vol. 6, pp. 1455–1467, Dec. 1988.
- [10] HUI J. Y., *Switching and Traffic Theory for Integrated Broadband Networks*, Boston, MA: Kluwer, 1990.
- [11] LEA C.-T., SHYY D.-J., *Tradeoff of horizontal decomposition versus vertical stacking in rearrangeable nonblocking networks*, IEEE Trans. Commun., Vol. 39, pp. 899–904, June 1991.
- [12] LEA C.-T., *Multi- $\log_2 N$  networks and their applications in high-speed electronic and photonic switching systems*, IEEE Trans. Commun., Vol. 38, pp. 1740–1749, Oct. 1990.

- [13] DANILEWICZ G., KABACINSKI W., *Log(N, m, p) broadcast switching networks*, in Proc. Int. Conf. Communications, Helsinki, Finland, June 2001, pp. 604–608.
- [14] —, *Wide-sense nonblocking multicast switching networks composed of  $\log_2 N + m$  stages*, in Proc. IEEE Int. Conf. Telecommunications, Bucharest, Romania, June 2001, pp. 519–524.
- [15] TSCHA Y., LEE K. H., *Non-blocking conditions for multi- $\log_2 N$  multiconnection networks*, in Proc. GLOBECOM, pp.1600–1604, 1992.
- [16] HWANG F. K., JAJSZCZYK A., *On nonblocking multiconnection networks*, IEEE Trans. Commun., Vol. COM-34, pp. 1038–1041, Oct. 1986.
- [17] YANG Y., MASSON M., *Non-blocking broadcast switching networks*, IEEE Trans. Computers, Vol. 40, pp. 1005–1015, Sept. 1991.
- [18] FUKUSHIMA Y., JIANG X., HORIGUCHI S., *Routing algorithm for packet/circuit switching optical multi- $\log_2 N$  networks*, IEEE Trans. Commun., Vol.E91-B, pp. 3913–3924, December 2008.



Wojciech M. KEMPA\*

## **ON TRANSIENT VIRTUAL DELAY IN A FINITE-BUFFER QUEUEING MODEL WITH SERVER BREAKDOWNS**

In the paper a finite-buffer queueing model with unreliable server subject to breakdowns is considered. Customers occur according to a Poisson process and are being served during exponentially distributed processing time. Successive exponential server's failure-free times are followed by generally distributed repair periods. Applying the paradigm of embedded Markov chain and the formula of total probability, a system of integral equations for the transient virtual queueing delay conditional distributions is built. Algebraic approach is used to obtain the solution of the corresponding system written for Laplace transforms in a compact form.

### 1. INTRODUCTION

Wide applications of finite-buffer queueing models are evident. They can be helpful in modelling of telecommunication and computer networks, automatized manufacturing lines, in transport and logistic problems. The appropriate queueing system can be used in the analysis of input/output process of packets in/from the buffer at the IP router's interface, it can also describe the accumulation of jobs at the "bottle neck" of manufacturing line. The most efficient use e.g. of telecommunication link or manufacturing line and hence cost and time saving, requires good knowledge of the behavior of the system under different conditions of its evolution. In this process of performance evaluation the observation and analysis of some important stochastic characteristics, like queueing delay, queue-size distribution, buffer overflow period duration, describing the operation of each queueing model, are very helpful. As it seems, particularly important is the knowledge of the behavior of these characteristics in the transi-

---

\* Silesian University of Technology, Institute of Mathematics, ul. Kaszubska 23, 44-100 Gliwice, Poland, e-mail: wojciech.kempa@polsl.pl

ent state of the system (i.e. at a fixed time epoch  $t$ ), not only in the stationary regime of the system's operation. The investigation of the system behavior in the transient state is essential if we are interested e.g. in the analysis of system's evolution just after the initial moment or after introducing a new service discipline (see e.g. [2]). Besides, sometimes important parameters describing the operation of the system (like e.g. the intensity of arrivals or service speed) are often and visibly changing in practice (as e.g. in the traffic of data packets in the Internet) so, in consequence, the system never stabilizes and the stationary-state idea has only theoretical significance.

In practice, especially, queueing systems with different-type limitation in service process are often used. In this class the model with cyclic failure-free and repair (conservation) times is one of the most important. First results related to queueing systems with unreliable server (or, with server subject to breakdowns) were published about fifty years ago. A single-server model with Poisson arrivals and generally distributed service times, and with unreliable server is considered e.g. in [1] and generalized for the multi-channel case in [14]. In [12] an infinite-buffer M/M/1-type queueing system with server subject to breakdowns is analyzed. Some models with server failures are also considered e.g. in [4], [5] and [13]. In [16] a multi-server queueing system with balking, reneging and server breakdowns is investigated. In [3] and [15] queueing models with unreliable server and negative customers (called also disasters), the arrivals of which cause dropping of some or all customers present in the system at the arrival epoch, are discussed.

In the paper we study a single-server queueing model with finite buffer and server breakdowns, in which successive interarrival, service and failure-free times of the operation are independent exponential random variables, while repair periods have general-type distribution. Using the approach based on the memoryless property of exponential distribution, the formula of total probability, integral equations and linear algebra, we obtain the explicit representation for the LT (= Laplace transform) of the virtual delay (virtual waiting time)  $v(t)$  at time  $t$  i.e. the waiting time of the customer arriving exactly at time  $t$ , conditioned by the number  $X(0)$  of customers present in the system at the opening.

Transient analysis of the virtual delay in general-type batch-arrival queueing model can be found e.g. in [6], while in [7] the distribution of the actual waiting time is discussed. In [8] the case of the virtual waiting time in a finite-buffer queue with single vacation policy is investigated, and in [9] new results for queueing delay in the system with queued waking up ( $N$ -policy) are derived.

So, the paper is organized as follows. In the next Section 2 we state a precise mathematical description of the considered queueing model and obtain the system of equations for the conditional virtual delay distributions. In Section 3 we obtain the solution of the corresponding system, written for LTs, in a compact form, using the algebraic method proposed in [10] (see also [11]). The last Section 4 contains short conclusion and remarks on the future work.

## 2. QUEUEING MODEL AND BASIC CONCEPTS

Let us take into consideration the M/M/1/N-type queueing model, in which the incoming jobs (packets, calls etc.) occur according to a Poisson process with intensity  $\lambda$  and are served individually with service speed  $\mu$ . A system contains one reliable server (machine) subject to breakdowns. Times of failure-free operation are supposed to have exponential distributions with parameter  $\gamma$ , while repair times are generally distributed with a DF (= distribution function)  $G(\cdot)$ . Besides, the maximal system capacity equals  $N$  i.e. we have a buffer (magazine) with  $N-1$  places and one place for service. Due to the finite buffer capacity, if the arriving job finds the system in state  $N$ , it is lost. The buffer may contain some jobs accumulated before the opening of the system. It is assumed that the server (machine) is in order at time  $t = 0$  and can break down only during the processing of a job. Introduce the following notation for the tail of the DF of virtual delay (virtual waiting time)  $v(t)$  conditioned by the number  $X(0) = n$  of jobs present in the system at the opening:

$$v_n(t, x) = P\{v(t) > x | X(0) = n\}, \quad 0 \leq n \leq N, \quad t > 0, \quad (1)$$

where  $X(t)$  stands for the number of jobs present in the system at time  $t$ .

Assume firstly that the system is empty at the opening. Denoting by  $y$  the first arrival epoch after  $t = 0$ , the following equation is true:

$$v_0(t, x) = \lambda \int_0^t e^{-\lambda y} v_1(t - y, x) dy. \quad (2)$$

In order to build the system of equations for the conditional virtual delay distribution in the system with at least one job present at  $t = 0$ , let us introduce the following random events, calling them the “key” ones, occurring during the evolution of the system: an arrival of a job; a service completion; a failure occurrence.

Let us define the following random events:

- $E_1$  – before time epoch  $t$  as the first “key” event an arrival of a job occurs;
- $E_2$  – before time  $t$  as the first “key” event a service completion (a departure) occurs;
- $E_3$  – before time  $t$  as the first from “key” events a failure of a machine (server) occurs;
- $E_4$  – all “key” events occur for the first time after time  $t$ .

Just from the formula of total probability we obtain the following equation:

$$v_n(t, x) = \sum_{i=1}^4 v_n^{E_i}(t, x) = \sum_{i=1}^4 P\{(v(t) > x) \cap E_i | X(0) = n\} \quad (3)$$

Assume now that the system is not empty and, simultaneously, not saturated at the initial moment i.e.  $X(0) = n$ , where  $1 \leq n \leq N-1$ . Since interarrival, service and failure-free times are exponentially distributed, then the moment of occurrence of any from “key” events is a Markov (renewal) moment during the operation of the system. As a consequence we get the following system of equations:

$$v_n^{E_1}(t, x) = \lambda \int_0^t e^{-(\lambda+\mu+\gamma)y} v_{n+1}(t-y, x) dy; \quad (4)$$

$$v_n^{E_2}(t, x) = \mu \int_0^t e^{-(\lambda+\mu+\gamma)y} v_{n-1}(t-y, x) dy; \quad (5)$$

$$\begin{aligned} v_n^{E_3}(t, x) = & \gamma \int_{y=0}^t e^{-(\lambda+\mu+\gamma)y} dy \left\{ \int_{u=0}^{t-y} \left[ \sum_{k=0}^{N-n-1} \frac{(\lambda u)^k}{k!} e^{-\lambda u} v_{n+k}(t-y-u, x) \right. \right. \\ & \left. \left. + \sum_{k=N-n}^{\infty} \frac{(\lambda u)^k}{k!} e^{-\lambda u} v_N(t-y-u, x) \right] dG(u) \right. \\ & \left. + \int_{u=t-y}^{\infty} \sum_{k=0}^{N-n-1} \frac{[\lambda(t-y)]^k}{k!} e^{-\lambda(t-y)} \cdot e^{-\mu(x-u-y+t)} \sum_{i=0}^{n+k-1} \frac{[\mu(x-u-y+t)]^i}{i!} dG(u) \right\}; \\ v_n^{E_4}(t, x) = & e^{-(\lambda+\mu+\gamma)t} v_n(0, x), \end{aligned} \quad (6)$$

$$v_n^{E_4}(t, x) = e^{-(\lambda+\mu+\gamma)t} v_n(0, x), \quad (7)$$

where

$$v_n(0, x) = \sum_{k=0}^{n-1} \frac{(\mu x)^k}{k!} e^{-\mu x} \quad (8)$$

i.e.  $v_n(0, x)$  is the tail of  $(n-1)$ -Erlang DF with parameter  $\mu$ .

Let us explain (4)–(7) in few words. In (4) the first arrival occurs at time  $y < t$  as the first one from the “key” events, so the service completion epoch and a failure occur after  $y$  (with probabilities  $e^{-\lambda y}$  and  $e^{-\gamma y}$  respectively). Hence, beginning with a renewal (Markov) moment  $y$ , the operation of the system coincides with the evolution of the system with  $n+1$  jobs present at the opening. Similarly we can comment (5). The representation (6) describes the situation in which the failure occurring before  $t$  is the first “key” event after the opening of the system. The first summand on the right side of (6) relates to the case in which a repair is also finished before time  $t$ , and during the service suspension the buffer does not become saturated. In the second summand on the right side of (6) the system continues the operation after the repair with the maximal

number  $N$  of jobs present. In the last summand the repair ends after  $t$ . The interpretation of (7) is obvious: since none of “key” events occurs before  $t$ , then the probability that the queueing delay at time  $t$  exceeds  $x$  is equal to the analogous probability measured at time 0.

Finally, for the system being saturated at the opening ( $n = N$ ), we obtain

$$v_N^{E_1}(t, x) = \lambda \int_0^t e^{-(\lambda+\mu+\gamma)y} v_N(t-y, x) dy; \quad (9)$$

$$v_N^{E_2}(t, x) = \mu \int_0^t e^{-(\lambda+\mu+\gamma)y} v_{N-1}(t-y, x) dy; \quad (10)$$

$$v_N^{E_3}(t, x) = \gamma \int_{y=0}^t e^{-(\lambda+\mu+\gamma)y} dy \int_{u=0}^{t-y} v_N(t-y-u, x) dG(u); \quad (11)$$

$$v_N^{E_4}(t, x) = e^{-(\lambda+\mu+\gamma)t} v_N(0, x), \quad (12)$$

### 3. LAPLACE TRANSFORM OF QUEUEING DELAY DISTRIBUTION

Introducing the LT of  $v_n(t, x)$  on the argument  $t$  as follows:

$$\hat{v}_n(s, x) = \int_0^\infty e^{-st} v_n(t, x) dt, \quad \text{Re}(s) > 0, \quad 0 \leq n \leq N, \quad (13)$$

and taking into consideration the following representations:

$$\begin{aligned} & \gamma \int_{t=0}^\infty e^{-st} dt \int_{y=0}^t e^{-(\lambda+\mu+\gamma)y} dy \int_{u=0}^{t-y} \frac{(\lambda u)^k}{k!} e^{-\lambda u} v_j(t-y-u, x) dG(u) \\ &= \frac{\gamma}{\lambda+\mu+\gamma+s} \hat{v}_j(s, x) \alpha_k(s); \end{aligned} \quad (14)$$

$$\begin{aligned} & \gamma \int_{t=0}^\infty e^{-st} dt \int_{y=0}^t e^{-(\lambda+\mu+\gamma)y} dy \int_{u=t-y}^\infty \sum_{k=0}^{N-n-1} \frac{[\lambda(t-y)]^k}{k!} e^{-\lambda(t-y)} e^{-\mu(x-u-y+t)} \\ & \quad \times \sum_{i=0}^{n+k-1} \frac{[\mu(x-u-y+t)]^i}{i!} dG(u) = \frac{\gamma}{\lambda+\mu+\gamma+s} \beta(s, x); \end{aligned} \quad (15)$$

$$\begin{aligned} & \gamma \int_{t=0}^{\infty} e^{-st} dt \int_{y=0}^t e^{-(\lambda+\mu+\gamma)y} dy \int_{u=0}^{t-y} v_N(t-y-u, x) dG(u) \\ &= \frac{\gamma}{\lambda + \mu + \gamma + s} g(s) \hat{v}_j(s, x) \alpha_k(s), \end{aligned} \quad (16)$$

where, for  $Re(s) > 0$ ,

$$g(s) = \int_0^{\infty} e^{-st} dG(s); \quad (17)$$

$$\alpha_k(s) = \int_0^{\infty} \frac{(\lambda t)^k}{k!} e^{-(\lambda+s)t} dG(t); \quad (18)$$

$$\begin{aligned} & \beta(s, x) \\ &= e^{-\mu x} \int_{u=0}^{\infty} e^{\mu u} dG(u) \sum_{k=0}^{N-n-1} \int_{z=0}^u \frac{(\lambda z)^k}{k!} e^{-(\lambda+s+\mu)z} \sum_{i=0}^{n+k-1} \frac{[\mu(x-u+z)]^i}{i!} dz, \end{aligned} \quad (19)$$

we obtain from (2)-(12) the following system of equations:

$$\hat{v}_0(s, x) = \frac{\lambda}{\lambda + s} \hat{v}_1(s, x); \quad (20)$$

$$\begin{aligned} & \hat{v}_n(s, x) = \frac{1}{\lambda + \mu + \gamma + s} \left[ \lambda \hat{v}_{n+1}(s, x) + \mu \hat{v}_{n-1}(s, x) \right. \\ & + \gamma \left( \sum_{k=0}^{N-n-1} \hat{v}_{n+k}(s, x) \alpha_k(s) + \hat{v}_N(s, x) \sum_{k=N-n}^{\infty} \alpha_k(s) + \beta(s, x) \right) \\ & \left. + v_n(0, x) \right], \quad 1 \leq n \leq N-1; \end{aligned} \quad (21)$$

$$\hat{v}_N(s, x) [\mu + s + \gamma(1 - g(s))] = \mu \hat{v}_{N-1}(s, x) + v_N(0, x). \quad (22)$$

Introducing now the following functions:

$$a_0(s) = \frac{\mu}{\lambda + \mu + \gamma + s}; \quad a_{k+1}(s) = \frac{1}{\lambda + \mu + \gamma + s} [\lambda \delta_{k1} + \gamma \alpha_k(s)]; \quad (23)$$

$$\begin{aligned} \psi_k(s, x) &= -\frac{1}{\lambda + \mu + \gamma + s} \left[ \gamma \left( \hat{\omega}_0(s, x) \sum_{i=k}^{\infty} \alpha_i(s) + \beta(s, x) \right) + v_{N-k}(0, x) \right] \\ &= A_k(s) \hat{\omega}_0(s, x) + B_k(s, x), \end{aligned} \quad (24)$$

where

$$A_k(s) = -\frac{\gamma}{\lambda + \mu + \gamma + s}; \quad B_k(s, x) = -\frac{1}{\lambda + \mu + \gamma + s} [\gamma \beta(s, x) + v_{N-k}(0, x)]; \quad \hat{\omega}_n(s, x) = \hat{v}_{N-n}(s, x), \quad (25)$$

we can transform the system (20)–(22) to the form

$$\hat{\omega}_0(s, x) [\mu + s + \gamma(1 - g(s))] = \mu \hat{\omega}_1(s, x) + v_N(0, x) \quad (26)$$

$$\sum_{k=-1}^{n-1} a_{k+1}(s) \hat{\omega}_{n-k}(s, x) - \hat{\omega}_n(s, x) = \psi_n(s, x), \quad 1 \leq n \leq N-1 \quad (27)$$

$$\hat{\omega}_N(s, x) = \frac{\lambda}{\lambda + s} \hat{\omega}_{N-1}(s, x). \quad (28)$$

The following algebraic result is proved in [11] (see also [12]):

**Lemma 1.** *Let  $(a_k)$ ,  $k \geq 0$ ,  $a_0 \neq 0$ , and  $(\psi_k)$ ,  $k \geq 1$ , be two known sequences. Every solution of the infinite-sized system of linear equations*

$$\sum_{k=-1}^{n-1} a_{k+1} x_{n-k} - x_n = \psi_n, \quad n \geq 1, \quad (29)$$

can be written as

$$x_n = C R_n + \sum_{k=1}^n R_{n-k} \psi_k, \quad n \geq 1, \quad (30)$$

where  $C$  is independent on  $n$  and the sequence  $(R_k)$ ,  $k \geq 0$ , is defined recursively by the formulae

$$R_0 = 0, \quad R_1 = a_0^{-1}, \quad R_{k+1} = R_1 \left( R_k - \sum_{i=0}^k a_{i+1} R_{k-i} \right), \quad k \geq 1. \quad (31)$$

Observe that the system (27) has the same form as (29) but with sequences  $(a_k)$  and  $(\psi_k)$  depending on  $s$  and  $s, x$ , respectively. Hence, for  $n \geq 1$ , the representation for  $\hat{\omega}_n(s, x)$  can be obtained just from (30) in the form

$$\hat{\omega}_n(s, x) = C(s, x)R_n(s) + \sum_{k=1}^n R_{n-k}(s)\psi_k(s, x), \quad (32)$$

where  $a_k(s)$  and  $\psi_k(s, x)$  were defined in (23) and (24), respectively. To present the main result in a compact form we need the formulae for  $C(s, x)$  and  $\hat{\omega}_0(s, x)$ .

Substituting  $n = 1$  into (32) we obtain

$$\hat{\omega}_1(s, x) = C(s, x)R_1(s) = C(s, m)a_0^{-1}(s). \quad (33)$$

After combining (26) and (33) we get the following representation for  $C(s, x)$ :

$$C(s, x) = a_0(s)\mu^{-1} \left[ \hat{\omega}_0(s, x) (\mu + s + \gamma(1 - g(s))) - v_N(0, x) \right]. \quad (34)$$

Next, substituting  $n = N$  in (32), we obtain

$$\hat{\omega}_N(s, x) = D_1(s, x)\hat{\omega}_0(s, x) + N_2(s, x), \quad (35)$$

where

$$D_1(s, x) = a_0(s)\mu^{-1}R_N(s) (\mu + s + \gamma(1 - g(s))) + \sum_{k=1}^N R_{N-k}(s)A_k(s); \quad (36)$$

$$N_2(s, x) = \sum_{k=1}^N R_{N-k}(s)B_k(s, x) - a_0(s)\mu^{-1}R_N(s)v_N(0, x). \quad (37)$$

Similarly, introducing (32) into the right side of (28), we get

$$\hat{\omega}_N(s, x) = D_2(s, x)\hat{\omega}_0(s, x) + N_1(s, x), \quad (38)$$

where now

$$D_2(s, x) = \frac{\lambda}{\lambda + s} a_0(s)\mu^{-1}R_{N-1}(s) (\mu + s + \gamma(1 - g(s))); \quad (39)$$

$$N_1(s, x) = \frac{\lambda}{\lambda + s} \left( \sum_{k=1}^{N-1} R_{N-k}(s)\psi_k(s, x) - a_0(s)\mu^{-1}R_{N-1}(s)v_N(0, x) \right). \quad (40)$$

Comparing now the right sides of (35) and (38) we find  $\hat{\omega}_0(s, x)$  as

$$\widehat{\omega}_0(s, x) = \frac{N_1(s, x) - N_2(s, x)}{D_1(s, x) - D_2(s, x)}. \quad (41)$$

Finally, the representations (25), (32), (34) and (41) lead to the following main theorem:

**Theorem 1.** *The Laplace transform  $\widehat{v}_n(s, x)$  of the tail of DF of virtual delay in the considered queueing system with server subject to breakdowns can be found from the following formula:*

$$\widehat{v}_n(s, x) = a_0(s)\mu^{-1} \left[ \widehat{\omega}_0(s, x) (\mu + s + \gamma(1 - g(s))) - v_N(0, x) \right] R_{N-n}(s) + \sum_{k=1}^{N-n} R_{N-n-k}(s) (A_k(s)\widehat{\omega}_0(s, x) + B_k(s, x)), \quad (42)$$

where  $Re(s) > 0, 0 \leq n \leq N$ , and the formulae for  $v_N(0, x)$ ,  $g(s)$ ,  $a_0(s)$ ,  $A_k(s)$ ,  $B_k(s, x)$ ,  $R_n(s)$  and  $\widehat{\omega}_0(s, x)$  were found in (8), (17), (18), (25), (31) and (41), respectively.

#### 4. CONCLUSION AND FUTURE WORK

In the paper an analytical approach to the study of transient queueing delay in a finite-buffer model with a single server subject to breakdowns is presented. The method is based on the idea of embedded Markov chain, the formula of total probability and linear algebra. The case of exponential interarrival, service and failure-free times and generally distributed service times is considered. In future numerical analysis of the impact of distribution parameters on the queueing delay behavior is planned.

The case of more general probability distributions describing the evolution of the system will be investigated as well.

#### ACKNOWLEDGEMENT

The project was financed with subsidies of from the National Science Centre in Poland, granted by virtue of the decision no. DEC-2012/07/B/ST6/01201.

#### REFERENCES

- [1] AVI-ITZHAK B., NAOR P., *Some queueing problems with the server station subject to breakdown*, Operations Research, Vol. 11, 1963, 303–320.
- [2] BERTSIMAS D.J., NAKAZATO D., *Transient and busy period analysis of the GI/G/1 queue: The method of stages*, Queueing Systems, Vol. 10, No. 3, 1992, 153–184.

- [3] BOXMA O.J., PERRY D., STADJE W., *Clearing models for M/G/1 queues*, *Queueing Systems*, Vol. 38, 2001, 287–306.
- [4] GRAY W.J., WANG P.P., SCOTT M.K., *A vacation queueing model with server breakdowns*, *Mathematical Modelling*, Vol. 24, 2000, 391–400.
- [5] KE J.C., *An M/G/1 queue under hysteretic vacation policy with an early startup and unreliable server*, *Mathematical Methods of Operations Research*, Vol. 63, 2006, 357–369.
- [6] KEMPA W.M., *The virtual waiting time for the batch arrival queueing systems*, *Stochastic Analysis and Applications*, Vol. 22, No. 3, 2004, 1235–1255.
- [7] KEMPA W.M., *Some results for the actual waiting time in batch arrival queueing systems*, *Stochastic Models*, Vol. 26, No. 3, 2010, 335–356.
- [8] KEMPA W.M., *The virtual waiting time in a finite-buffer queue with a single vacation policy*, In: *Analytical and stochastic modeling techniques and applications*, Al-Begain K., Fiems D., Vincent J.-M. (Eds.), *Proceedings of the 19<sup>th</sup> International Conference AMSTA 2012*, Grenoble, France, *Lecture Notes in Computer Science*, Berlin, Springer, 2012, Vol. 7314, 47–60.
- [9] KEMPA W.M., *On queueing delay in WSN with energy saving mechanism based on queued wake up*, *Proceedings of the 21<sup>st</sup> International Conference on Systems, Signals and Image Processing IWSSIP 2014*, Mustra M. et al. (Eds.), Dubrovnik, Croatia, Faculty of Electrical Engineering and Computing, University of Zagreb, 2014, 187–190.
- [10] KOROLYUK V.S., *Boundary-value problems for compound Poisson processes*, Kiev, Naukova Dumka, 1975 (in Russian).
- [11] KOROLYUK V.S., BRATIICHUK M.S., PIRDZHANOV B., *Boundary-value problems for random walks*, Ashkhabad, Ylym, 1987 (in Russian).
- [12] LAM Y., ZHANG Y.L., LIU Q., *A geometric process model for M/M/1 queueing system with a repairable service station*, *European Journal of Operational Research*, Vol. 168, 2006, 100–121.
- [13] MADAN K.C., *A M/G/1 type queue with time-homogeneous breakdowns and deterministic repair times*, *Soochow Journal of Mathematics*, Vol. 29, No. 1, 2003, 103–110.
- [14] NEUTS M.F., LUCANTONI D.M., *A Markovian queue with n servers subject to breakdowns and repairs*, *Management Science*, Vol. 25, 1979, 849–861.
- [15] SHIN Y.W., *Multi-server retrial queue with negative customers and disasters*, *Queueing Systems*, Vol. 55, 2007, 223–237.
- [16] WANG K.-H., CHANG Y.-C., *Cost analysis of a finite M/M/R queueing system with balking, reneging and server breakdowns*, *Mathematical Methods of Operations Research*, Vol. 56, 2002, 169–180.

Jan KWIATKOWSKI\*, Andrzej GNATOWSKI\*

## **PARALLELIZATION OF WIRELESS ROUTERS PLACEMENT ALGORITHMS FOR THE GPU**

To ensure user comfort and lower costs of construction and maintenance of the wireless network, it is necessary the appropriate planning of network range and placement of access points. For this aim algorithms for simulating electromagnetic wave propagation and WLAN optimization are used. These algorithms are demanding in terms of the computing power in order for receiving the results in the reasonable time. As an alternative the parallel processing can be used. The aim of the presented research is verification if using GPU is possible to accelerate required computation in reasonable way. The way of parallelization of used for network planning algorithms are presented. During experiments the COST231 algorithm is used for simulation of routers range and as optimization algorithms: pruning, simulated annealing and neighborhood search have been studied. The first obtained results are very promising, for some cases the received speedup is close to 50.

### 1. INTRODUCTION

Nowadays wireless services are a part of the everyday life of billions of people around the world. There are a number of technologies that can be used to supply this service, one of them is a local wireless network (WLAN). These networks can be found commonly in many public buildings, stores, offices, companies headquarters, as well as in private houses. To ensure user comfort and lower costs of construction and maintenance of such networks, it is necessary the appropriate planning of network range and placement of access points (AP). On the other hand there are many parameters, which can be used for the overall quality assessment of the wireless network. During network planning, it is necessary to predict their values before the network will be created. It is mainly done by simulation of routers behavior. Moreover, the ability

---

\* Wrocław University of Technology, Faculty of Computer Science and Management, Institute of Informatics, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27.

to predict network configuration parameters is not sufficient to create a network with the satisfactory quality. The number of possible routers locations is too large to take into consideration all of them. It causes the need of using optimization algorithms, which optimized the routers locations. During this step the information about different network components, their expected behavior and defined requirements are used for propose the best suitable solution. Such approach is currently commonly used in practice and is supported by a number of available software tools [1, 6]. In both above mentioned steps of network planning the large amount of information is required and used algorithms are computationally intensive. The problem is particularly acute for the large buildings and a large number of possible routers locations. It causes the need for using the powerful computers for receiving the results in reasonable time. As alternative of this, the parallel processing can be used. The aim of the research is to determine if using GPU is possible to accelerate required computations in reasonable way.

The chapter describes the way of parallelization of different algorithms used during network planning, and is organized as follows. Section 2 introduces design overview. In the section 3 it is shown how used algorithms are parallelized. Section 4 presents the results of performance evaluation of the developed parallel algorithms. Finally, section 5 summarized presented work.

## 2. DESIGN OVERVIEW

The simulation the propagation of electromagnetic waves used by WLAN require the creation of propagation models for interior buildings. This environment is characterized by smaller distances between the transmitting and receiving antennas, than in the case of open spaces. This is due to the presence of large amounts of the obstacles from the transmitter to the receiver, the signal strength is often very weak. What's more transmitters used in WLAN benefit from unlicensed bands, where the signal strength is limited by the law. All of these factors, in combination with a relatively large wave suppression by construction materials, causes the need for the creation of specific models of propagation.

To simulate the routers ranges the COST231 (multi-wall model) algorithm [9] has been chosen for implementation. Single-storey building approach is considered (eq. 1).

$$L = L_{FS} + L_C + \sum_{i=0}^I k_{wi} * L_{wi} \quad (1)$$

where  $L$  – suppression between antennas [dB],  $L_{FS}$  – suppression of free space [dB],  $L_C$  – fixed suppression,  $I$  – the number of walls category,  $k_{wi}$  – the number of  $i$ -th walls

category between the transmitter and the receiver,  $L_{wi}$  - suppression of  $i$ -th wall category [dB]. The used model simplification does not constitute a significant simplification of the solved problem. For many floors when calculating the router range at the some point also the routers from other floors should be considered. In this case ceiling behaves as another wall suppressed the signal.

In general, optimization problem can be defined as minimization or maximization of the objective function  $f: X \rightarrow R$ , where  $X$  is a finite set of allowed states and represents the routers configuration. Used in the presented research the objective function is weighted average of the number of factors. This approach is similar to that described in [7]. The objective function can be given by the formula:

$$f(x, p_0, p_1, \dots) = \frac{\sum_{i=0}^{N_F} a_i f_i(x_i, p_i)}{\sum_{i=0}^N a_i} \quad (2)$$

where:  $f$  – objective function,  $x$  - allowed states,  $N_F$  - the number of components of the objective function,  $f_i$  –  $i$ -th component of the objective function,  $a_i$  – weight of  $i$ -th component,  $p_i$  – parameter of  $i$ -th component.

In the formula 2, the values of  $p_i$  parameters of the objective function should be determined before the start of the optimization task. Examples of parameters that can be used by objective function are as follows: theoretical average speed of data transmission, average delays, the required signal strength in different places of the building [7], [2], network load – if more users, the less the maximum speed of data transfer [2], interference – decrease the rate of data transfer, can enhance the delay [2, 5], and others.

In the presented research only signal strength and network load have been taken into consideration. In algorithms for routers placement the number of routers is a parameter. Therefore, by running simulations for different values of this parameter, knowing desirable parameters of network, the minimum number of routers that meet the specified requirements can be determine.

Both of above mentioned types of algorithms have been parallelized. As optimization algorithms pruning, simulated annealing (SA) and neighborhood search (NS) [4, 7] have been chosen for the experiments. For optimization algorithms only their part responsible for calculation the value of the objective function has been parallelized.

### 3. PARALLELIZATION OF USED ALGORITHMS

The pseudo code of the parallel implementation of COST231 algorithm for GPU is presented below. For each router  $r$  from the set of routers  $R$  its signal strength is calculated for each point  $p$ , from the set  $P$ , that is defined basing on the building map. The

signal strength of the router  $r$  in the point  $p$  with coordinates  $(p_x, p_y)$  is marked by  $o_{p_x, p_y}^r$  and is initiated by the suppression of the free space ( $L_{FS}$ ). Then the suppression of all walls ( $W_{suppression}$ ) which are located between the router and the measuring point is added ( $L_{wi}$ ). The  $intersects(w, r, p)$  procedure is used to determine the intersection. In the next step the suppression arising from propagation waves of a particular frequency ( $F$ ) and from the distance between router and analysed point is added ( $L_{FS}$ ). Finally, the value of signal strength in the point  $p$  is calculated by reducing the router signal strength ( $R_{power}$ ) by the value of  $o_{p_x, p_y}^r$ . As a result the signal strength from each router in each measurement point is obtained.

```

1: procedure COST231 ( $R, P, W, F$ )
2:   for all  $o \in O$  do
3:      $o \leftarrow 0$ 
4:   end for
5:   for all  $r \in R$  do
6:     for all  $p \in P$  do
7:       for all  $w \in W$  do
8:         if  $intersects(w, r, p)$  then
9:            $O_{p_x, p_y}^r \leftarrow O_{p_x, p_y}^r + W_{suppression}$ 
10:        end if
11:      end for
12:       $d \leftarrow distance\_Router\_Point$ 
13:       $O_{p_x, p_y}^r \leftarrow O_{p_x, p_y}^r + (-27.55 + 20 \log_{10}(F) + 20 \log_{10}(d))$ 
14:      if  $O_{p_x, p_y}^r < 0$  then
15:         $O_{p_x, p_y}^r \leftarrow 0$ 
16:      end if
17:       $O_{p_x, p_y}^r \leftarrow R_{power} - O_{p_x, p_y}^r$ 
18:    end for
19:  end for
20:  return  $O$ 
21: end procedure

```

In the code above it can be find two nested loops (line 5 and 6). The first loop is parallelized by mapping it into the grid of blocks at the GPU. The second loop is splitting into two loops using the coordinates of measuring points  $(p_x, p_y)$ . To do this the size of the block has been declared as  $(32*32)$ . This resulted in that a single tread performs operation from lines from 7 to 17.

The pseudo code of the parallel implementation of calculation the value of the objective function for GPU is presented below. For each measuring point  $p$  from the set of all measuring points  $P$ , is searched the router  $r$  from the set of active routers  $R$  which has the greatest signal strength value  $o_{p_x, p_y}^r$  in the measuring point. If this signal is greater than the desired value of the minimum signal  $d_p$ , it's value is reduced to the value of  $d_p$ . If at a point  $p$  is the user  $C_{p_x, p_y}$ , it causes increasing the counter  $u_r$  that indicates the number of users registered to the router  $r$ . Then the signal strength ( $M_{\text{sum}}$ ) and network load are calculated using the equations from [3]. Finally knowing the weights of  $a_0$  and  $a_1$  the value of the objective function is calculated.

```

1: procedure F(O, a0, a1, C, P, dp, R)
2:   m ∈ M
3:   u ∈ U
4:   for all p ∈ P do
5:     mp ← -∞
6:     for all r ∈ R do
7:       if mp < Opxpyr then
8:         mp ← Opxpyr
9:         rbest ← r
10:      end if
11:    end for
12:    if mp > dp then
13:      mp ← dp
14:    end if
15:    if cpxpy ∈ C then
16:      ur ← ur + 1
17:    end if
18:  end for
19:  Msum ← add(M)
20:  return calculate_value_of_objective_function(M, U, a0, a1)
21:end procedure

```

For parallelization the most inner loop has been chosen (lines from 4 to 18). Each iteration of this loop is executed as a separate thread at the GPU. All operations can be executed independently with the exception of operation from the line 16 that may requires modifying the same memory area by multiple threads at the same time. To avoid it, the function `atomicAdd()` is used. An important feature of the above parallelization method is that the results of the algorithm that simulates the propagation of electromagnetic waves are used as input for the optimization algorithm. thanks to this

there is no need to transfer large amounts of data to GPU, making it possible to achieve larger speedup.

#### 4. COMPUTATIONAL EXPERIMENTS

To confirm the usefulness of developed algorithms the series of tests have been performed. The tests have been carried out on the PC computer with Intel Core 2 Duo E8500 processor with 4 x 2 GB DDR2-800 DDR2 SDRAM memory and Gigabyte GeForce GTX 660 graphic card running under Windows 8.1 Pro operating system. To reduce the influence of the Windows multiprocessing environment, running tests have had a high priority and the small number of the background processes have been used. As programming language CUDA C / C + +, version 6 has been used [8]. For GPU memory allocation `cudaMallocManaged ()` function has been used.

During experiments as the program execution time the wall clock time has been measured due to the measured GPU memory allocation time were significantly less than computation time (in average 63 times smaller). Due to the implementation environment square maps with the side length equal to multiple of 32 have been used, the walls have been spaced in the random way, when routers have been evenly distributed on the map. Suppression of walls was set at 10 [dB], and the routers signal strength at 20 [dBm]. During the tests the impact of three parameters on the execution time has been checked: the size of the map, the number of possible routers locations and the number of walls.

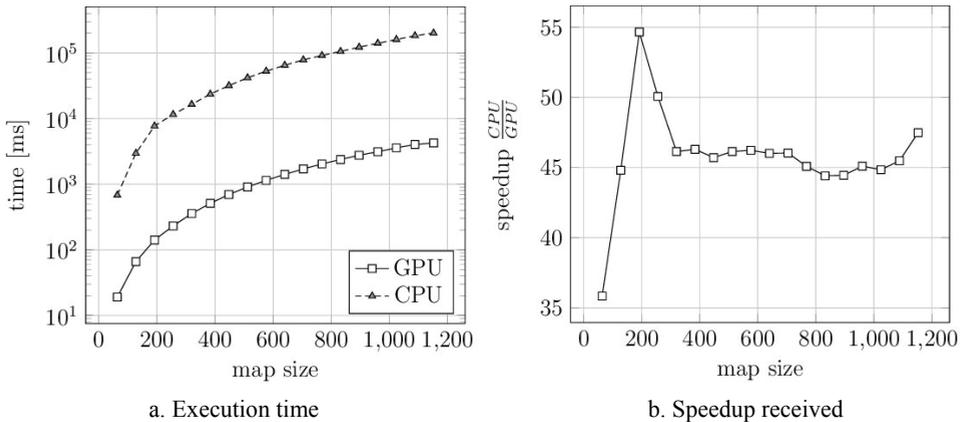


Fig. 1. Execution time and speedup as a function of the maps size (walls – 30, router locations – 81)

In the first test the influence the size of the map on the execution time for simulation the range of the routers algorithm has been checked. It may be noted in figure 1,

that for small-sized maps, the speedup is small and with increases the size of the problem it reaches about 46. Such a significant speedup is achieved due to the high degree of parallelization. Moreover, the algorithm does not require a lot of communication between the computer's RAM and GPU due to a lot of needed input data are created and used directly on the GPU. The same experiments have been done for larger maps sizes with less walls and possible routers location (figure 2) and obtained results were similar.

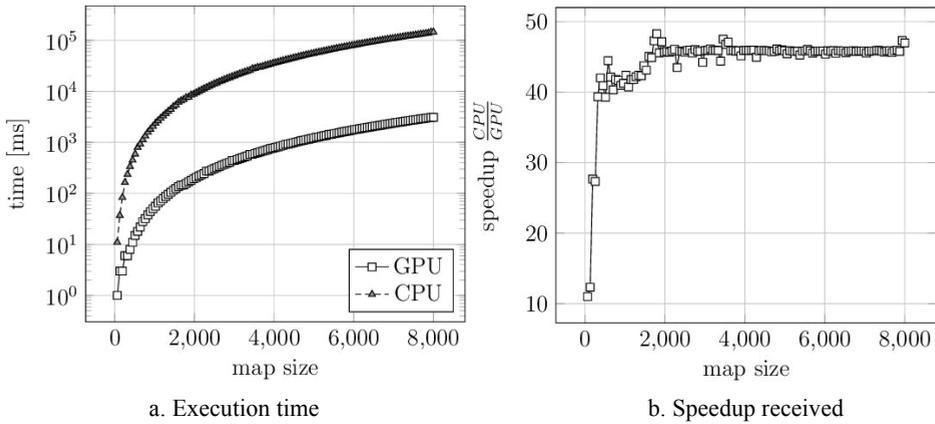


Fig. 2. Execution time and speedup as a function of the maps size (walls – 5, router locations – 4)

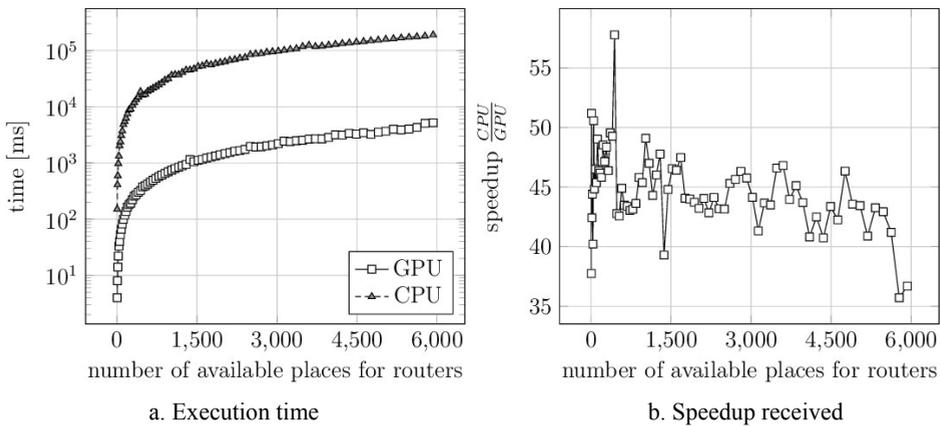


Fig. 3. Execution time and speedup as a function of the number of routers locations (maps size 128\*128, walls – 20)

In the second test the influence of the number of routers location on the execution time for simulation the range of the routers algorithm has been checked. The number of routers locations changed from 4 to 5929. The results of experiments are presented in figure 3. The average speedup is less than 45. Unlike the previous test, speedup

with increasing amounts of routers locations began to fall, however does not fall below 35.

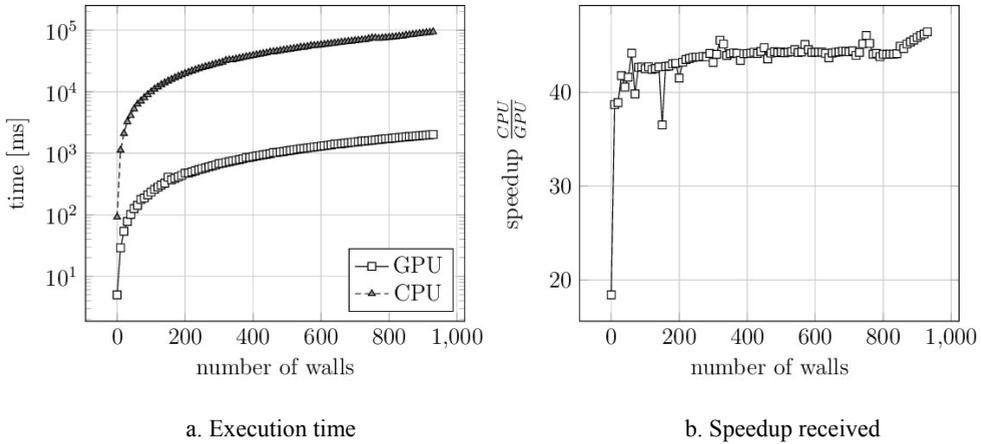
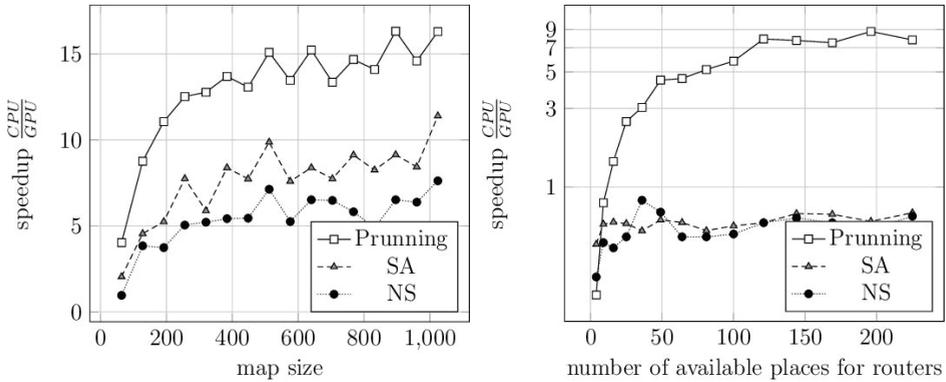


Fig. 4. Execution time and speedup as a function of the number of walls (map size – 128\*128, router locations – 64)

In the third test the speedup for different numbers of walls for simulation the range of the routers algorithm has been checked. Number of the walls changed from 0 to 930, in increments of 10. Results of experiments are show in Figure 4. Like in the first test, speedup increases with the complexity of the problem, until it stabilizes at a certain level. An average speedup is about 43.5.

During testing of algorithms that optimizes the routers placement, as previously the impact of three parameters on the execution time has been checked: the size of the map, the number of possible routers locations and the number of walls. For all investigated algorithms 1000 iteration have been performed during experiments. Similarly to tests performed for simulation of routers range square maps of different sizes were used, the walls have been spaced in the random way and routers have been evenly distributed on the map. Suppression of walls and the routers signal strength were set as previously. As one of the parameters of the objective function the number of WLAN users has been used. Users were equally distributed on the map, every 8 units (1 unit is 1 meter). Data concerning the walls have not a practical interpretation, because their spatial distribution, number and parameters do not affect the computation time. For the SA and NS algorithms some parameters, which influenced on the quality of obtained results should be set [7]. Due to that the aim of these tests is to compare the execution times of computation on the GPU and the CPU, not the quality of solutions, the values of these parameters are considered as satisfactory, if the NS and SA algorithms give results comparable to the pruning algorithm. Therefore these value have been fixed during experiments basing on the results obtained from the pruning algorithm.



a. 64 places for routers, 20 routers, 10 wall                      b. map size 128\*128, 3 routers

Fig. 5. Speedup as a function of map size (a) and possible routers location (b)

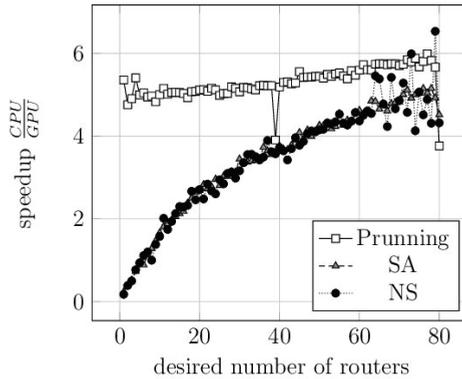


Fig. 6. Speedup as a function of the number of used routers (map size  $128 \times 128$ , possible routers locations - 64, 10 walls)

In the first test the influence of the map size on the speedup has been checked (figure 5a). The size of map changes from  $64 \times 64$  to  $1024 \times 1024$ . The best results have been received for pruning algorithm. In the next test the influence of the number of possible routers locations on the speed of computation has been checked (figure 5b). The number of possible routers locations changes from 4 to 225. For a small number of routers locations, for all algorithms, the CPU has advantages over the GPU. For the pruning algorithm the speedup was received when the number of possible routers locations was greater than 9 and increases to value close to 8. In the case of NS and SA algorithms, implementation on the CPU performs better even for larger number of possible routers locations. It is because of the time needed to calculate the value of objective function depends on the size of the map and the number of routers, not the possible routers locations. In the last test the influence of the number of used routers

on the speed of computation has been evaluated. The used number of routers changes from 1 to 80 (figure 6). In this test pruning algorithm from the beginning is characterized by the speedup over 1, when for NS and SA algorithms for small problem sizes CPU outperforms GPU.

## 5. CONCLUSIONS

The project is in the current study, the first obtained results are very promising. When using GPU for simulation the range of the routers received speedup was up to 46 for the most of tested configurations, even for small problem sizes GPU outperform when comparing with CPU. For algorithms used for optimization of router placement results are not so good, however still satisfied. In general the execution time at the CPU is better for small problem sizes (small maps), with exemption of pruning algorithm, which requires a lot of computation for large number of possible router locations. For the simulated annealing and neighborhood search algorithms for large maps sizes and number of used routers speedup was up to 10, when for pruning algorithm up to 15. From the above it can be concluded that optimising algorithms can be classified into two classes, the first that need a lot of computation and these are suitable for parallelization and other that are not suitable for parallelization because there are not computation intensive.

## REFERENCES

- [1] AWE-Communications, *AWE-communications*, 2014. <http://www.awe-communications.com/>.
- [2] GARCIA M., FERNANDEZ-DURAN A., ALONSO J., *Automatic planning tool for deployment of indoor wireless local area networks*, Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly, pp. 1428–1432, 2009.
- [3] GNATOWSKI A., *Solving wireless routers planning problem within the building using GPU*, Wroclaw University of Technology, Faculty of Computer Science and Management, MSc – 2014.
- [4] GUNTHER J., POPOV S., SEIDEL H., SLUSALLEK P., *Real-time ray tracing on gpu with bvh-based packet traversal*, Interactive Ray Tracing – RT '07, IEEE 2007, pp. 113–118.
- [5] GUMMANDI R., WETHERALL D., GREENSTEIN B., SESHAN S., *Understanding and mitigating the impact of RF interference on 802.11 networks*, ACM SIGCOMM Computer Communication Review, 37(4), pp. 385–396, 2007.
- [6] IBWAVE DESIGN ENTERPRICE, *In-Building Wireless Network Design Software—iBrave Design*, iBrave, 2014, <http://www.ibwave.com/Products/iBwaveDesignEnterprise/>.
- [7] KAMENETSKY M., UNBEHAUN M., *Coverage planning for outdoor wireless LAN systems*, Proceedings of Broadband Communications – 2002, Zurich Seminar.
- [8] NVIDIA, *CUDA C Programming Guide*, v6.0, February 2014. [http://docs.nvidia.com/cuda/pdf/CUDA\\_C\\_Programming\\_Guide.pdf](http://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf).
- [9] PEDERSEN G., *Digital Mobile Radio Towards Future generation Systems*, COST Action 231 – Final Report, European Communities, 1999.

Krzysztof JUSZCZYSZYN\*

## **A MIXED APPROACH TO THE LINK PREDICTION PROBLEM IN DYNAMIC COMPLEX NETWORK**

This chapter proposes a mixed approach to the link prediction problem in complex, dynamic social networks. Due to extreme link volatility in short timeframes most predictors return bad results when applied to such networks. Hence we propose a weighted sum of three predictors based on totally different assumptions – time series analysis, global and local network topology (simple time series analysis, Katz and TTM predictors). It is shown that this approach results in very good results, with average score exceeding 10% which is in general an unusual result for link prediction problem and is very hard to achieve in large dynamic networks.

### 1. INTRODUCTION – DYNAMIC SOCIAL NETWORKS

The complexity and dynamics are inherent properties of technology-based social networks. As a result, they are very difficult to investigate in terms of traditional social network analysis methods that can effectively cope with static networks of size up to few hundred nodes. Currently, one of the main challenges is to investigate the evolution of networks at the right level of granularity and the dynamics of this evolution. In technology-based networks a relation between two individuals is a result of set of discrete events (like emails, phone calls, blog entries) about which the knowledge is available. Because these events have some distribution in time, this adds a new dimension to the known problems of network analysis [11]. As shown in [9] for various kinds of human activities related to communication and information technologies, the probability of inter-event times (periods between the events, like sending an email) may be expressed as:  $P(t) \approx t^{-\alpha}$  where typical values of  $\alpha$  are from (1.5, 2.5). This distribution inevitably results in series of consecutive events (“activity bursts”) divided by longer periods of inactivity.

---

\* Faculty of Computer Science and Management, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland, e-mail: krzysztof.juszczyszyn@pwr.wroc.pl

The standard approach to dynamic complex network is to divide the available timeframe into windows to compute the chosen structural network properties for networks created on the basis of data from these windows [10]. This should show how the measures like node centrality, average path length, group partitions etc. change over time, providing an insight into the evolutionary patterns of the network. However, the bursty behaviour of the users causes dramatic changes of any measure when switching from one time window to another [13][19]. There is an inevitable trade-off: short windows lead to chaotic and noisy dynamics of network measures, while long windows give us no chance to investigate time evolution of the network [14]. This opens a new research area, which encompasses a number of approaches designed to predict changes in the structure of dynamic networks [15][16]. The special case is the so-called *link prediction* problem – the estimation of probability that a link will emerge/disappear during the next time window. Liben-Nowell and Kleinberg identify a variety of topological measures as features for link prediction, the problem of predicting the likelihood of users/entities forming social ties in the future, given the current state of the network. A broad survey of link prediction methods is presented in [20]. It should be noted that most methods of the link prediction give rather poor results – the best predictors discussed in [12] can identify less than 10% of emerging links. It should be also emphasised that the networks analysed in [12] were built from arxiv publication record which differs significantly from our test cases (email social networks) presented below. Email networks are highly dynamic in short timescales and – for big networks, the number of disconnected pairs of nodes increases quadratically while the number of links grows only linearly [21].

Based on our previous research, which shows that the distribution of subgraphs in complex networks is statistically stable and typical for the considered network even in the face of significant structural changes [17], and it is possible to use it in the link prediction problem [8] we have applied the method from [8] in an experiment evaluating chosen predictors against the sparse and dynamic social network.

In this work we propose an application of weighted mixed link predictors with various properties to link prediction problem in highly volatile and time-dependent dynamic social network. It is shown, that this approach allows to gain much better results than any of the predictors used separately.

## 2. LOCAL TOPOLOGY OF COMPLEX NETWORKS

Standard approaches exploiting network analysis by means of listing several common properties, like the degree distribution, clustering, network diameter or average path lengths often fail when applied to complex networks [1]. During last years we have experienced the development of a number of methods investigating complex networks by means of their local structure (especially – frequent patterns of

connections between nodes). Investigation of evolutionary dynamics of the network at the link level leads to time series analysis, which can be used to predict if the link will emerge or not, during a given period of time, based on the known history of the link presence in the past. However, it is often reasonable to consider higher-order topological structures, which leads to the subgraph analysis.

The simplest, and therefore the most popular way to characterize the network in the context of current configuration of the local connections is to examine the links between the smallest non-trivial subgraphs, the triads (a.k.a *network motifs*), consisting of three nodes. If we additionally decide to distinguish between the nodes (which is our case, for our network they are corporate email addresses) we obtain 64 patterns of possible connections between any three identifiable nodes.

In our former research we have investigated the local structure of numerous technology-based networks, among them an e-mail social network of Wroclaw University of Technology (WUT), consisting of more than 5 800 nodes and 140 000 links [2], [3]. Our aim was to check if the known properties of local topology in social networks (known on the basis of motif analysis conducted for small social networks [4]–[7]) are also present in large email-based social structures, and if there are some distinct features characteristic to the email communication. The most important conclusion from these experiments was that the general motif profile of the network (expressed by so-called *triad significance profile* – *TSP* – a vector of the Z-score measures of the motifs) is stable over long periods of time [17]. These observations led to the idea of characterizing the evolutionary patterns of the network by means of the changes in elementary subgraphs, in this particular case – directed triads. In the following section we define Triad Transition Matrix as an indicator for characterizing inherent network dynamics at local topology level [8].

In this research in order to compute the connection score between a pair of nodes we have decided to use an approach that sums over the collection of all paths between the nodes, damped by their length, which allows including the neighbourhood of the node in the analysis. Out of the number of methods which are based on path analysis we have chosen a prediction algorithm which uses Katz centrality measure [18], shown in the survey of Lieben-Nowell as one of the best link predictors [12]. In the following section we present all the methods we have used in more detail.

### 3. LINK PREDICTION METHODS

The evaluation of link prediction methods in our experiments was based on the principles proposed in [12]. It was assumed there, that all the predictors assign a predicted connection weight  $score(x, y)$  to unlinked pairs of nodes  $\langle x, y \rangle$ , based on the input graph, and then produce a ranked node pair list  $L$  in decreasing order of  $score(x, y)$ , whose values are treated as proportional to the estimated probability of

forming a new link between  $x$  and  $y$ . In this way each link predictor outputs a ranked list of node pairs which would eventually form predicted new links. From this list (sorted in decreasing values of scores) the set of first  $n$  entries is taken, then the size of its intersection with the set of new links (of the same size  $n$ ) is computed. The percentage of the links from the predicted set, which are also present in the set of new links, is the prediction accuracy.

The three approaches used in our experiments are:

- Link state transition predictor – relatively simple and intuitive method, which uses statistics of the link presence in the previous time windows,
- TTM (Triad Transition Matrix) predictor – exploiting the changes in the connection patterns between the triads of nodes [8],
- Katz predictor – basing on Katz connection scores [12].

The above methods address the local topology features on different levels, starting from the single link, elementary subgraphs (triads) to the node neighbourhood with weights exponentially decreasing with the distance. We may think of them as acting on the link, triad and network neighbourhood level.

#### 4. EXPERIMENTS AND RESULTS

Former observations have led us to conclusion, that when dealing with sparse, fast changing networks analysed in short time windows, attention should be given to:

- Choosing effective prediction algorithms.
- Pruning the dataset in order to exclude information about abnormal nodes (in email networks: rarely active, automatic – mailing lists etc.) and links.

The approach to further pruning of our dataset, taking the above considerations into account will be described in the following section.

As we know, in every organization staff uses several modes of communication like phones, videoconferencing, external email accounts etc. Therefore in a full email server dataset we observe numerous records concerning individuals who tend to prefer other communication channels than email. This resulted in the creation of a dynamic social network consisting of individuals who use an email service as a primary mode of organizational communication (the similar approach was utilized and justified in [20]). The resulting dataset (from here called the *pruned WUT dataset*) holds data reflecting email communication between 340 users during 500 days (from February 2009 to June 2010). During this period we have observed 8733 different communication channels which are associated with the directed links in our network.

Basing on the pruned WUT dataset we have created 500 networks which reflect the internal university communication in the consecutive one-day long time windows.

Despite the smaller size, the social network of active email users preserves the dynamic characteristics already discovered when analysing the full dataset. First of all, there are periodic weekly changes in the number of links present in time windows – during weekends (especially: Sundays) it is reduced by the order of magnitude. The average number of links per window is 133.9. From Fig. 1 it is also possible to recognize the effect of summer holidays and even shorter week-long breaks associated with Christmas and Easter.

For the pruned WUT dataset we have evaluated the three link predictors described in sec. 1. In contrary to Katz predictor (which relies only on the network graph created for the preceding time window) in the case of SP and TTM predictors longer history is needed, in order to compute the necessary statistics. We have used the first 100 time windows to:

1. Assess the link state transition probabilities for all links.
2. Create 99 TTMs for transitions from one window to another. The averaged values of the entries of these TTMs were used by the TTM predictor.

After preparatory phase described above we have evaluated all predictors on networks starting from the time window no. 101. The results are presented on Fig. 1

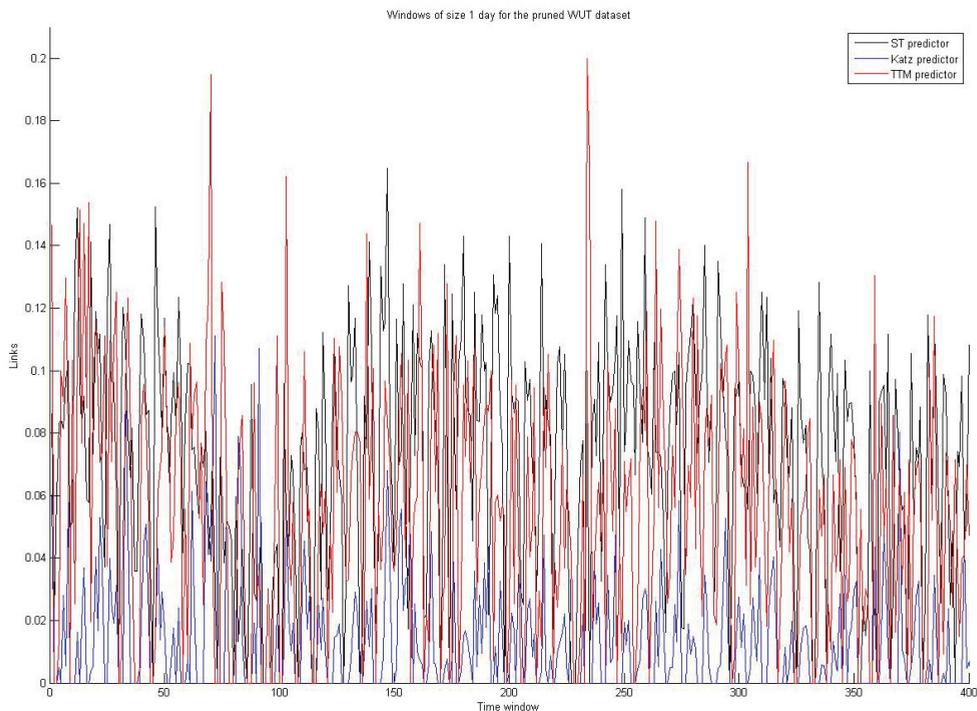


Fig. 1. The performance of SP, Katz and TTM predictors

We observe that:

- As in our first experiment with this dataset, the reduction of social network structure during weekends significantly impacts the performance of all predictors – for some time windows all of them have accuracy below 1% or even zero.
- Pruning of the dataset described in the preceding section leads to better performance of the TTM predictor (more specifically, its average performance was 6.8% and 4.4% on networks created from the full dataset). This may be explained by easier to discover dynamics of the triads reflecting the changes of communication patterns between active email users.
- The performance of the Katz predictor is lowest, which was expected due to the sparsity of the networks, for 1-day time windows they are reflecting only part of the structural information allowing to derive the future connections between the nodes which have high Katz centrality.
- The performance of SP predictor is surprisingly good (5.9% on average), however we should remember that it is very selective – the SP scores can be effectively computed only for the links already recorded in the past time windows. As a result it can choose only the links which appear frequently between a given pair of nodes. For comparison: the TTM scores are computed for all entries of the incidence matrix, and in our experiment 74% of them have non-zero values (however the majority is close to zero). This suggests that we may use the SP predictor to pinpoint the links which are to appear only because they are frequent, even if the information in current network graph is not enough to support the structural predictors.
- For the windows in which the corresponding networks have similar number of links, it is not clear which of the predictors should be preferred – it is not possible to associate the network size or time instant with the predictor which will be the best. Moreover, for different windows, different predictors show best accuracy.

We consider the last two observations to be the most important, for they suggests the way of improving the link prediction performance for this dataset – fusing the results produced by different predictors. In order to check this approach, we have defined a simple *Mixed Predictor* (MP) which uses the weighted sum of the scores of the three predictors evaluated in the first experiment. Given the scores for SP, Katz and TTM are computed and normalized (in order to have values from 0 to 1 for all of them), we define the  $score_{mixed}$  as:

$$score_{mixed}(p) = w_{TTM} \cdot score_{TTM}(p) + w_{Katz} \cdot score_{Katz}(p) + w_{SP} \cdot score_{SP}(p)$$

where:

$$w_{TTM} + w_{Katz} + w_{SP} = 1$$

The weights  $w_{TTM}$ ,  $w_{Katz}$  and  $w_{ST}$  sum up to one and reflect the importance we are assigning to their respective predictors. If one of the weights equals one, it means that the MP is reduced to one of the formerly considered predictors. Starting from the already computed scores for SP, Katz and TTM, we have investigated the performance of the MP for the different configurations (combinations of the values of  $w_{TTM}$ ,  $w_{Katz}$  and  $w_{ST}$ ). The values of the weights were changing with the step of 0.1.

Fig. 2 shows the results (prediction accuracy of the MP), with the values of  $w_{TTM}$  and  $w_{Katz}$  of the vertical and horizontal axis, respectively. For all cases the value of  $w_{ST}$  is computed as  $w_{ST} = 1 - w_{TTM} - w_{Katz}$ . The entries with coordinates (0,1), (1,0) and (0,0) correspond to the performance of TTM, Katz and SP predictors when used alone (like in the first experiment).

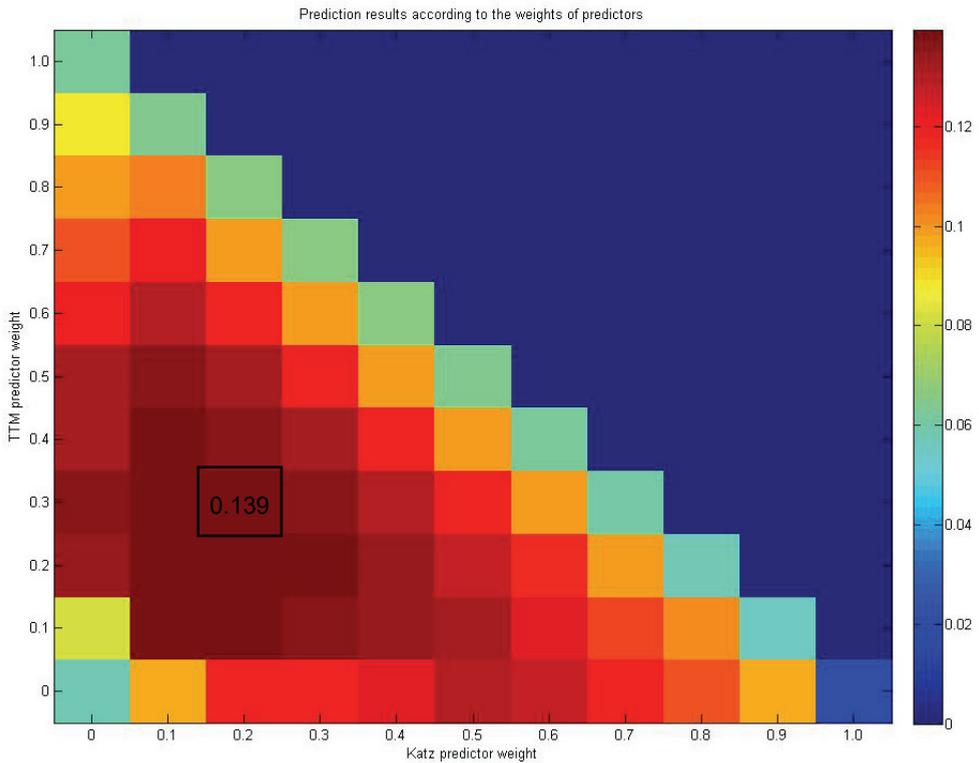


Fig. 2. Prediction results for different configurations of MP

The most important conclusions from Fig. 2 are:

- Combining predictors always leads to better performance (regardless of the weights), with one exception: Katz and TTM taken alone. The values on the diagonal show, that averaging scores returned by these two predictors degrades

the performance, which is never better than that of TTM. Interestingly, both are based on analysing the network structure – the triads and the paths in the link neighbourhood. This fact alone could suggest that it is not reasonable to mix different strategies of structural analysis, but the other values on Fig. 3 deny this interpretation:

- When all the weights have non-zero values we can achieve better predictions than for any combination of two predictors. The maximum result is 13.9% for  $w_{TTM} = 0.3$ ,  $w_{Katz} = 0.2$  and  $w_{ST} = 0.5$ . The best configuration is reached with the value of  $w_{SP}$  bigger than the others which implies that for short timescales even simple time series analysis could be as important as techniques derived from structural analysis.
- However, closer investigation shows that there are (on average) 9.55 links per time window (roughly half of the 18.6 predicted for an average window) which were positively predicted by Katz and/or TTM but could not be detected by the SP due to the lack of historical data in the past windows or its score being too small. Finally, the performance of MP at its best configuration doubles the performance of any single predictor.

Table 1. Performance of all predictors

	Result < 1% (# of time windows)	Result > 10% (# of time windows)	Best result (# of time windows)	Best result without MP (# of time windows)	Mean result
SP	67	107	2	224	<b>5.9%</b>
Katz	264	2	0	3	<b>2.1%</b>
TTM	127	98	7	173	<b>6.8%</b>
Mixed (MP)	73	343	391	-	<b>13.9%</b>

Table 1 summarizes the results, in the first four columns the number of time windows for which the respective predictors returned the best results. The fifth column shows the accuracy of the predictors.

## CONCLUSIONS

We have analysed the performance of link predictors for an extreme example of dynamic social network investigated in relatively short timescales. Our aim was to check the methods against an environment which is fast changing and undergoes seasonal reduction of communication activity, which is a feature characteristic for social networks built from the data gathered from modern communication and group work-supporting systems.

The three link prediction strategies we have used act at different levels of analysis:

- SP uses simple time series analysis, preferring the links which are frequent and emerging regardless of the higher-order network structures.
- TTM joins the statistical features of network links with their topological connection patterns. The method, although based on graph analysis, utilizes the inherent network dynamics based on the observations of the recorded network history. The method shows good performance especially in the case of sparse, dynamic networks analyzed in short time scales.
- Katz predictor is a classic technique shown to be effective in the case of many networks, mostly analysed in longer periods (like citation networks or food webs). Here, we have illustrated its usefulness even under conditions that should potentially exclude structural analysis due to short time frame and not enough of structural information stored in network graphs.

We have shown that fusion of methods scaling up in the network structure (from links to broad node neighbourhood) is beneficial – we demonstrated how this approach may improve the performance of link prediction.

We also conclude that factors influencing changes of the network structure are mixed and vary from individual interactions between individuals (which implies that simple time series analysis is quite effective in short timescales) to local evolutionary patterns (triad transitions) and the influence from the node neighbourhood (Katz link centrality).

#### REFERENCES

- [1] ITZKOVITZ S., MILO R., KASHTAN N., ZIV G., ALON U. (2003) *Subgraphs in random networks*, Physical Review E, 68, 026127.
- [2] JUSZCZYSZYN K., MUSIAŁ K., KAZIENKO P. (2008), *Local Topology of Social Network Based on Motif Analysis*, 11th International Conference on Knowledge-Based Intelligent Information & Engineering Systems, KES 2008, Croatia, Springer, LNAI.
- [3] KASHTAN N., S. ITZKOVITZ S., MILO R., ALON U. (2004) *Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs*, Bioinformatics, 20 (11), 1746–1758.
- [4] MILO R., SHEN-ORR S., ITZKOVITZ S., KASHTAN N., CHKLOVSKII D., ALON U. (2002) *Network motifs: simple building blocks of complex networks*, Science, 298, 824–827.
- [5] MANGAN S. ALON U. (2003) *Structure and function of the feedforward loop network motif*, Proc. of the National Academy of Science, USA, 100 (21), 11980–11985.
- [6] MANGAN S., ZASLAVER A. ALON U. (2003) *The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks*, J. Molecular Biology, 334, 197–204.
- [7] VAZQUEZ, A., DOBRIN, R., SERGI, D., ECKMANN, J.-P., OLTVAI, Z.N., BARABASI, A., 2004. *The topological relationship between the large-scale attributes and local interaction patterns of complex networks*, Proc. Natl Acad. Sci. USA 101, 17 940.

- [8] JUSZCZYSZYN K., MUSIAŁ K., BUDKA M., *Link prediction based on subgraph evolution in dynamic social networks*, IEEE International Conference on Privacy, Security, Risk and Trust and IEEE International Conference on Social Computing, PASSAT/SocialCom 2011, 9–11 October 2011, Boston, USA, IEEE Computer Society, 2011. pp. 27–34.
- [9] BARABÁSI A.-L., *The origin of bursts and heavy tails in humans dynamics*, Nature 435, 207 (2005).
- [10] DYNES S., GLOOR P., LAUBACHER R., ZHAO Y., *Temporal Visualization and Analysis of Social Networks*, NAACSOS Conference, Pittsburgh PA, June 27–29, 2004.
- [11] KLEINBERG J., *The convergence of social and technological networks*, Communications of the ACM Vol. 51, No.11, 66–72, 2008.
- [12] LIEBEN-NOWELL D., KLEINBERG J.M., *The link-prediction problem for social networks*. JASIST (JASIS) 58(7), pp. 1019–1031, 2007.
- [13] BRAHA D., BAR-YAM Y., *From Centrality to Temporary Fame: Dynamic Centrality in Complex Networks*, Complexity, Vol. 12 (2), pp. 59–63, 2006.
- [14] KEMPE D., KLEINBERG J., KUMAR A., *Connectivity and inference problems for temporal networks*, Journal of Computational System Science, 64(4):820–842, 2002.
- [15] M. LAHIRI, TANYA Y. BERGER-WOLF: *Mining Periodic Behavior in Dynamic Social Networks*, ICDM pp. 373–382, 2008.
- [16] SINGH L., GETOOR L., *Increasing the Predictive Power of Affiliation Networks*. IEEE Data Eng. Bull. (DEBU) Vol. 30 No. 2, pp. 41–50, 2007.
- [17] JUSZCZYSZYN K., MUSIAŁ K., KAZIENKO P., GABRYS B., *Temporal Changes in Local Topology of an Email-Based Social Network*, Computing and Informatics 28(6): 763–779, 2009.
- [18] KATZ L., *A new status index derived from sociometric analysis*, Psychometrika, 18:39–43, 1953.
- [19] BATAGELJ V., MRVAR A., *A subquadratic triad census algorithm for large sparse networks with small maximum degree*, Social Netw. 23, 237–243, 2001.
- [20] GETOOR L., DIEHL C.P., *Link mining: a survey*, ACM SIGKDD Explorations Newslett., Vol. 7, pp. 3–12, 2005.
- [21] HUANG Z., LIN D.K.J., *The Time-Series Link Prediction Problem with Applications in Communication Surveillance*, INFORMS Journal on Computing, Vol. 21, No. 2, pp. 286–303, 2009.

**PART 2**  
**INTERNET OF THINGS USE CASES**



Michał DYK\*, Andrzej NAJGEBAUER\*, Dariusz PIERZCHAŁA\*

## **AUGMENTED PERCEPTION USING INTERNET OF THINGS**

Internet of Things (IoT) is considered a big change in IT world or even a revolution. It is true that number of connected devices grows fast every day. Nowadays, the big questions are how to utilize such amount of data which comes with IoT and what is the role of human in it. We propose a concept of *augmented perception* in which connected smart devices can support our senses. In that approach human is the central element of the IoT, and can gather new information by describing his need. It is responsibility of smart devices to satisfy them. We chose macroprogramming method as a way to change behaviour of the network as a whole. In this paper we also introduce SenseSim simulator which allows us to build augmented perception virtually.

### **1. INTRODUCTION**

Internet Of Things (in short IoT) is currently very fast growing phenomenon. Some of the authors claim that we are even facing new Industrial Revolution [12, 13] or at least the biggest change in IT world since the beginning of the Internet. However, the concept was developed by Kevin Ashton already at the end of the last century [11]. He defines IoT as a network of uniquely identifiable devices (or things) connected with each other in an Internet-like structure. He even claimed that the Internet of Things would be a kind of interface between the real world and the Web.

The purpose of this paper is to present the concept of the augmented perception as a way to utilize still growing and becoming more accessible IoT issue. We believe, that with proper use, ubiquitous and intelligent devices may give us the possibility to “see” more than using our natural senses. Our work is a part of the wider concept of data fusion. Most of the solutions in this field, in particular based on the JDL process [10], treats the user mainly as a consumer of the results. It is often criticized as an

---

\* Faculty of Cybernetics, Military University of Technology, gen. S. Kaliskiego Str. 2, 00-908 Warsaw.

approach which starts from the wrong end. However, in our work we place human in center of attention and focus on the information needs of the user and use the augmented perception to satisfy them.

The outline of this paper is as follows: chapter. 2 is a short comment on Big Data issue in IoT. We claim that more data does not always mean more information. We also emphasize that the information is what really user needs from the IoT. In chapter 3 we propose a formal way to describe user's information needs. It is based on infon semantic introduced by Keith Devlin in [3]. Chapter 4 introduces a concept of augmented perception. It bases on Formal Theory of Perception proposed by Bennet et al. In [2]. We claim that IoT can be used to enhance our perception to satisfy our information needs. Chapter 5 provides short description on key aspects of a simulator named SenseSim, which is able to build augmented perception virtually. Chapter 6 is a short conclusion of work described in this paper.

## 2. BIG DATA $\neq$ INFORMATION

According to CISCO, it is expected to be approximately 50 billion devices connected to the Internet by year 2020 [14]. It means that it will be 50 billion sources of data which are relatively easily accessible. In other words with IoT we are also facing a Big Data issue. It is definitely great opportunity to explore the World and better understand as well as monitor both our environment and society. However we still have "only" data and no matter how big it is and how do we manipulate or explore it we won't get nothing more. As "more" we understand an information.

To make the difference clear it is important to define what the data and information are. We will follow Davenport et al. [4] who describe data as a set of discrete, objective facts about events. For example, when a customer does shopping as a data we can consider what he or she bought; the quantity of items; the price etc. Basing on those facts we do not know why the customer chose those particular shop and it is impossible to predict if he or she is going to come back next time. That is because there is no inherent meaning in data, but it is in information. An information can be understand (just like Davenport et al.) as a *message*. It has a sender and a receiver. The receiver decides what is information for him and what is not. It has an impact on our judgment and behaviour. Originally the verb "to inform" meant "to give shape to" and information is meant to shape the person who gets it, to make some difference in his outlook or insight.

Therefore having more data does not mean that we know more or we can make better decisions. In other words 50 billion expected data sources is not everything to have more and better information. That is why we propose an approach which allows user to define his needs for information.

### 3. INFORMATION NEEDS

Consider a situation when someone wants to park a car in a crowded city centre. One important information which she or he needs can be “*there is (or isn't) free parking place in range of 100 m*”. We call it a need for information and will mark as  $n$ . Having that we are looking for response for the need, which we mark as  $i_n$ . In the given case  $i_n$  will be just a binary value taking true when there really is free parking place in range of 100 m and false otherwise. Of course it is very simple case (more accurate need for information could be “*location of the nearest parking place*” and response could be GPS coordinates), but is useful to show basic concept of our approach. Having  $n$  and  $i_n$  we are looking for a transformation  $k$ , which satisfies basic statement:

$$n \xrightarrow{k} i_n. \quad (1)$$

In such a case we require a formal approach to describe an information and our need for an information.

In the information theory there are many definitions which describe information in a mathematical way. The fundamental work was done by Claude Shannon, who introduced a measure for information [6]. It allows to count how many bits are needed to transfer a message. However Shannon's theory does not take into account a content of the message which is crucial for our work. Another scientist, Fred Dretske, used part of the Shannon's work to build his own theory [7]. The most important element of his approach is that, he distinguished two forms of information: *analog* and *digital*. Example of the analog form of information can be a photo of a car. Watching it we extract, during cognition process, some facts about it. As an output of this process we get an information in a digital form. Dretske claimed that information in its digital form should be understand as fact that some object  $a$  has a property  $P$ . Keith Devlin in his work [3] extended Dretske's idea and introduced concept of *infor* which is item of (digital) information. In a matter of fact it is very useful approach which can be used to describe our needs for information in a formal way.

An infor  $\sigma$ , in its basic form, can be understand as a fact that some objects  $a_1, \dots, a_n$  are in some relation  $R$ . It is described in the following way:

$$\sigma = \langle \langle R, a_1, \dots, a_n, i \rangle \rangle \quad (2)$$

where  $R$  is  $n$ -place relation and  $a_1, \dots, a_n$  are objects appropriate for  $R$ . Element  $i$  is called polarity of the infor and can take values 1, if objects  $a_1, \dots, a_n$  are really in relation  $R$ , and 0 otherwise. Moreover basic infor description can be extended by adding elements which describe spatial location  $l$  and temporal location  $t$ . Full infor semantic has the form of:

$$\sigma = \langle\langle R, a_1, \dots, a_n, l, t, i \rangle\rangle. \quad (3)$$

Keith Devlin, in his theory, claims that natural source of information about the world are situations. Only in particular situation  $s$  we can tell if the infon  $\sigma$  is factual. To indicate that we write:

$$s \models \sigma. \quad (4)$$

It should be read as " $s$  supports  $\sigma$ ". Situation  $s$  in this case is not a part of the real world. We call it an *abstract* situation which is a mathematical construct. Of course there is an intuitive sense in which to every real situation corresponds an abstract one. We understand abstract situation as a set of infons:

$$\{\sigma \mid s \models \sigma\}. \quad (5)$$

From this point we can say that the need for information  $n$  can be formally described by an abstract situation using presented notation of infons. In other words we want to verify if some hypothetical situation like "*there is free parking place in range of 100 m*" exists in the real world. We can describe it by infons in the following way:

$$s = \{\langle\langle atLocation, me, l, t, 1 \rangle\rangle, \langle\langle dist100m, me, parkingPlace, l, t, 1 \rangle\rangle\}. \quad (6)$$

First infon describes that the user (object called  $me$ ) is at a specific location  $l$  at given time  $t$  and the second one describes that there is free parking place within 100 m from the object  $me$ . If that verification is successful, then we say that  $s$  becomes *actual* situation and we can write:

$$s \equiv S \quad (7)$$

where  $S$  is corresponding to  $s$  a real situation. In such a case  $i_n$  (which is response for our need) takes the value of true. There are some conditions which must be met to consider  $s$  situation as actual. First of all, the description of an abstract situation in infon's notation must be *coherent*. It means that it should satisfy following conditions [3]:

- (i) for no  $R$  and  $a_1, \dots, a_n$  is it the case that:  
 $s \models \langle\langle R, a_1, \dots, a_n, l, t, 0 \rangle\rangle$  and  $s \models \langle\langle R, a_1, \dots, a_n, l, t, 1 \rangle\rangle$ ,
- (ii)  $s \models \langle\langle same, a_1, a_2, 1 \rangle\rangle \Rightarrow a_1 = a_2,$  (8)

(iii) for no  $a$  is the case that  $s \models \langle\langle \text{same}, a, a, 0 \rangle\rangle$ .

In fact being coherent is not enough to claim that situation  $s$  is actual, but only coherent situations could correspond to real ones. This definition assumes that exists predefined, two-place relation *same*, which corresponds to genuine equality of given objects. To be able to conclude, that situation  $s$  is actual we must be sure that its every piece of information (infon) is correct in sense of the real world. It means that actual situation should meet, in addition to the coherence requirement, yet another two conditions [3]:

(i) whenever  $s \models \langle\langle R, a_1, \dots, a_n, 1 \rangle\rangle$  then in the real world it really is the case that  $a_1, \dots, a_n$  stand in the relation  $R$ ,

(ii) whenever  $s \models \langle\langle R, a_1, \dots, a_n, 0 \rangle\rangle$  then in the real world it really is the case that  $a_1, \dots, a_n$  do not stand in the relation  $R$ .

Presented conditions show that it is two-step process to verify if abstract situation really is an actual. It is also an essence of transformation  $k$  introduced previously. Firstly a description of the need for information must be consistent by satisfying coherence conditions and secondly objects in real world should be in relations described by infons. While the first step is quite easy to validate, the second one needs communication with the real world. That is where the task for Internet of Things with its smart sensors begins.

#### 4. AUGMENTED PERCEPTION

Growing number of devices connected to Internet of Things gives us great opportunity to “see” more. Matt Welsh from Harvard University said once that sensor networks implement a concept of “macroscope”, which means that they are a scientific instrument that observes entire system [1]. He distinguished three fundamental features of such observer:

- 1) Observe the world (environment, buildings, people, etc.) at very high spatial resolutions.
- 2) Make these observations continuously.
- 3) Collect the observations in digital form.

The above features are also adequate for Internet of Things (which is a special case of a sensor network). Moreover it is possible to push some “intelligence” into a network,

which makes the observation process smart. Basing on those facts we claim that IoT phenomenon can be used to expand our perception.

Just as we did with information, we also try to describe perception in a formal manner. We based our model upon the Formal Theory of Perception [2] proposed by Bennet et al. In that theory an individual observer is a six-tuple:

$$O = \langle X, Y, E, S, \pi, \eta \rangle, \quad (9)$$

where  $X$  and  $Y$  are measurable spaces,  $E$  and  $S$  are subsets of  $X$  and  $Y$  respectively,  $\pi$  is a surjective function with domain in  $X$  and values in  $Y$ ,  $\eta$  is so called conclusion kernel. Figure 1 shows how an observer works.

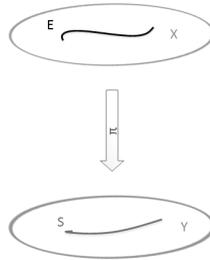


Fig. 1. Visualization of an observer. Source: on the basis of [2]

When an  $O$  observes it does not perceive the object of perception, but rather a representation of some property of the interaction. Space  $X$  represents all properties of relevance to the observer. It is mathematical representation of a part of the external world which the observer is somehow interested in. Space  $Y$  represents all premises about events which occur in  $X$ . Based on those premises the observer can conclude what happen in the outside world. Function  $\pi$  in that case is a called perspective of the observer and is responsible for the transformation of events from space  $X$  into premises in space  $Y$ . For example, in a digital camera the lens is a physical realization of  $\pi$ , because it transforms 3D scene into 2D picture on digital image sensor. Set  $E$  holds those configurations (scenarios) which represent phenomena of interest for the observer. In forest fire observation it can be fire or smoke [9]. Set  $S$ , on the other hand, holds the premises which correspond to given configuration  $E$ . The last element of the observer's definition is the conclusion kernel  $\eta$ . Its role is to provide measure which tells what is the probability that with given set of premises  $S$ , an event  $E$  occurred.

Presented construction of an observer is a very handy way to model, in a formal manner, various sensors (an example could be system for forest fire observation [9]). Moreover they can work together. In such a case conclusions from one observer can become premises for other one as shown in Figure 2 (left side).

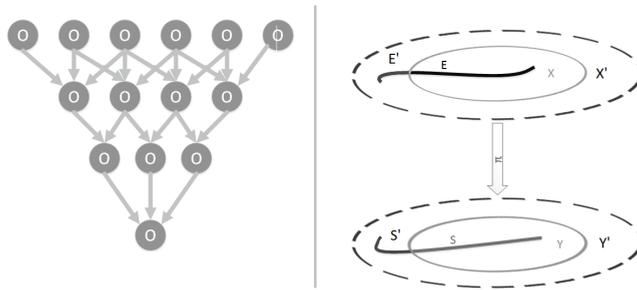


Fig. 2. Hierarchy of the observers and augmented perception concept. Source: own preparation

We can utilize this fact to build an *augmented perception*. Figure 2 (right side) helps to explain that term using cited definition of observer. Let's assume that spaces  $X$  and  $Y$ , define configurations and premises of human senses. Using them we can only perceive some limited part of the world. Basing on the considerations from chapter 4, we defined our need for information as “*there is (or isn't) free parking place in range of 100 m*”.

In this context, set  $E$  represents the free parking places and set  $S$  represents corresponding premises which we can collect using our senses (for example, the sense of sight). In many situations, this can be not enough to satisfy our need. As we said before, observations from smart devices can become our premises, which means they may expand our space  $Y$  into  $Y'$ , thus, we can obtain more premises about the interesting free parking spaces, hence the set  $S$  expands to  $S'$ . Naturally our space  $X$  expands as well into  $X'$ . In such a case the set of observable parking spaces expands from  $E$  to  $E'$ . In other words, the appropriate use of intelligent, ubiquitous devices can give us the possibility to sense more.

Naturally, having a lot of sensors in IoT does not mean that our perception becomes automatically expanded. Behaviour of the devices should be adapted to the given need for information. That is one of the reasons why we define our needs in a formal way. After that we can transform them into macroprogram for sensors, which will verify if relations described by infons exist in the real world. Macroprogramming is a way to program network of devices as a whole. There is one, high level, program which is distributed between sensors. Their responsibility is to interpret the program and adjust their behaviour to it. With respect to nomenclature which we introduced in chapter 4, macroprogramming is a way to realize transformation  $k$ .

We currently evaluate Abstract Task Graph (in short ATaG) as a macroprogramming language [8], where programs are graphs. Figure 3 shows an simple example of a program which monitors temperature. It contains two tasks *Sampler* and *Cluster-Head* which are annotated with launch and installation rules (*Sampler* is installed once on

each sensor and *Cluster-Head* on one node in cluster). Sampler simply collects observations in *Temperature* data item, which is then transferred to *Cluster-Head*.

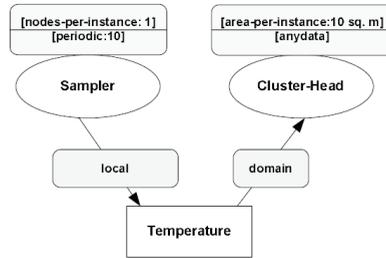


Fig. 3. Example of macroprogram in ATaG language. Source: [8]

## 5. SIMULATION FRAMEWORK

There are many sensor networks simulators that can also be used to research over IoT. However, they mostly focus on the technical aspects of the connection and communication between nodes. As examples we can mention: OMNeT++, Opnet, NS 2, NS 3, Exata. On the other hand, there are solutions, such as CupCarbon or Multi-source-level Simulator on Report which can simulate the interaction between the environment and sensors, however it is limited and difficult to extend or adjust. For our study of the augmented perception the most important is the ability to freely define the observed phenomena or sensors and compliance of the simulator with a Formal Theory of Perception. That is why we propose simulator named SenseSim which has, among others, the following features:

- It generates sensory data from heterogeneous devices;
- Sensors can move according to given routes or randomly within the limits of selected algorithms;
- It is compatible with IoT concept;
- It allows to adjust network behaviour by macroprogramming (with possibility to support different languages);
- Sensors observe a phenomena which can change during experiments. Every phenomenon can be freely reconfigured.
- It is open for cooperation with other simulators like NS3 or Opnet.

Devices in SenseSim are modelled as eight-tuple, which can discretely change its state during an experiment:

$$s(t) = \langle L(t), V, r, \{O_i\}_{i=1..n}, \{z_i^i\}_{i=1..n}, P(t), R(t), SNT(t) \rangle, \quad (10)$$

where  $L(t)$  is geographical localization of the device at time  $t$ ;  $V$  is a speed;  $r$  is a radio range (for wireless communication);  $\{O_i\}_{i=1..n}$  is a set of observers (which are consistent with introduced in chapter 4 definition) associated with every perceptual ability  $i$  of the device;  $\{z_i^i\}_{i=1..n}$  is a set of collected observations from every ability;  $P(t)$  is a set of programs executed on the device,  $R(t)$  is a set of available resources at time  $t$ ;  $SNT(t)$  represents part of the network which given device know. A network of the devices is modeled as an undirected graph, which can change its topology during simulation time:

$$SN(t) = \langle S, E(t), b(t, s_1, s_2) \rangle, \quad (11)$$

where  $S$  is a set of devices, which build the network;  $E(t)$  is a set of edges which represent direct communication between devices;  $b$  is a function defined on edges, which returns an available network bandwidth at simulation time  $t$ . Every edge means that two devices can directly communicate.

We assume that sensors gather data by observing the phenomena which occur in the world. An phenomenon we define as a four-tuple:

$$e(t) = \langle R(t), \{s_i(t)\}_{i=1..n}, t_b, t_e \rangle, \quad (12)$$

where  $R(t)$  is a set of points which represents geographical area, where the phenomenon occurs;  $\{s_i(t)\}_{i=1..n}$  is a set of functions which describes observation for each ability  $i$  at time  $t$ ;  $t_b$  and  $t_e$  describe moment of time when phenomenon begins and ends respectively.

SenseSim is an agent-based discrete-event simulator written in Java. It has modular structure, which allows to easily expand its functionality. For example module responsible for wireless communication between devices can be replaced with more complex one if the communication issues should be more precisely simulated.

## 6. CONCLUSIONS

In this work we introduced a concept of *augmented perception*. It is one of many ways to utilize the benefits of the growing IoT issue. We propose an method which allows user to define his needs for information and network of smart devices should satisfy it. It is different approach than in many data fusion systems based on JDL process, which firstly gather observations from sensors and then proceed to explore in-

formation from them. We consider described in this paper concept as more flexible and intuitive. Moreover it ensures that given output is really information for the user.

Our method is still under development. Further works will focus on building more complex ontology of the relations which can be used in infons and on algorithms which will allow network to learn that ontology dynamically.

#### ACKNOWLEDGMENTS

This work was partially supported by the National Centre for Research and Development: ROB 0021/01/ID21/2 and DOBR/0069/R/ID1/2012/03.

#### REFERENCES

- [1] WELSH M., *Where do we go from here? The Big Problems in sensor networks*, Keynote, Wireless Sensing Solutions Conference, Chicago, IL, September 28, 2005.
- [2] BRUCE M. BENNET, DONALD D. HOFFMAN, CHETAN PRAKASH, *Observer Mechanics. A Formal Theory of Perception*, Academic Press, 1989, ISBN 0-12-088635-9.
- [3] DEVLIN K., *Logic and information*, Cambridge University Press, 1991, ISBN 0-521-49971-2.
- [4] DAVENPORT T., PRUSAK L., *Working knowledge: how organizations manage what they know*, by Thomas H. Davenport and Laurence Prusak. Harvard Business School Press, 1998. 199p index afp ISBN 0875846556,
- [5] LA RUE F., *Report of the Special Rapporteur on the promotion and protection of the right to freedom of opinion and expression*, UN Human Rights Council, Seventeenth session, 16 May 2011.
- [6] SHANNON, C.E., *The mathematical Theory of Communication*, Bell System Technical Journal, 1948,
- [7] DRETSKE F., *Knowledge and the Flow of Information*, Bradford Books, MIT Press, 1981,
- [8] BAKSHI A., PRASANNA V., REICH J., LARNER D., *The Abstract Task Graph: A Methodology for Architecture-Independent Programming of Networked Sensor Systems*, Workshop on End-to-End, Sense-and-Respond Systems, Applications, and Services (EESR 05), June 5, 2005.
- [9] ŠERIĆ L., STIPANIČEV D., ŠTULA M., *Agent Based Sensor and Data Fusion in Forest Fire Observer*. In: Sensor and Data Fusion, Dr. ir. Nada Milisavljević (Ed.), In-Teh, 2009, ISBN 978-3-902613-52-3
- [10] ŠERIĆ L., STIPANIČEV D., ŠTULA M., *Sensor and Data Fusion*, Dr. ir. Nada Milisavljević (Ed.), In-Teh, 2009, ISBN 978-3-902613-52-3
- [11] ASHTON K., *That 'Internet of Things' Thing*. RFID Journal, 22 June 2009. <<http://www.rfidjournal.com/articles/view?4986>>
- [12] GLEN M., *How The Internet Of Things Is More Like The Industrial Revolution Than The Digital Revolution*. Forbes Magazine, 10 Feb. 2014. <<http://www.forbes.com/sites/oreillymedia/2014/02/10/more-1876-than-1995/>>.
- [13] JOHNSON S. *The 'Internet of Things' Could Be the next Industrial Revolution*. MercuryNews.com. San Jose Mercury News, 01 Feb. 2014. <[http://www.mercurynews.com/business/ci\\_24835528/internet-things-could-be-next-industrial-revolution](http://www.mercurynews.com/business/ci_24835528/internet-things-could-be-next-industrial-revolution)>.
- [14] TILLMAN K. *How Many Internet Connections Are in the World? Right. Now*. Cisco Blogs, 29 July 2013. Web. 01 July 2014. <<http://blogs.cisco.com/news/cisco-connections-counter/>>.

*operational picture, data fusion, mobile systems,  
decision support, situation awareness, SOA*

Mariusz CHMIELEWSKI, Wojciech KULAS, Marcin KUKIEŁKA,  
Damian FRĄSZCZAK, Dawid BUGAJEWSKI, Jakub KĘDZIOR,  
Damian RAINKO, Piotr STĄPOR\*

## **DEVELOPMENT OF OPERATIONAL PICTURE IN DSS<sup>3</sup> USING DISTRIBUTED SOA BASED ENVIRONMENT, TACTICAL NETWORKS AND HANDHELDS**

This paper summarizes the overall experience collected during the development of Network Enabled Capabilities demonstration (NEC-DEMO) project prepared for the European Defense Agency. The initial aim triggered the development of efficient data fusion and integration methods implemented in SOA environment and delivered for specialized tactical handhelds. In order to collect and fuse operational situation of a given scenario using JC3IEDM, NFFI and TSO integration standards, this elaborated environment has been enriched with further integration options, such as Battle Management Systems and simulation software. Developed system – tCOP – serves as a server solution offering both an integration platform and a GWT-based command and control stateful portal. One of the most crucial part of the system is mCOP, the Android-based mobile application for tactical level commanders serving as a toolkit providing current operation situation, portable combat simulator and a tactical communicator.

### 1. INTRODUCTION AND MAIN CONCEPT

The main concept of the project and the demonstration itself was to prepare set of server and mobile software components, which will provide scalable environment capable of developing and delivering COP (*Common Operational Picture*) and CTP (*Common Tactical Picture*) products for both tactical and operational level commands.

---

\* Cybernetics Faculty, Military University of Technology, gen. Kaliskiego 2, 00-908 Warsaw, Poland.

<sup>1</sup> Decision Support Systems.

Crucial features of the environment deliver distributed, SOA (*Service Oriented Architecture*) based, and reliable set of integration and decision support services, tuned for web and mobile access. The application of new technologies has been done intentionally in order to verify the NEC (*Network Enabled Capabilities*) concepts in military tactical networks and commercially available technologies for dedicated security-oriented solutions.

tCOP is dedicated for a decision support of tactical level command. System has been developed using state of the art technologies and best practices. Supplied tools enable allied (NATO) commanders to increase their productivity of mission planning and operation execution. tCOP is one of few web-based applications used for heterogeneous data integration and COP development. mCOP mobile application is dedicated to the lowest levels of command supporting efficient operational data flow and automatic monitoring of forces (Fig. 1a). The solution supports delivery of detailed battlespace information concerning objects, units, equipment and installations their combat potentials and tasks. Using the application the soldier is able to gather specific tactical information on equipment, warfare as well as unit resources and tasks. Furthermore, detailed combat information contain not only static data, but also description of unit's behavior in time (tasks).

## 2. TCOP SOFTWARE'S REQUIREMENTS ACQUISITION PROCESS

Despite requirements being modelled in a conventional way, the "untypical" term can be still applied to this projects specification gathering phase, as the requirements were gathered in quite an uncommon fashion:

- very few general stakeholder requests described in the contract;
- minimal ability to discuss with the stakeholder or users which greatly impeded identification and further understanding of requirements;
- the discovery of features desired by the stakeholder has been regarded as one of the project goals;

In case of our project the stakeholder requests refer to the area of NEC capability demonstration. We found the stakeholder requirements to be general and curt, that is why the process of investigating similar technical solutions has been applied.

Features of the created software were verified in cooperation with stakeholders and the development team. Features are a product of reaching an agreement regarding extent and understanding of the final product. Prepared use cases and supplementary specifications translate the stakeholder's product perception. In applied approach we have decided that the basic model can be applied, however, the process of obtaining the requirements and roles should be different than usual.

The tCOP and mCOP on the other hand aim at utilizing state-of-the-art technologies and satisfy the demand for innovative command support features. Specificity of this project enforced further refinement of the tCOP and mCOP application features. Elaborated process for requirements acquisition and modeling was a compromise between known, proved methods and innovative elements forced by the specificity of the project. This modeling process has been placed as a template and a good practice to be applied in other C4ISR (*Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance*) related projects.

### 3. SITUATION AWARENESS DEVELOPMENT USING SENSORS AND AUGMENTED REALITY

SA (*Situation Awareness*) has been defined in [1] as “the perception of the elements in the environment within a volume on time and space, the comprehension of their meaning and projection of their status in near future” and is now being a subject of study in a number of domains [2]. The system utilizes handhelds at the lowest levels of command for the development of near real-time tactical picture and commander’s situation awareness.

The mCOP application automates many monitoring activities such as geo location, intensity of activity, speed, bearing and even vital functions of a commander or a soldier. The process consists of two concurrent activities. The first one is aimed at the battlefield situation’s data acquisition and delivery. (Fig. 1c–d) The data provided by the sensors and obtained from the subordinates is derived and filtered on the handheld itself and then sent to the tCOP server for further processing. The data describes the movement parameters of the unit based on the fused data from the unit’s staff. The second activity is oriented at acquisition of the battlefield situation’s data from the server. The tCOP server obtains data from the automatized processes as well as from the template-based messages and other sources. The tCOP server narrows collected data to match the situation in the vicinity of the user. Then the data is sent to the mCOP devices via sustained feedback connection.

The enrichment of users’ SA is achieved by presenting inertial sensor data and using Augmented Reality perspectives containing the operational information. (Fig. 1b) For SA development we combined common handheld sensors’ data such as GPS, magnetometer, accelerometer, and gyroscope. Furthermore mCOP delivers efficient route planning capabilities and combat scenario simulation assessment. To enrich the communication capabilities within the tactical network mCOP and tCOP provides secure and reliable template based ADat-P3 military messaging.

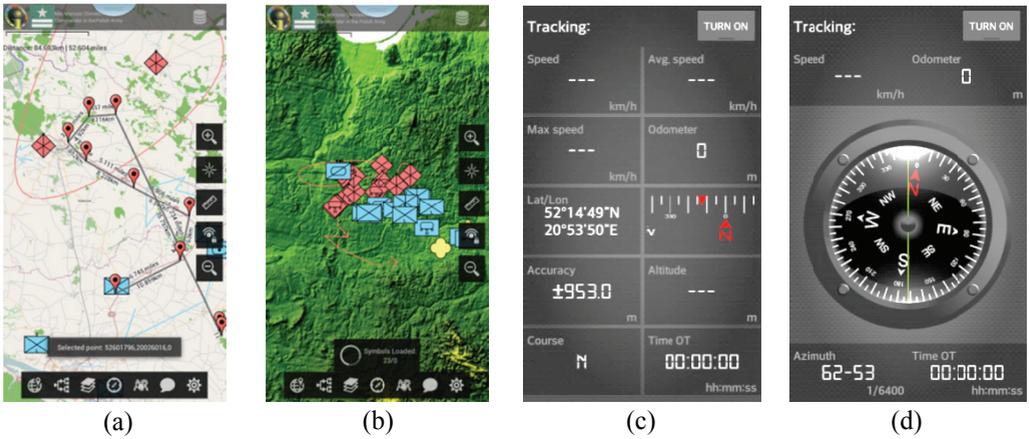


Fig. 1. Features of mCOP mobile toolkit demonstrating: the operational scenario collected from the military decision system, merged operational picture fusing allied systems, sensor fusion data, compass

Military operations are conducted mostly in difficult conditions in terms of the environment and the information. In such situation the lack of visibility and proper orientation is curtail problem general problem of locating and identifying enemy and friendly forces better situational awareness is required for effective operation.



Fig. 2. mCOP's Augmented Reality view (left). Operation's data fusion schema (right)

Believing that situational awareness in these circumstances cannot be met by using traditional approaches we decided to research AR (*Augmented Reality*) view.

The Augmented Reality view provides the data fusion products in form of one picture on which the application evaluates the positions of allied and enemy forces within the range of operation supplemented with the current observer location and device orientation. (Fig. 2 left) This information can be further semantically processed to

recommend the commander most efficient course of action or movement route. Augmented Reality is construct of the data about battlefield, location, data derived from sensor fusion and above all the camera view (Fig. 2 right). To create the AR view there is a need to describe the vertical and horizontal field of the camera's view. This is done by measuring the visible angle one meter away from the camera. Specifying the angles and maximum distance to the shown units allows further view creation. Decision whether the unit should be displayed or not is taken on the basis of the following formula

$$visible = bearingTo(unit) \leq \frac{azimuth}{2} \wedge distanceTo(unit) \leq max \quad (1)$$

and the position of the unit marker on top of the camera display is computed by formulas

$$x = \frac{width}{2} + \frac{bearingTo(unit) - azimuth}{horizontalFOV} \cdot width, \quad y = \frac{height}{2} + \frac{pitchTo(unit) - pitch}{verticalFOV} \cdot height \quad (2)$$

where:

- *width* – width of screen,
- *height* – height of screen,
- *bearingTo(·)* – a function that returns a bearing<sup>2</sup> to the specified object,
- *pitchTo(·)* – a function that returns a pitch to the specified object,
- *distanceTo(·)* – a function that returns a distance to the specified object,
- *azimuth*, *pitch* – an azimuth<sup>3</sup> and pitch of the device
- *horizontalFOV* – horizontal field of view,
- *verticalFOV* – vertical field of view.

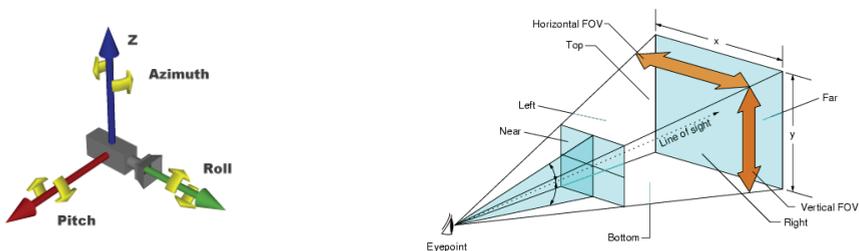


Fig. 3. The camera's azimuth, roll and pitch (left) [3]. Horizontal and vertical FOV (right) [4]

<sup>2</sup> A bearing is an angle less than 90° within a quadrant defined by the cardinal directions (N,E,S,W). [15]

<sup>3</sup> An azimuth is an angle between 0° and 360° measured clockwise from North. [15]

Creating advanced sensor based views requires fusing the sensors output to obtain reliable information. In created solution data from inertial sensors and magnetometer are derived to obtain the position of the handheld in 3-dimensional space described by the Euler angles, pitch, roll and yaw, with yaw being nothing else than the azimuth. (See Fig. 3) Moreover these angles have to be appropriate despite of the orientation of the phone, meaning that the coordinates system is changing along with orientation of the device. When the device is held flat the azimuth is based on the direction where the top of the device is pointing.

$$value = \alpha \cdot old\_value + (1 - \alpha) \cdot new\_value \quad (3)$$

To smooth the sensor data a low-pass filter has been applied. The handheld creates a data buffer which helps to evaluate the present value which is derived based on the new data sample and the previous one (3). We have also experimented with Kalman filters, but because of the efficiency we discarded that solution.

#### 4. CRISIS OPERATION DATA INTEGRATION AND FUSION

The fact that the operational data is often supplied from various, heterogeneous sources causes many integration problems on technical, syntax and semantics. Implementing data fusion process provides data association and state estimation methods. Data processed in such way are more reliable and accurate which is significant in terms of developing multi-resolution operational picture for crisis management support.

Data association is commonly used in cluttered environments (meaning that targets are close to one another). These situations hinder tracking targets and accurately determine their position. It is worth mentioning that data association is the key element due to higher levels of fusion process, like e.g. state estimation, which leads to incorrect behaviour if data association is incoherent. In order to improve the reliability of developed environment, we used three data association techniques and algorithms: nearest-neighbour algorithm, K-means algorithm and probabilistic data association including credibility factor (Fig. 2 right). Obtained results are passed through majority voting mechanism. This approach reduces disadvantages of separately used methods providing reliable and accurate data association.

State estimation techniques have been used in order to determine, a state of the target especially while moving. The ultimate goal of these methods is to obtain a global state of the target basing on the given observations. During the revision of available sources for conducting an estimation process, one of most flexible methods has been applied – the particle filter. They cope successfully with both non-linear dependences and non-Gaussian density functions in dynamic systems and noise. Nevertheless, in order to obtain low variance, the filter needs huge quantities of data, which comes at the cost of computational power. In order to reduce the amount of particles a signifi-

cant assumption has been made – tracking of the targets is processed individually with an assumption that they do not interfere with any particles. Estimated state of the target enriches the COP product and provides additional information to support a decision maker. In developed prototype this method is used mainly to estimate the route of the object and determine its combat potential.

Utilised heterogeneous data sources deliver data, which can be considered to be insignificant or invalid. We implemented methods which extract the important data by utilizing fusion techniques. Such algorithms support the reliability and accuracy of the COP and situation awareness [5] development thus improving decision support tools. An important aspect of the system is its data integration and acquisition capabilities. Such features in tCOP have been implemented as NFFI (*NATO Friendly Force Information*) standard services and JC3IEDM (*Joint Consultation Command & Control Information Exchange Data Model*) compliant database.

NFFI [6] is an interoperability standard initiated in 2005 by ACT (*Allied Command Transformation*) in response to an urgent need to exchange friendly force tracks between national C2IS (*Command and Control Information System*) within the ISAF (*International Security Assistance Force*) coalition. In July 2012 was merged with STC (*SHAPE Technical Centre*) into the NCI Agency (*NATO Communications and Information Agency*). TSO (*Tactical Situation Object*) [7] defines standard for data representation and a semantic model of a situation representation for crisis management and dispatching system perceived by a particular observer at a particular time. It is utilised to transfer data from such a view to console system or a dispatcher.

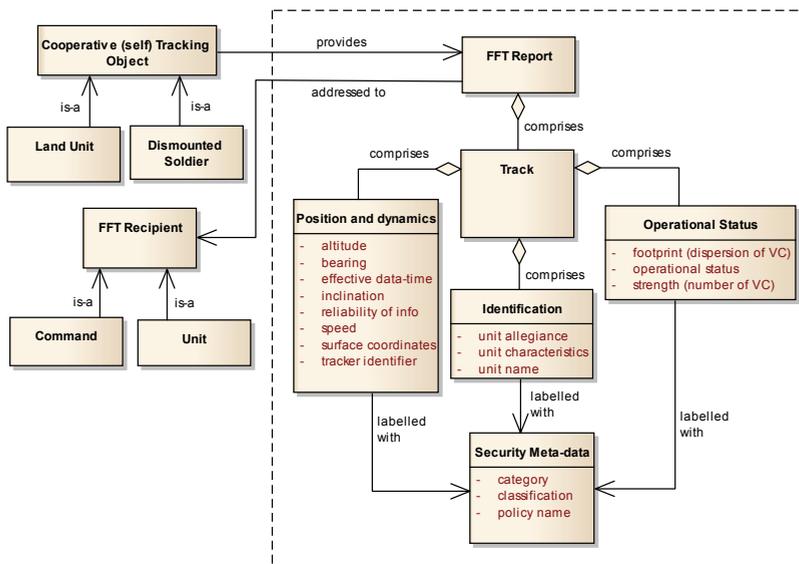


Fig. 4. NFFI message structure – based on specification provided in [6]. The acronym FFT stands here for Friendly Force Tracking. VC is the abbreviation for “vehicles covered by a single track” [8]

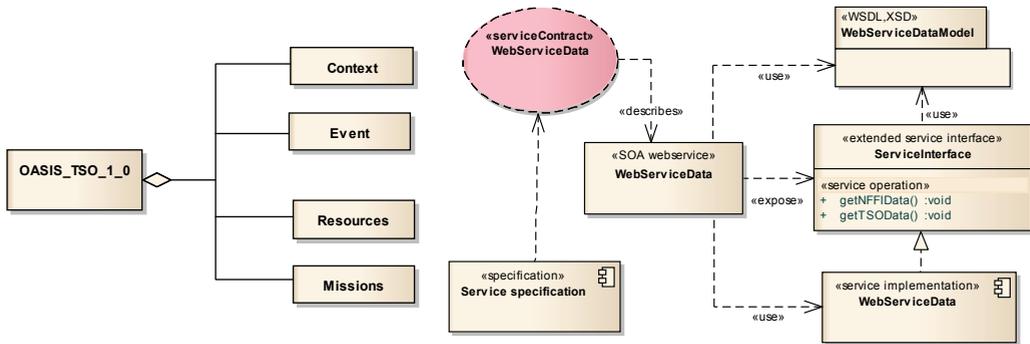


Fig. 5. TSO message (OASIS\_TSO\_1\_0) structure (left) [9], NFFI service contract in tCOP integration layer (right). WSDL - *Web Service Definition Language*; XSD - *XML Schema Definition*

The TSO standard defines minimal requirements for data interoperability between government agencies during the disaster and emergency operations. The TSO standard is a specification for information exchange between emergency response institutions such as the police, border guard, fire department, etc. The standard can be mainly used to coordinate and integrate emergency response forces in crisis operations mostly by sharing the operational picture. TSO specification has been defined and published in the disaster and emergency management [9], and formulates message structure consisting of: context, event, resources, and missions' descriptions (Fig. 5 left). It is worth mentioning that the information describes both static and dynamic parts of crisis operation. tCOP system provides both standards data exchange. It can download and deliver data in NFFI (Fig. 4) and TSO (Fig. 5 left) format. External data are mapped to tCOP scenario structure in order to view tactical situation. Scenarios that were created by tCOP can be translated to either NFFI or TSO standard.

Integration in IT refers to such a systems' configuration, that each one of them is capable of communicating with the other. In the described project we went for the SOA (Service Oriented Architecture) approach. Web services use SOAP protocol for communication and XML mark-up language to represent messages' structure. tCOP uses web services (e.g. Fig. 5 right) to communicate with other similar, JC3IEDM compatible, command support systems – e.g. JAŚMIN.

## 5. SYSTEM FUNCTIONALITIES AND RCHITECTURE

A command center distributed application is an interactive system for monitoring, supervising tasks and formulating unit tasks. The rendering of a COP utilizes data fusion algorithms and integration connectors. The major functionality of any military decision support system is the ability to provide GIS (*Geographic Information System*)

support for reflecting current or planned operational situation expressed in tactical symbology. The system should represent combat scenarios in form of pure relational data requires additional algorithms to present the situation picture, which can further be reused by the decision makers [10]. These tools enable allied (NATO) commanders to increase their SA (*Situation Awareness*), which in case of our system can be understood as a perception of battlespace environment within a volume of time and space, and the projection of their status in the near future [11]. Such an approach enables collaboration between environment elements, improves synchronization and reduces delays of the communication. mCOP simplifies the delivery of CTP and COP to every soldier in that particular military unit. Usage of smartphones and wireless tactical networks improves the availability of toolkit. Services provided by the tCOP command system provides current situation including both military and civilian objects, information concerning both in AOR (*Area of Responsibility*) and in AOI (*Area of Interest*). All requirements for COP are described in [12].

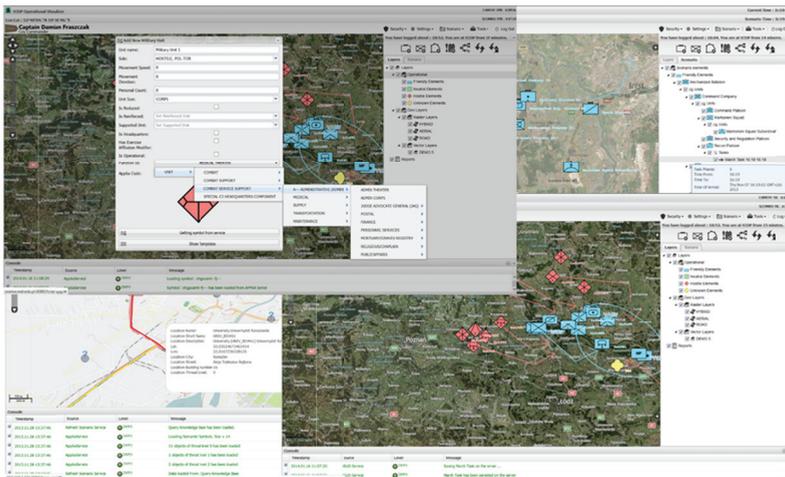


Fig. 6. The tCOP as CCA application, which runs in the web browser environment

tCOP solution allows visualization of operational situation in terms of military and civilian objects, their actions as well as equipment and supplies. It reflects the situation both by placing objects on the map and in the scenario objects tree widget representing formation's command chain, all objects and administrative structure. tCOP provides mechanism to visualize both military objects which match APP-6A standard and civilian object which match TSO standard. Our solution is not only a tool for visualization of current battlespace situation, but it is also robust application mechanisms for mission planning. The product allows management of military objects with capability of sorting and filtering them. (Fig. 6)

Our solution has been developed for Java EE, which is Oracle's enterprise Java computing platform. It provides environment for developing and running enterprise scalable, reliable, and secure network application. Java EE apps are very portable because of being vendor-neutral.

We have designed and implemented our product according to multilayered architecture so as to allocate the different responsibilities of software to different layer. Moreover, we have used component-based approach to the design, development, and deployment of the application which is compatible with SOA approach.

Business logic layer consists of SOA based services implemented using EJB 3.1 (*Enterprise Java Beans*) components. tCOP web client layer backend software communicates with EJB's using Business Delegate object's implemented over HTTP (*Hypertext Transfer Protocol*) standard: JAX-WS (*Java API for XML Web Services*) web services and GWT-RPC servlets.

Persistence layer have been implemented using JPA 2.1 (*Java Persistence API*) standard approach for ORM (*Object to Relational Mapping*) as well as EclipseLink, which has been used for both management and auditing of data.

We have reviewed some available CCA (*Command C4ISR Architecture*) products on the market and we have realized that most of them are built on the thick client model. This solution has a lot of advantages but it is impractical on the battleship because of poor motor skills. Based on this fact we have developed web-based application which removes defects of standard CCA products.

Client side has been developed according to the concept of RIA (*Rich Internet Application*). This model utilizes desktop application approach delivered as AJAX (*Asynchronous JavaScript and XML*) based client components executed in any web browser. (Fig. 6) GWT (*Google Web Toolkit*) is a set of tools and components supporting the development of RIA applications in Java language. The resulting GWT application consists of client side layer - HTML and JavaScript generated by GWT transformation engine from Java code and set of server side services (mostly asynchronous). The framework provides support for communication with the server using GWT-RPC (*GWT Remote Procedure Call*) mechanism wrapped for AJAX calls from client to server side, providing stateful user interface and rapid application delivery.

tCOP also utilizes a security layer consisting of authentication and authorization services for mobile application's functionalities access as well as for integration services. The security layer makes use of SSO (*Single Sign-On*) infrastructure to simplify distributed and multiplatform authentication. Such approach reduces time needed for logging and registering within the tactical network and simplifies user access within military C4ISR systems domain. Moreover the data and messages are decrypted using RSA algorithm to ensure high level of security.

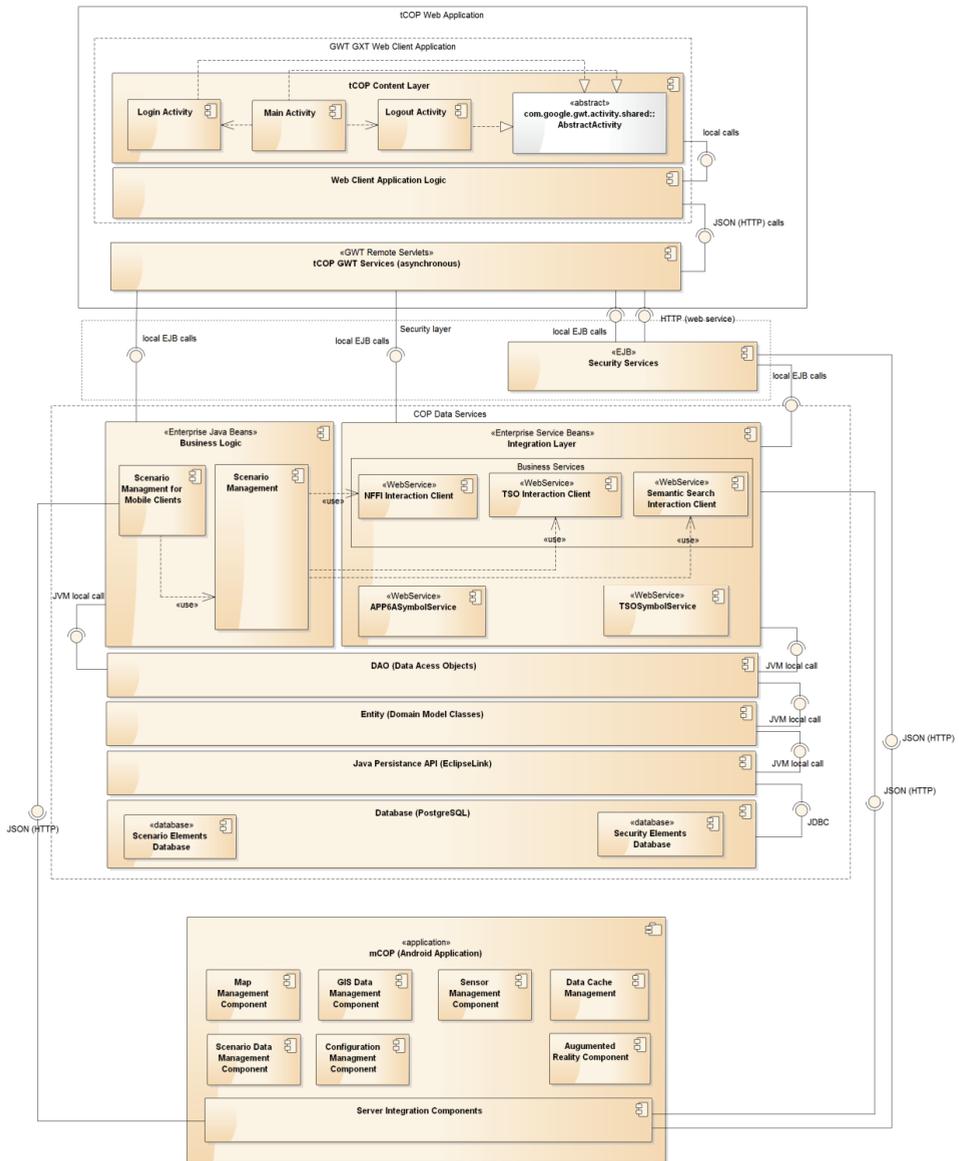


Fig. 7. The Diagram depicts our very own design of presented (tCOP and mCOP) applications. It exhibits their multi-layered architecture design as well as technologies<sup>4</sup> applied

<sup>4</sup> **JSON** (*JavaScript Object Notation*) – is a lightweight XML data exchange format; **JVM** – Java Virtual Machine; **JDBC** – Java DataBase Connectivity.

The solution provides extensive GIS mechanisms in form of SOA components dedicated at obtaining spatial data and maps. Moreover it is capable of rendering both raster and vector type terrain layers in the same time with possibility of setting order of them. User has ability both to use one of many predefined web mapping services sources: Google, OpenStreetMaps, Bing and to specify a custom web mapping service source, which is consistent with either WMS (*Web Map Service*) or WFS (*Web Feature Service*) standard. To further expand the capabilities of the spatial data visualization, KML (*Keyhole Markup Language*) and GML (*Geography Markup Language*) standards have been attached to spatial data sources.

The architecture has been outlined in Fig. 7.

## 6. SUMMARY

iCOP family of products (tCOP, mCOP) provides the information infrastructure for military and civilian crisis operation support. The tool is able to consume and produce standardised data sources in form of NFFI and JC3IEDM data messages and TSO dispatch services. The physical architecture of the prototype have been verified and validated in several configurations utilising ad-hoc tactical networks, active-active server clusters and variety of ruggedized android devices. Our product is able to process and refine operational scenario elements fusing the data from many heterogeneous data sources. We have implemented three stage data integration algorithms for technical, syntactical and semantic data interoperability utilising ontology applications and semantic bridges approach [13]. Application of Augmented Reality resulted in the increase of the decision making efficiency during combat exercises. The tools themselves have been recognised by EDA (*European Defence Agency*) officials during the EU NEC demonstration, which provided rich feedback from the command staff and operational researchers. These data may be useful in further development of iCOP application suite as well as in devising better (in terms of accuracy and efficiency) algorithms. We are planning also to expand the tCOP support for various other mobile platforms.

## ACKNOWLEDGMENTS

This work was partially supported by the National Centre for Research and Development: ROB 0021/01/ID21/2, DOBR/0069/R/ID1/2012/03, DOBR/0023/R/ID3/2013/03.

## REFERENCES

- [1] ENDSLEY M., *Design and Evaluation for Situation Awareness Enhancement*, In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 1998.
- [2] ENDSLEY M.R., GARLAND D.J., *Situation Awareness Analysis and Measurement*, CRC Press, 2000.
- [3] SPACEYE, *SpaceEyes3D Plugin SDK Documentation*, [Online]. Available: [http://www.spaceeyes3d.com/plugin/doc/group\\_camera.html](http://www.spaceeyes3d.com/plugin/doc/group_camera.html). [Accessed: 06 08 2014].
- [4] sgi, *OpenGL Performer Programmer's Guide*, [Online]. Available: [http://techpubs.sgi.com/library/tpl/cgi-bin/getdoc.cgi?coll=linux&db=bks&srch=&fname=/SGI\\_Developer/Perf\\_PG/sgi\\_html/ch02.html](http://techpubs.sgi.com/library/tpl/cgi-bin/getdoc.cgi?coll=linux&db=bks&srch=&fname=/SGI_Developer/Perf_PG/sgi_html/ch02.html). [Accessed: 06 08 2014].
- [5] CHMIELEWSKI M., *Ontology Applications for Achieving Situation Awareness in Military Decision Support Systems*, Lecture Notes in Computer Science, nr DOI: 10.1007/978-3-642-04441-0\_46, pp. 528–539, 2009.
- [6] NATO, *NATO Friendly Force Interoperability standard, STANAG 5527*.
- [7] CEN WS/ISDEM, *Definition of the OASIS Tactical Situation Object*, 2006.
- [8] EAPC, *Approval of the Friendly Force Tracking (FFT) Reference Architecture. Draft V.1*, 2013. [Online]. Available: <http://img.slivmail.com/attachments/petrenkoan/112960/0.pdf>. [Accessed: 06 08 2014].
- [9] CEN, *Disaster and emergency management – Shared situation awareness – Part 1: Message structure*, 2009.
- [10] ENDSLEY M.G.D., *Situation Awareness Analysis and Measurement: Analysis and Measurement*, Lawrence Erlbaum Associates, 2000.
- [11] CHMIELEWSKI M., *Data fusion based on ontology model for common operational picture using OpenMap and Jena semantic framework*, In Proceedings of the Military Communications and Information Systems Conference, Cracow, 2008.
- [12] NATO, *NATO Common Operational Picture (NCOP) – NC3A*, 2006.
- [13] CHMIELEWSKI A.G.M., *Semantic Battlespace Data Mapping Using Tactical Symbology*, Advances in Intelligent Information and Database Systems, nr DOI: 10.1007/978-3-642-12090-9\_14, pp. 157–168, 2010.
- [14] W3C, *Web Service Glossary*, 2011.
- [15] *Shared situation awareness – Part 1: Message structure*, CEN.
- [16] The John A. Dutton e-Education Institute - College of Earth and Mineral Sciences, „Measuring Angles,” [Online]. Available: <https://www.e-education.psu.edu/natureofgeoinfo/book/export/html/1785>. [Accessed: 06 08 2014].



*Decisional DNA, the Internet of Things,  
Set of Experience Knowledge Structure,  
Smart Systems, Knowledge Representation*

Haoxi ZHANG\*, Cesar SANIN\*\*, Edward SZCZARBICKI\*\*\*,  
Zhongyuan REN\*\*\*\*, Zhou FANG\*\*\*\*\*, Tailiang CHEN\*\*\*\*\*

## **THE DECISIONAL DNA-BASED SMART BIKE FOR INTERNET OF THINGS**

In this paper, we introduce a novel application of the Internet of Things, the Decisional DNA-based Smart Bike. The Decisional DNA is a domain-independent, flexible and standard knowledge representation structure; it allows its domains to acquire and store experiential knowledge and formal decision events in an explicit way. By using Decisional DNA, the sensor-equipped bicycle is able to learn its user's weight, riding habits, etc. Main issues in using this Decisional DNA-based approach include adapting Decisional DNA for IoT applications, capturing and storing data on IoT, and discovering and reusing knowledge from captured data. We demonstrate our approach in a set of experiments, in which the bike captures data of the user and reuses such data to distinguish its user. The result shows that the Decisional DNA can be used on IoT applications enabling knowledge capturing and reusing.

---

\* Chengdu University of Information Technology, No. 24 Block 1, Xuefu Road, Chengdu, China, haoxi@cuit.edu.cn

\*\* The University of Newcastle, University Drive, Callaghan, 2308, NSW, Australia. Cesar.Sanin@Newcastle.edu.au

\*\*\* Gdansk University of Technology, Gdansk, Poland

\*\*\*\* Chengdu University of Information Technology, No. 24 Block 1, Xuefu Road, Chengdu, China, renzhongyuan@live.com

\*\*\*\*\* Chengdu University of Information Technology, No. 24 Block 1, Xuefu Road, Chengdu, China, wamakerdevelop@gmail.com

\*\*\*\*\* Chengdu University of Information Technology, No. 24 Block 1, Xuefu Road, Chengdu, China, chentailiang@outlook.com

## 1. INTRODUCTION

During the past decade, the Internet of Things (IoT) [1][2][3] has received significant attention from industry as well as academia. The capabilities of the IoT for seamlessly integrating classical networks and networked objects [4] are the main reasons behind this interest [1], [5]. The basic idea of IoT is to connect all things in the world to the Internet, and the ultimate goal of IoT is to build an intelligent environment around us, where things can communicate with each other, make decisions by themselves, and act accordingly without explicit instructions, and even know what we need, what we want, and what we like [3], [5]. Furthermore, the great progresses on computer networks and relevant technologies make many conceptual applications possible. Therefore, more and more governments, academics, and researchers are taking part in constructing such an intelligent environment that is composed of various computing systems, such as intelligent transportation, smart health care, global supply chain logistics, and smart home [6]–[8]. And how do we transform the data captured or generated by IoT into knowledge to provide a more convenient environment to people becomes one of the most important emerging questions now.

Several IoT smartness researches and theories can be found in literature. Li et al. [11] introduced the Smart Community as a new Internet of Things application, which used wireless communications and networking technologies to enable networked smart homes and various useful and promising services in a local community environment. The smart community architecture was defined in their paper, then solutions for robust and secure networking among different homes were described, at the end, two smart community applications, Neighborhood Watch and Pervasive Healthcare, were presented. In [12], a cognitive management framework that will empower the Internet of Things to better support sustainable smart city development was presented. The framework introduced the virtual object (VO) concept as a dynamic virtual representation of objects and proposed the composite VO (CVO) concept as a means to automatically aggregate VOs in order to meet users' requirements in a resilient way. In addition, it illustrated the envisaged role of service-level functionality needed to achieve the necessary compliance between applications and VOs/CVOs, while hiding complexity from end users. The envisioned cognition at each level and the use of proximity were described in detail, while some of these aspects are instantiated by means of building blocks. A case study, which presented how the framework could be useful in a smart city scenario that horizontally spans several application domains, was also described. López et al. [13] proposed an architecture that integrates fundamental technologies for realizing the IoT into a single platform and examined them. The architecture introduces the use of the Smart Object framework [15][16] to encapsulate sensor technologies, radio-frequency identification (RFID), object ad-hoc networking, embedded object logic, and Internet-based information infrastructure. They evaluated the architecture against a number of energy-based

performance measures, and showed that their work outperforms existing industry standards in metrics such as delivery ratio, network throughput, or routing distance. Finally, a prototype implementation for the real-time monitoring of goods flowing through a supply chain was presented in detail to demonstrate the feasibility and flexibility of the architecture. Key observations showed that the proposed architecture has good performance in terms of scalability, network lifetime, and overhead, as well as producing low latencies in the various processes of the network operation. In [14], Lee et al. applied human learning principles to user-centered IoT systems. Their work showed that IoT systems can benefit from a process model based on principles derived from the psychology and neuroscience of human behavior that emulates how humans acquire task knowledge and learn to adapt to changing context.

Decisional DNA, as a standard, flexible and domain-independent knowledge representation structure [17], can potentially solve and improve the problem just stated: it not only captures and stores knowledge in an explicit and formal way [18], but also can be easily applied to various domains to support decision-making and standard knowledge sharing and communication among these systems [15].

## 2. DECISIONAL DNA AND SET OF EXPERIENCE KNOWLEDGE STRUCTURE

The Decisional DNA is a novel knowledge representation theory that carries, organizes, and manages experiential knowledge stored in the Set of Experience Knowledge Structure [20]. The Set of Experience Knowledge Structure (SOEKS or shortly SOE) has been developed to capture and store formal decision events in an explicit way [18]. It is a flexible, standard, and domain-independent knowledge representation structure [17]. And is a model based upon available and existing knowledge, which must adapt to the decision event it was built from (i.e. it is a dynamic structure that depends on the information provided by a formal decision event) [21]; moreover, SOEKS can be stored in XML or OWL files as ontology in order to make it transportable and shareable [19,] [22].

SOEKS consists of variables, functions, constraints and rules associated in a DNA shape enabling the integration of the Decisional DNA of an organization [21]. Variables normally implicate representing knowledge using an attribute-value language (i.e. by a vector of variables and values) [23], and they are the centre root and the starting point of SOEKS. Functions represent relationships between a set of input variables and a dependent variable; besides, functions can be applied for reasoning optimal states. Constraints are another way of associations among the variables. They are restrictions of the feasible solutions, limitations of possibilities in a decision event, and factors that restrict the performance of a system. Finally, rules are relationships between a consequence and

a condition linked by the statements IF-THEN-ELSE. They are conditional relationships that control the universe of variables [21].

Additionally, SOEKS is designed similarly to DNA at some important features. First, the combination of the four components of the SOE gives uniqueness, just as the combination of four nucleotides of DNA does. Secondly, the elements of SOEKS are connected with each other in order to imitate a gene, and each SOE can be classified, and acts like a gene in DNA [21]. As the gene produces phenotypes, the SOE brings values of decisions according to the combined elements. Then, a decisional chromosome storing decisional “strategies” for a category is formed by a group of SOE of the same category. Finally, a diverse group of SOE chromosomes comprise what is called the Decisional DNA [18].

### 3. DECISIONAL DNA-BASED SMART BIKE

The Decisional DNA-based Smart Bike is designed and developed to assist the user’s riding by using knowledge learned from the user’s or other people’s previous riding. Moreover, the smart bike can also estimate whether the rider is the user (owner) of it through comparison of rider’s weight and riding habits. In this section, we introduce the system architecture, and main hardware components of the smart bike concept.

#### 3.1. SYSTEM ARCHITECTURE

In order to achieve the goal stated at the beginning of this section, the conceptual four-layer architecture is designed for the Decisional DNA-based Smart Bike. These four layers are: Physical Layer, Embedded Operating System Layer, Decisional DNA Layer, and Application Layer (see Fig. 1).

The *Physical Layer* consists of the computing units, networking hardware, memory, sensors, peripherals, and networking transmission technologies of the smart bike. It is a fundamental layer underlying the data of the higher level functions in the smart bike.

At the second level, there is the *Embedded Operating System Layer* that manages the computing hardware of the smart bike system and provides interfaces between the Decisional DNA, the applications, and the underlying hardware platform.

The *Decisional DNA Layer* is the key software component of our research. It is designed to work as the “brain” of the smart bike application: It analyses and routes data, learns from data, manages knowledge, cooperates with other mechanisms, and interacts with the smart bike application. The Decisional DNA Layer is composed of a set of middleware applications between the Embedded Operating System Layer and the Application Layer, namely: *Interface*, *Diagnoser*, *Prognoser*, and *Knowledge Repository*. The

*Interface* connects the Decisional DNA Layer with its underlying layers, and provides knowledge-based services and functionality to the upper layer. The *Diagnoser* is the place where the smart bike's scenario data are gathered and organized. In our case, we link each experience with a certain scenario describing the circumstance under which the experience is acquired. Scenario data are essential for learning and estimating the smart bike's status. The *Prognoser* is in charge of analyzing scenario data, and creating experiential knowledge based on machine learning algorithms. The *Knowledge Repository* is where experiential knowledge is stored and managed. It uses XML described in [17] to store knowledge, which makes standard knowledge sharing and communication become easier.

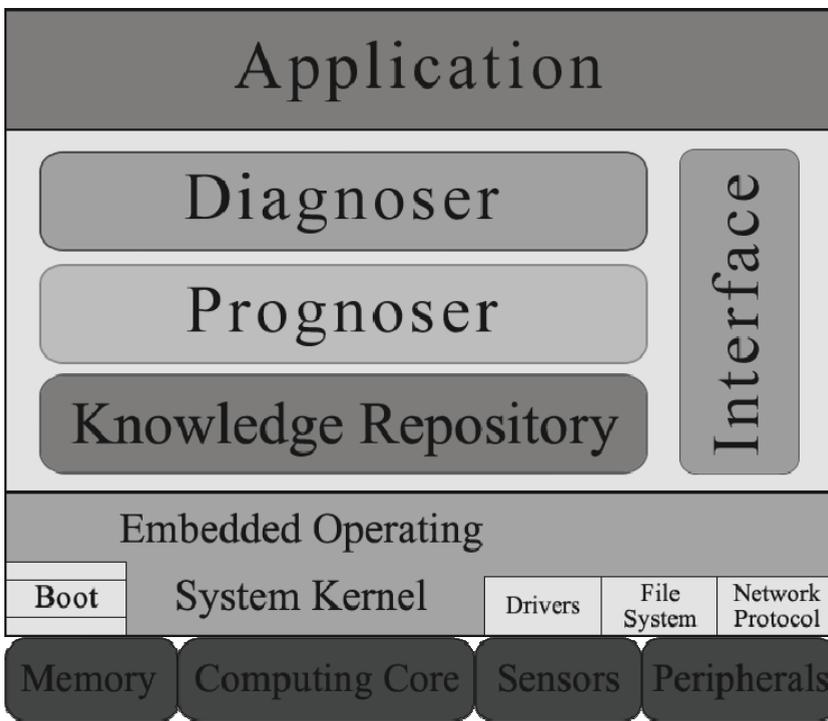


Fig. 1. System Architecture for Decisional DNA-based Smart Bike

Finally, the *Application Layer* is at the top. There are applications developed to fulfil different tasks and offer various functionalities to the end-user; and with the help of the Decisional DNA Layer, these applications and the whole system, become intelligent with capabilities of acquiring, reusing, improving and sharing knowledge.

### 3.2. MAIN HARDWARE COMPONENTS

The main hardware components of Decisional DNA-based Smart Bike consist of a NXP LPC1769 board [24], a HC-06 Bluetooth module, and a set of sensors. The NXP LPC1769 is an ARM 32-bit Cortex-M3 Microcontroller with MPU, CPU clock up to 120MHz, 64kB RAM, 512kB on-chip Flash ROM with enhanced Flash Memory Accelerator. It supports In-Application Programming (IAP) and In-System Programming (ISP), has eight channel general purpose DMA controller, nested vectored interrupt controller, AHB Matrix, APB, Ethernet 10/100 MAC with RMI interface and dedicated DMA, USB 2.0 full-speed Device controller and Host/OTG controller with DMA, CAN 2.0B with two channels, four UARTs, one with full modem interface, three I2C serial interfaces, three SPI/SSP serial interfaces, I2S interface, General purpose I/O pins, 12-bit ADC with 8 channels, 10-bit DAC, and four 32-bit timers with capture/compare. The NXP LPC1769 board is easy to use, low power, and very handy to have different peripherals and sensors working together. Through the HC-06 Bluetooth module, the board is able to communicate with other devices, such as a smart phone, so that the captured data can be sent for further processing.

## 4. EXPERIMENTAL TESTING OF THE CONCEPT

We implemented the first prototype of the Smart Bike concept with two MD-PS002 pressure sensors based on the hardware platform introduced in the previous section. The aim of this experiment is to evaluate the usability of the Decisional DNA applying to Internet of Things. First, as a whole new exploring in combination of knowledge representation and IoT applications, whether the Decisional DNA can be adapted for IoT must be examined. Second, the capability of knowledge capturing of Decisional DNA on IoT needs to be tested. Finally, we experiment the smartness of our application by asking it to remember its user.

In order to examine the adaptability of Decisional DNA, we changed the file format of the Decisional DNA from XML to plain text so that the captured data can be organized and stored as SOEKS on the NXP LPC1769 board. Then, two MD-PS002 pressure sensors were attached to the two tires of the Smart Bike to collect the real-time tire pressure data. We measure the tire pressure every one minutes both for two tires (i.e. the front tire and the rear tire). Table 1 illustrates a sample of tire pressure data collected at one time. The ID is used to differentiate two tires: number one stands for the front tire, while number two stands for the rear. Besides pressure, date and time are captured at the same time too, they are collected for future use, such as learning the user's riding schedule.

Table 1. A sample of tire pressure data captured

ID	Pressure (bar)	Date	Time
1	1.60	2001-05-03	10:12:43
2	1.52	2001-05-03	10:12:43

By organizing and sending the captured data to an Android phone via Bluetooth connection, tire pressure information is collected, stored, and ready for post-processing. Finally, we introduced the FarthestFirst [24] algorithm to help Smart Bike learn its user’s normal weight and distinguish its user from other riders: we collect tires’ pressure when the user is riding for some time, then if we change the rider, the bike is able to detect the change from the tire pressure differences. Fig. 2. and Fig. 3 illustrate the results of the tire pressure clustering for front tire and rear tire separately on Weka[25]. As we can see from these two figures, the FarthestFirst algorithm clusters the tire pressure correctly for the user (cluster A, marked as cross) and the another rider(cluster B, marked as solid dot).

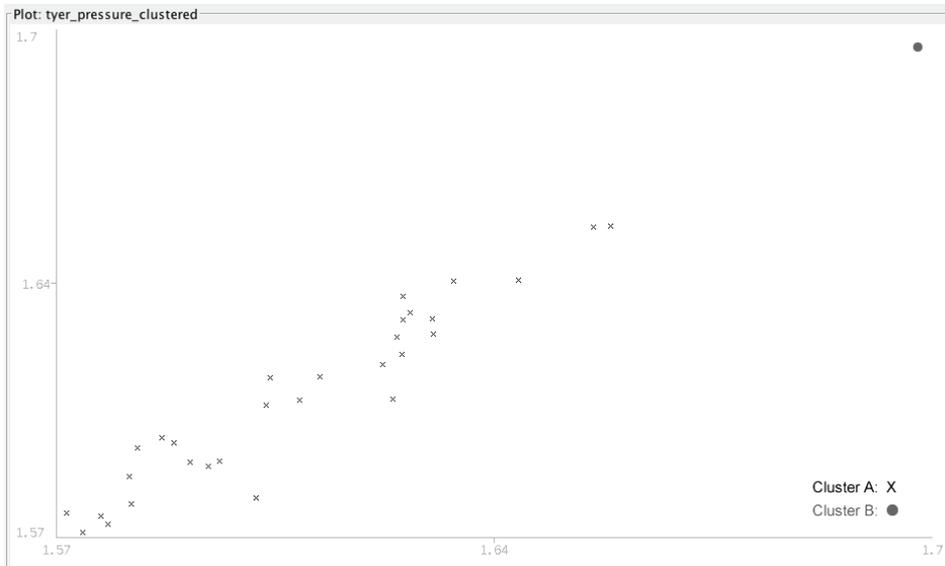


Fig. 2. Result of the front-tire pressure clustering

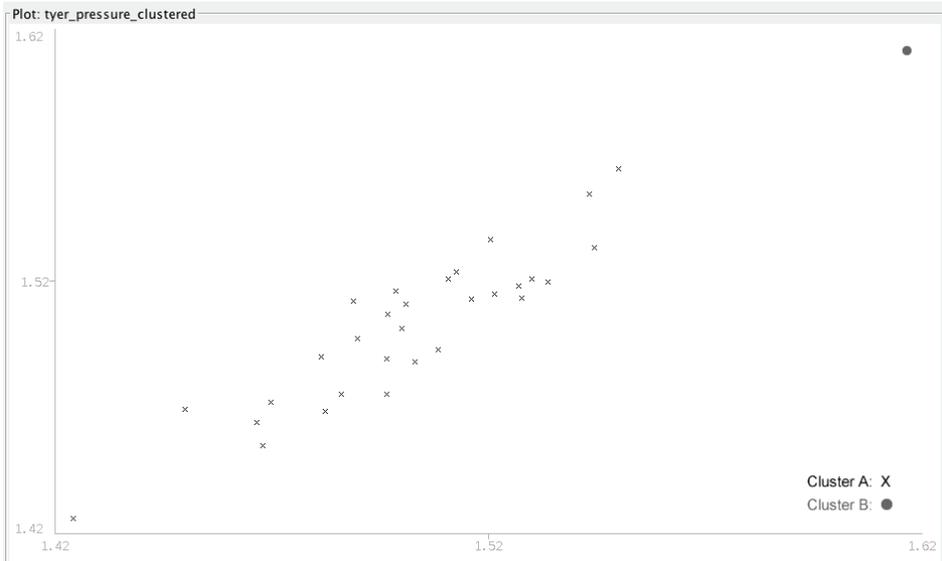


Fig. 3. Result of the rear-tire pressure clustering

## 5. CONCLUSION AND FUTURE WORK

In this chapter, we presented an initial approach that allows a bicycle to capture and reuse its own experiences to distinguish its user from other riders by using Decisional DNA. Moreover, the adaptability and usability of the Decisional DNA applying to IoT has been tested through the implementation and experiments.

As the research of the Decisional DNA-based Smart Bike is at its early stage, there are quite a few further improvements and refinements remaining to be done, some of them are:

- Refinement of the requirements of Decisional DNA-based Smart Bike, such as inter-process communication protocols, life cycle management, and battery saver.
- Further development and enhancement of the compactness and efficiency of Decisional DNA applied to IoT.
- Evaluation and comparison of different knowledge discovery approaches in order to optimize the data mining algorithm.
- Further design and development for multi-sensor supporting.

## REFERENCES

- [1] ATZORI, Luigi, Antonio IERA, and Giacomo MORABITO. *The internet of things: A survey*. Computer networks 54.15 (2010): 2787–2805.
- [2] K. ASHTON, *That 'Internet of Things' Thing*, 2009, RFID Journal, Available at <http://www.rfidjournal.com/article/print/4986>.
- [3] TSAI, C., et al., *Data Mining for Internet of Things: A Survey*. (2013): 1–21.
- [4] KORTUEM, Gerd, et al., *Smart objects as building blocks for the internet of things.*, Internet Computing, IEEE 14.1 (2010): 44–51.
- [5] PERERA, Charith, et al., *Context aware computing for the internet of things: A Survey*. (2013): 1–41.
- [6] BANDYOPADHYAY, Debasis, and Jaydip Sen. *Internet of things: Applications and challenges in technology and standardization*, Wireless Personal Communications 58.1 (2011): 49–69.
- [7] DOMINGO, Mari Carmen, *An overview of the Internet of Things for people with disabilities*, Journal of Network and Computer Applications 35.2 (2012): 584–596.
- [8] MIORANDI, Daniele, et al., *Internet of things: Vision, applications and research challenges*, Ad Hoc Networks 10.7 (2012): 1497–1516.
- [9] LÓPEZ, Tomás Sánchez, et al., *Taxonomy, technology and applications of smart objects*, Information Systems Frontiers 13.2 (2011): 281–300.
- [10] LÓPEZ, Tomás Sánchez, et al., *Adding sense to the Internet of Things*, Personal and Ubiquitous Computing 16.3 (2012): 291–308.
- [11] LI, Xu, et al., *Smart community: an internet of things application*, Communications Magazine, IEEE 49.11 (2011): 6875.
- [12] VLACHEAS, Panagiotis, et al., *Enabling smart cities through a cognitive management framework for the internet of things*, Communications Magazine, IEEE 51.6 (2013).
- [13] LÓPEZ, Tomás Sánchez, et al., *Adding sense to the Internet of Things*, Personal and Ubiquitous Computing 16.3 (2012): 291–308.
- [14] LEE, Sang Wan, Oliver PRENZEL, and Zeungnam BIEN, *Applying human learning principles to user-centered IoT systems*, Computer 46.2 (2013): 46–52.
- [15] VASSEUR, Jean-Philippe, and Adam DUNKELS., *Interconnecting smart objects with ip: The next internet*, Morgan Kaufmann, 2010.
- [16] The IPSO Alliance. <http://www.ipso-alliance.org>. Accessed 4 June 2014.
- [17] MALDONADO SANIN C.A., *Smart Knowledge Management System*, PhD Thesis, Faculty of Engineering and Built Environment – School of Mechanical Engineering, University of Newcastle, E. Szczerbicki, Doctor of Philosophy Degree, Newcastle, 2007.
- [18] SANIN C., SZCZERBICKI E., *Experience-based Knowledge Representation SOEKS*, Cybernetics and Systems, Vol. 40, No. 2, 2009, 99–122.
- [19] SANIN C. and E. SZCZERBICKI, *An OWL Ontology of Set of Experience Knowledge Structure*, Journal of Universal Computer Science, 2007, Vol. 13, pp. 209–223.
- [20] ZHANG, Haoxi, Cesar SANIN, and Edward SZCZERBICKI., *Implementing Fuzzy Logic to Generate User Profile in Decisional DNA Television: The Concept and Initial Case Study*, Cybernetics and Systems 44.2–3 (2013): 275–283.
- [21] SANIN C., MANCILLA-AMAYA L., SZCZERBICKI E., and CAYFORDHOWELL P., *Application of a Multi-domain Knowledge Structure: The Decisional DNA*, Intel. Sys. For Know. Management, SCI 252, 2009, 65–86.

- [22] SANÍN C., Carlos TORO, Eider SANCHEZ, Leonardo MANCILLA-AMAYA, Haoxi ZHANG, E SZCZERBICKI, Eduardo CRASCO, Wang PENG, *Decisional DNA: A Multi-technology Shareable Knowledge Structure for Decisional Experience*, *Neurocomputing*, 88 (2012) pp. 42–53.
- [23] LLOYD J.W., *Logic for Learning: Learning Comprehensible Theories from Structure Data*, Springer, Berlin, 2003.
- [24] The NXP LPCXpresso Board for LPC1769. <http://www.nxp.com/demoboard/OM13000.html>. Accessed 4 June 2014.
- [25] HOCHBAUM, Dorit S., and David B. SHMOYS,. *A best possible heuristic for the k-center problem*, *Mathematics of operations research* 10.2 (1985): 180–184.
- [26] WITTEN, Ian H., and Eibe FRANK. *Data Mining: Practical machine learning tools and techniques*, Morgan Kaufmann, 2005.

**PART 3**

**COMPRESSION, RECOGNITION  
AND SCHEDULING ALGORITHMS**



Aleksandr KATKOW\*

## **ADDITIVE ALGORITHMS FOR ACCELERATED COMPRESSION OF INFORMATION**

In this article is discussed one from variants of realization of the additive spectral analysis in which the Fourier spectral analysis represents the lossy compression computational process of the algebraic summing or real signal samples or samples of basic functions taken at certain points of the interval independent variable. The application of the additive processes allows to obtain the very fast computing schemes with minimum delay, that so are necessary for real time systems. In this paper is offered a new additive algorithm for the calculation of Discrete Cosine Transform (DCT), which is widely used in the practice of converting graphics and numerical solution of differential equations. The conception of the accelerated calculation of DCT with use of the additive algorithm is considered on example of a real signal. The programs for realization of the additive algorithms for performing of the accelerated Fourier analysis and for calculating spectrum of DCT were developed. In the paper the fragments of the programs are represented in meta language.

### 1. INTRODUCTION

Nowadays, spectral analysis is a crucial part of many analytic tasks. Fast Fourier Transform (FFT) algorithms and its realization in special processors are widely used in practice. The calculation process in accordance with the FFT algorithm begins after receipt all components of the vector of values of the real signal. In the real-time processing it is associated with a certain delay. It is noteworthy that in the real-time spectrum analyzer is difficult to use algorithms for the discrete Fourier transform on the one hand due to the presence of multiplicative operations, and on the other because of

---

\* University of Computer Sciences and Skills, Lodz, Poland

the fact that to obtain high resolution in the spectral domain, it is necessary to use sufficiently extended interval analysis.

There are two implementations of additive algorithms for accelerated calculation of the Fourier integral using additive formulas, which are respectively called direct additive formulas and dual additive formulas [1, 2]. When implementing the second implementation of additive algorithm for computing the Fourier integral it is done by means of the algebraic summation of basic functions, taken at the points, which are determined by calculating the definite integral of the analyzed function and comparing its value with a certain value that called as threshold. The main idea of the basic algorithm of dual additive formulas of the Fourier transform has been proposed at [2]. Later, in [3], it was proposed to use this algorithm to compute the Fourier coefficients of the solution of differential equations in partial derivatives. The bases of theory of direct additive formulas was represented at [4]. In this paper we are considering to use additive dual formulas for performing of the Fourier analysis and for calculating the spectrum of Discrete Cosine Transform.

## 2. PRELIMINARIES

Consider in more detail a method for producing of the dual formulas of additive spectral analysis for the approximate calculation of Fourier integrals. Its essence is as follows. The technology of processing of information used in the construction of dual formulas is similar to technology of processing of information in neural networks. It is known that in neural networks individual neurons generate an output signal when the sum of original signals exceeds a certain threshold. Also in the implementation of dual AFT formulas is proposed to integrate of the investigated function and compare the value obtained with a threshold. It is important to note that the proposed approach is based on conversion of the original signal which is represented as a voltage value or as a digital code in sequence of the values of independent variable, which often can be represented as moments of time. This sequence of time is determined on the one hand by value of the original signal and the other threshold value equal to the value of the definite integral of the original signal at each integration interval. In other words, converts the coding method of the original signal. When the value of the integral of the original function equals the value of the threshold, it is proposed to fix value of the basic functions at these points. The obtained values of the basic functions are used in the dual formulas of AFT to calculate values of the spectral density of the original function.

Consider the theoretical foundations of the proposed method. If the basic part of energy of continual signals is concentrated in the limited intervals of time  $[-T/2, T/2]$  and frequencies  $[-\Omega/2, \Omega/2]$ , then we have the following approximate relations

between a signal  $F(t)$  and a spectrum  $S(\omega)$  of this signal

$$S(\omega) = \int_{-T/2}^{T/2} F(t) \exp(-j\omega t) dt, \quad (1)$$

$$F(t) = 1/2\pi \int_{-\Omega/2}^{\Omega/2} S(\omega) \exp(j\omega t) d\omega, \quad (2)$$

where

$$S(\omega) = A(\omega) + jB(\omega), \quad (3)$$

$$F(t) = f(t) + j\varphi(t). \quad (4)$$

We consider a way, how may to receive integrals (1, 2) by means of algebraic addition of values  $\sin\omega t$  and  $\cos\omega t$ , taken in discrete values of time and frequency.

The whole interval of integration in (1) we shall divide on some so small intervals, that we can present functions  $\exp(-j\omega t)$  and  $F(t)$  ( $t-t_u$ ) with an adequate accuracy by the linear members of Taylor series on intervals  $[t_u - t_{u-1}]$ . On each interval of integration we have

$$\int_{t_q}^{t_{q+1}} F(t) \exp(-j\omega t) dt = \exp(-j\omega t_u) \int_{t_u}^{t_{u+1}} F(t) dt + \delta_u^S(\omega), \quad (5)$$

where

$$\delta_u^S(\omega) = -1/2j\omega \exp(-j\omega t_u) F(t_u) (t_{u+1} - t_u)^2, t_u \in [0, T]. \quad (6)$$

Possible for a real signal  $F(t) = f(t)$  such a way determine obtained discrete values of time so that were satisfied the conditions

$$\int_{t_u^{(\omega)}}^{t_{u+1}^{(\omega)}} f(t) dt = T \lambda_u^f(\omega), \quad (7)$$

where

$${}^T \lambda_u^f(\omega) = \lambda^T \text{sign} {}^T \lambda_u^f(\omega). \quad (8)$$

It is obvious, that the sequences of discrete values of time  $t_u(\omega)$  are defined completely by function  $f(t)$ , as it is the moments, in which integrals (7) from function  $f(t)$  receive an increment  $\lambda^T$ . It is natural that the number of intervals of integration  $N^f$ , depends on  $\lambda^T$ , as a parameter, and also from function  $f(t)$ , and consequently beforehand cannot be defined.

Then we shall obtain evaluation of (1,2)

$$\tilde{A}(\omega) = \lambda^T \text{sign} {}^T \lambda_u^f(\omega) * \sum_{u=0}^{N^f} \cos \omega t_u, \quad (9)$$

$$\tilde{B}(\omega) = -\lambda^T \text{sign} {}^T \lambda_u^f(\omega) * \sum_{u=0}^{N^f} \sin \omega t_u, u \in \{0,1,2,\dots,N^f\} \quad (10)$$

what is more in calculating the values of (9) and (10) is used the same grid values of the analyzed function.

Similarly for evaluation of integral (2), it is possible to receive the dual additive formulas for determination of values of function on the given values of its spectral density [2].

$$\tilde{f}(t) = \lambda^\Omega \text{sign} \lambda_v^\Omega(t) * \sum_{v=0}^{N^\Omega} \cos \omega t_v, v \in \{0,1,2,\dots,N^\Omega(t)\}. \quad (11)$$

### 3. COMPUTER SIMULATION

We have developed a computer program to simulate of the additive algorithms and have investigated filtering properties of this method calculating the spectral density of a signal. For performing test of additive algorithms, we had to simulate the work analog to digital converter. This converter instead to give on the output a value of the original function, produces integration of the original function by the independent variable, and then compares the integral with a threshold value set a priori and when the value of the integral becomes equal to the threshold, gives value of the independent variable at this point in the form of digital code. Computer simulation have been performed with the following conditions: the interval of investigation of a real function

and the interval of the spectral density are equal to  $2\pi$ . The text of the main part of the program is very simple.

**Input:**

```
interval_T - interval of a time analysis
lambdaF    - threshold
nr         - number of intervals of integration
function(time_tq)- original function
```

**Output:**

```
AAk(omega_k) - value of the spectral density in a
               point omega_k
while time_tq <= interval_T
nr++
AAk(omega_k) = AAk(omega_k)+
               cos(omega_k,tableTime[nr])*lambdaF
end while
return AAk(omega_k) = AAk(omega_k)/M_PI
```

Simulation of the work of the analog-digital converter is performed in accordance with the following fragment program's, that is given below. The result of the work of this fragment program's is obtaining of the table of values of the independent variable, where integral of the original function is equal to the threshold value. This table labeled as `tableTime[]` in the main fragment of the program (see up). As mentioned above, this work is carried out only once, and then the results are used repeatedly to compute all values of the spectral density.

**Input:**

```
time_tq    - value of time in a point q
dtime_tq   - elementary interval of a time
interval_T - interval of a time analysis
lambdaF    - threshold
intl       - value integral on elementary interval
intlm      - module of value integral
nr         - number of intervals of integration
function(time_tq)- original function
```

**Output:**

```
tableTime[nr]

while time_tq <= interval_T
```

```

while intlm <= lambdaF
  intl = intl + function(time_tq)*dtime_tq
  intlm = intl > 0 ? intl: - intl
  time_tq = time_tq + dtime_tq
end while
intlm = 0
nr++
tableTime[nr] = time_tq
end while

```

To estimate of the result of modeling we will compare it with the result of the integration (1) using Euler numerical algorithm of integration. The text of the main part of the program is represented below.

**Input:**

```

omega      - value of the frequency original
            function
omega_k    - value of the frequency in a point k
timeTk    - value of time in a point k
dtimeTk   - elementary interval of a time
interval_T - interval of a time analysis
nk        - number of intervals of integration
function(omega,timeTk)- original function

```

**Output:**

```

Ak - value of the spectral density in a point
    omega_k

```

```

while time_tq <= interval_T
  Ak = suma_Ak + function(timeTk)*
                cos(omega_k,timeTq)*dtimeTk
  timeTk=timeTk+deltaTk
  nk++
end while
return Ak = Ak/M_PI

```

We performed experiments when original signal it is:

$$f(t) = \sum_{s=1}^6 \cos(s * \omega * time), \quad time \in [0, interval\_T], \quad (12)$$

that is represented on Fig. 1. This figure shows the values of the original function, in which the definite integral of a function on the elementary integration interval becomes equal to the threshold value. On the graph these values of the function denoted as black points.

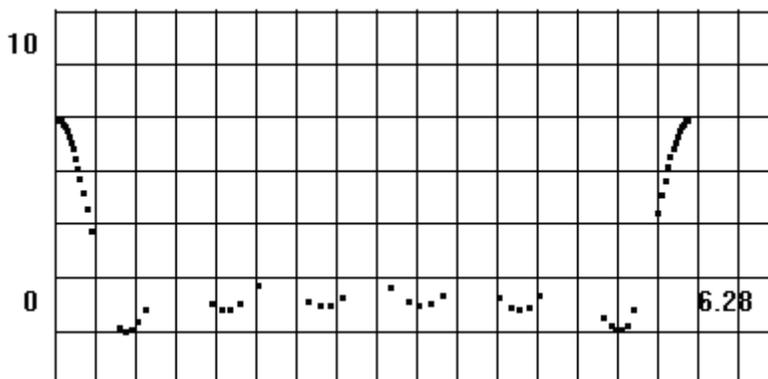


Fig. 1. Original signal with set of points, where are taken the values of the basic functions

Spectral analysis was performed by means of additive algorithm with such parameters:  $interval\_Q \in [0, 6.28]$ ,  $lambdaF=0.1$ ,  $dtime\_tq=0.01$ ,  $nr=58$ , at that the number of intervals of integration is dependent from original function. The result of the experiment is shown on Fig. 2.

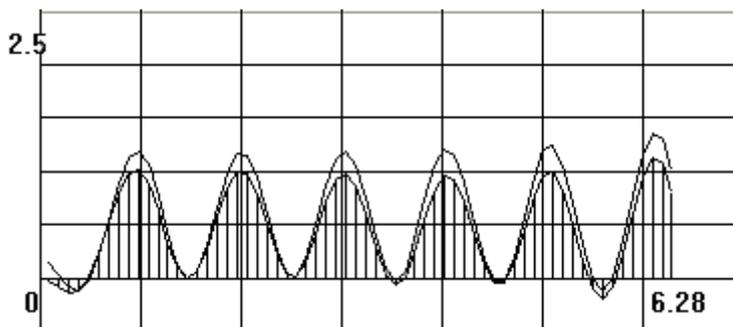


Fig. 2. Spectrum of the original signal, was calculated by means of additive algorithm and Euler algorithm

Graph showing the results of analysis performed by means of additive algorithm is shown as a shaded surface and a graph showing the results of the algorithm is shown in curve Euler without hatching. Spectral analysis was performed by means of Euler algorithm with such parameters:  $interval\_Q \in [0, 6.28]$ ,  $dtime\_tq=0.098$ ,  $nk=64$ . The simulation results are also presented in Table 1. In the table represented value of the

spectrum, received by means of additive and Euler algorithms, for original function, graph of which shown above, Fig. 1.

Table 1. Comparison of the additive algorithm and Euler algorithm

Algorithm	1*omega	2*omega	3*omega	4*omega	5*omega	6*omega
SOA	1.022	1.029	1.04	1.019	1.007	0.969
Euler	1.187	1.187	1.187	1.187	1.187	1.187

We see that errors that is received as a result of the simulation with applying additive algorithm less errors that were received as a result of the simulation with applying Euler algorithm. Important, that such result was received, when number of points in grid of values of the independent variable additive algorithm was less than in Euler algorithm.

Now consider the application of additive algorithm for to calculate DCT conversion, which is widely used for image compression. It is known, that a DCT is a Fourier-related transform. In general, there are eight standard DCT variants, of which four are often used. Let us consider development DST-II conversion which the most commonly used in modern technologies of compression of visual and audio images, such as JPEG and MPEG [5]. DCT, as cosine transform, assumes an even extension of the original function. In DCT the  $N$  real numbers  $x_0, \dots, x_{N-1}$  are transformed into the  $N$  real numbers  $X_0, \dots, X_{N-1}$  according to the formulas:

$$G(0) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} f(n), \tag{13}$$

$$G(k) = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} f(n) \cos \frac{\pi k(2n+1)}{2N}, \quad k = 1, 2, \dots, N-1. \tag{14}$$

Thus DCT is used to convert  $f(0), \dots, f(N-1)$  real numbers in a series of real numbers  $G(0), \dots, G(N-1)$ , where  $G(0)$  is the average value of the sample sequence. According to (9) transform the formula (14) as follows

$$G(k) = \sqrt{\frac{2}{N}} \sum_{u=0}^{N-1} \lambda_u^f(k) \cos \frac{\pi k(2u+1)}{2N}, \quad k = 1, 2, \dots, N-1. \tag{15}$$

Suppose that at our disposal, there is only a table of values of the original function, which is represented with a uniform step in the independent variable. For testing of the additive algorithm as original function was selected the result of the scan real

image. For example, take the result of scanning of a monochromatic color component of the image represented in the system RGB at 24 bits. In this case, each color is represented by 8 bits, therefore brightness of the image points lie in the range 0–255. On Fig. 3 is shown the testing function and with offset by 200 units below DCT conversion of the testing function performed by traditional algorithm. The main fragment of the program for DCT calculation is represented below.

**Input:**

N - number of values of the spectrum  
 n - the current number of values of the original function  
 k - the current number of values of the DCT function(n)-value of the original function in point n

**Output:**

DCT(k) - value of the spectral density in the point k

**for** (n=0; n<=N-1; n++)

DCT(k) = DCT(k) +  
           function(n)\*cos(M\_PI\*k\*(2\*n+1)/(2\*N))

**end for**

**return**

DCT(k) = DCT(k)\*sqrt(2/N)

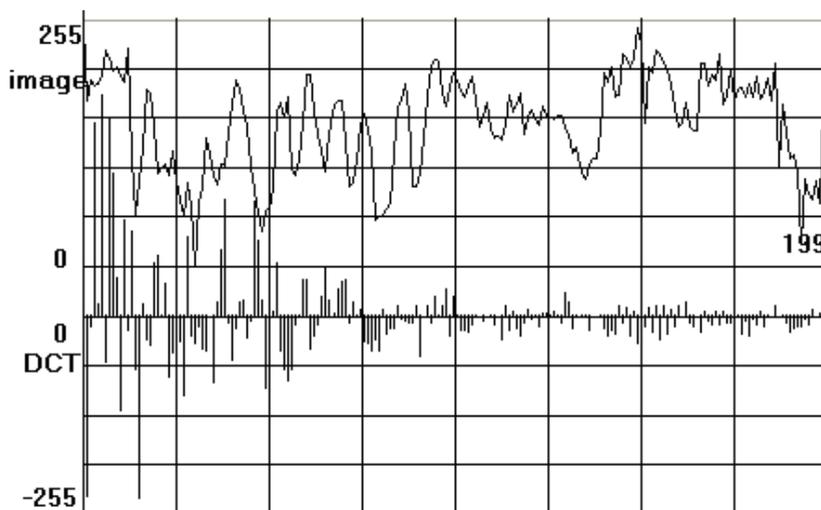


Fig. 3. Original signal and spectrum of the original signal, performed by DCT algorithm

In accordance with the idea of additive algorithm a grid of values of the independent variable, in nodes of which should be chosen values of the basic functions is irregular and is determined in the process of integration of the original function. For the simulation of analog digital convertor in this case, we have to perform a linear interpolation in the intervals between the values of the original function, defined in the table of its values. After interpolation, it is necessary to carry out the integration, the resulting function so as to obtain a table of values of the independent variable (`tableTime[nr]`), just as was done earlier in the program for simulation of analog digital convertor. The text of the main part of the program for modeling of the work of additive algorithm at calculation ADCT is represented below.

**Input:**

```

lambdaF    - threshold
N          - number of values of the spectrum and
            number of values of the original function
Nd         - number of intervals of integration
n          - the current number of values of the original
            function
k          - the current number of values of the ADCT
nr         - number of interval of integration

```

**Output:**

```

ADCT(k)    - value of the spectral density in a point k

```

```

for (n=0; n<=Nd; n++)
ADCT(k) = ADCT(k) +
            cos(M_PI*k*(2*tableTime[nr]+1)/(2*N))
end for

```

**return**

```

ADCT(k) = ADCT(k)*lambdaF*sqrt(2/N)

```

In accordance with the above fragment of the program experiment have been performed. The experimental conditions were as follows:  $N \in [0, 199]$ , `tableTime[]`, was defined with parameters `dttime_tq = 0.098` and different value of `Nd` and `lambdaF`. The simulation results are presented in Fig. 4, on which is shown the graph of the ADCT and graphs of errors, that are displaced downwards along the ordinate on 100 and 200 units respectively. On graph of errors, displaced downwards along the ordinate on 100 units, is represented the experiment, when `lambdaF` is equal 132 units. The number of the points, in which we are needed to take the value of the basic function, equal 221. That is in the last fragment of the program `Nd = 221`. On graph of

errors, displaced downwards along the ordinate on 200 units, is represented the experiment, when  $\lambda F$  is equal 73 units. The number of the points, in which we are needed to take the value of the basic function, equal 397. That is in the last fragment of the program  $N_d = 397$ .

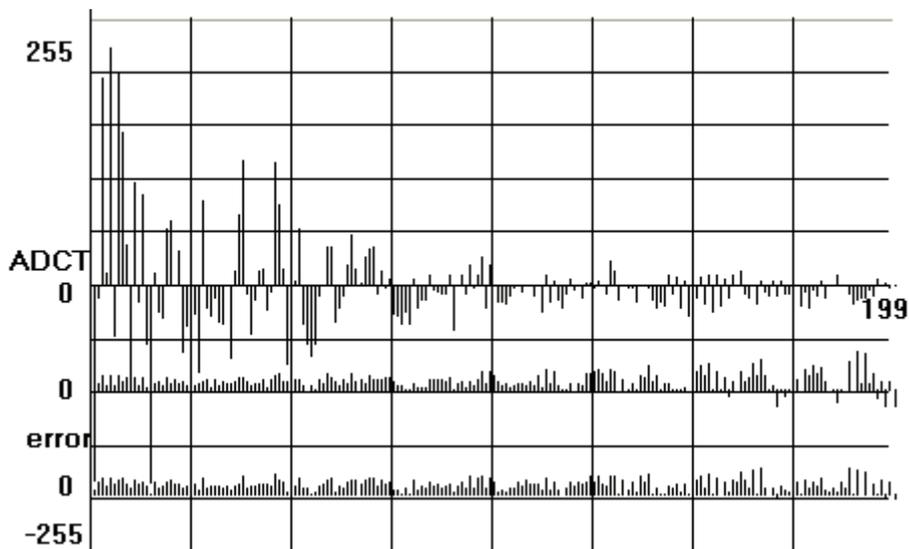


Fig. 4. Spectrum of the original signal, performed by additive algorithm, and graphs of the errors

In addition in the experiment was conducted calculating of the average reducial error according to formula

$$errorAv = \frac{1}{N * \max DCT} \sum_n^N [DCT(n) - ADCT(n)], \quad (16)$$

and is represented average reducial error modulo according to formula

$$errorAvm = \frac{1}{N * \max DCT} \sum_n^N \text{mod}[DCT(n) - ADCT(n)]. \quad (17)$$

Table 2. Comparison errors of the additive algorithm

lambdaF	Nd	errorAv	errorAvm
64	452	0.038	0.040
73	397	0.036	0.041
96	303	0.034	0.041
132	221	0.027	0.064

The results of the experiment are shown in Table 2. Table provides information as about average error as and about average error obtained by addition modules of the errors on the analysis interval, where  $maxDCT = 244$ . It is interesting to note that the reducial average error in a certain range is only slightly dependent on the number of samples of basic functions  $N_d$ . On the other hand reducial average modules errors increases rapidly when the number of samples approaches  $N$ , to number of samples of the original signal in DCT algorithm. Generally practical equality of the reducial average error to reducial average module errors suggests that there is a systematic displacement of results towards smaller values. This phenomenon can be seen in Figure 4, where were shown graphic representation of errors. This trend makes sense to explore.

#### 4. CONCLUSIONS

The positive feature of the proposed method is that it can be applied in every integral transform of the signal with different basic functions. For this the original signal have to be converted into a sequence of values of the independent variable, in every of where value of the definite integral from original function are equal to the value of the threshold as described above. Application of the proposed algorithms in practice makes sense if is opportunity to have the values of basic functions in points irregular grid values of the independent variable studied function.

#### REFERENCES

- [1] BORKOVSKY B.A., KATKOW A.F., ROMANTHOV V.P., *Additive Formulas Discrete Fourier Transformation*, In: Mathematical Modeling and Theory of Electrical Circuit, Puchov G.E. (Ed.), Kiev, Naukova Dumka, No. 14, 1976, 80–84.
- [2] KATKOW A.F., *Dual Formulas Additive Fourier Transformation*. In: Mathematical Modeling and Theory of Electrical Circuit, Puchov G.E. (Ed.), Naukova Dumka, Kiev, No. 16, 1978, 53–59.
- [3] KATKOV A., WEGRZYN-SKRZYPCZAK E., *Speedy Numerical Algorithms and Architecture for Additive Spectral Analysis*, Journal of Mathematical Modeling and Algorithms, **V.1**, No.3, 2002, 225–241.
- [4] KATKOW A., *Additive Algorithms of Spectral Analysis of Signals in Real Time*, In book: An Introductory Guide to Image and Video Processing, iConcept Press Ltd., USA, 2013, 251–269.
- [5] SHAO X., JOHNSON S. G., *Type-IV DCT, DST, and MDCT algorithms with reduced numbers of arithmetic operations*, Signal Processing, Vol. 88, No. 6, 2008, 1553–1564.

Jarosław KOSZELA, Karol WÓJCIK, Roman WANTOCH-REKOWSKI\*

## **GESTURE RECOGNITION INTERFACE ARCHITECTURE FOR VIRTUAL SIMULATION**

One of the methods of human-computer communication during training using virtual simulators is the ability to detect people's gestures. The development of an interface between the device designed to detect gestures and the virtual simulation environment allows the introduction of direct interaction between real people and simulation objects. The gestures made by a real person can be used to issue commands to other people (avatars) operating in the virtual simulation environment. This article presents an exemplary application of the Kinect device in conjunction with a virtual simulation environment VSB2 (Virtual Battlespace 2). The proposed mechanism is part of a wider interface allowing advanced human-computer interaction.

The paper presents the interface architecture designed by the authors of this article, as well as an example of its implementation in a virtual VBS2 simulation environment.

### 1. INTRODUCTION

One of the uses of virtual simulators is enabling carrying out communication between the real world and simulated objects. This need stems from the necessity of interacting in the field of training using simulators [7]–[9]. There are many different ways to implement such communications. One method is to detect gestures of real people and transferring them to the virtual world. The presented solution is one of the components of the “immersive”-type virtual simulation, which ensures greater realism during training, and increases the range of possible to use functionalities associated with the human-computer interaction.

The architecture is presented in this paper, which enables the connection of support devices detecting gestures with a programmable virtual simulation environment VBS2 [1]. The Microsoft Kinect device was used in the proposed solution for detect-

---

\* Military University of Technology, Faculty of Cybernetics, gen. S. Kaliskiego Str. 2, 00-908 Warsaw.

ing gestures. A method of combining the gesture detecting module implemented in C language with the virtual simulation environment in the form of responses of simulation objects on selected gestures of a real person.

In the next few chapters properties of the used Kinect device are presented, as well as the designed architecture of the gesture recognition interface, basic software components of the implemented interface, the method of communication between the software running on the Kinect device and the simulation software. Finally, the issues on predicting states of objects in virtual simulation will be presented as well as a summary of the entire paper.

## 2. BASIC PROPERTIES OF THE KINECT DEVICE

The Kinect device was originally designed only for the Xbox 360 as a motion sensor input device. The Kinect device interface uses gestures of the entire human body to communicate with the program. The device is currently available on PC, and thanks to Microsoft releasing the SDK package using C++ or C #, you can design your own software on this platform.

The device consists of two cameras, an infrared emitter, a set of four directional speakers, an accelerometer, and a head position control motor. One of the cameras is a simple RGB camera recording at a resolution of 640x480. It is used inter alia for applying texture to virtual objects. The second camera is the so-called “depth camera”, which in cooperation with the infrared emitter forms an image depth map. This allows the user gesture recognition to be performed in three-dimensional space. The infrared emitter displays a series of points in front of the camera that are thrown onto the player and his/her environment, the depth camera is equipped with an infrared filter, so that it “sees” these points, a map creates them and applies the image from the first camera. This way, the device distinguishes which of the points in the image is closer, and which further enables three-dimensional interpretation of the recorded space. Kinect allows tracking two active players simultaneously by dividing their body outline into forty-eight sectors (the so-called joints), which are tracked throughout the operation of the program.

## 3. THE ARCHITECTURE OF THE SOLUTION

The architecture of the gesture recognition subsystem, using the Kinect device in the VBS2 simulation environment, includes the following components:

- *VBS2MainModule.exe* – Main Module of the system.
- *VBSPlugin.dll* – VBS2 system *plug-in*.
- *KinectVBS2.exe* – Microsoft Kinect service module.

- *ScriptSender.exe* – SQF script console module.
- *Connector.exe* – connector module.

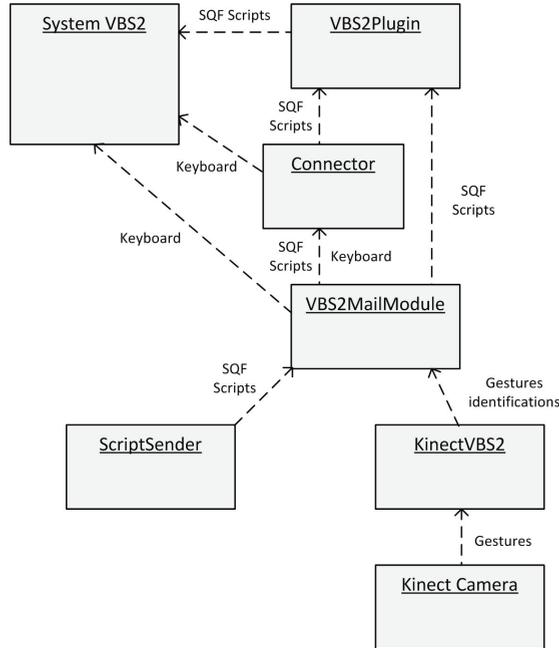


Fig. 1. The architecture of the interface

The *main module* should be a window application that runs on Windows. Its task will be to establish a connection with the *plug-in* and all the external controllers. For each external controller the module should allow the determination of what kind of control signals will be received and a specific action for each of them should be assigned within the VBS2 simulation. These actions will be: simulating the pressing of any key on the keyboard, performing one of the predefined scripts, performing one of the user-written scripts. The module should contain a fully editable library of ready scripts in SQF [1], which stores basic commands that a user might want to use during training. External controllers such as Microsoft Kinect will not have to be physically connected to the computer, on which the module will be started; the communication between modules was designed to operate via LAN. The module allows simultaneous communication with a number of controllers of the same type, and each of them can be assigned an individual dictionary of events.

The purpose of the *plug-in* is to receive signals from the scripts sent by the *Main Module* and executing them in the same order as they were sent. In the first version of the system one-way communication was assumed, that is, the *plug-in* should only

receive information, the implementation of two-way communication is not ruled out in the future, so that, for example, a player possessing a phone on him/her was informed through vibrations that he was hit. In the event of an unforeseen connection interruption with the *Main Module*, the *plug-in* should perform all previously received scripts, and then wait for reconnection. Regardless of the fact whether the *Main Module* is active and connected with the *plug-in*, the *plug-in* should allow standard user unit control using the mouse and keyboard

The application should detect the Kinect connected to the computer, check and report its status (whether it is turned on and ready to work) and connect to the *Main Module*. The user should be able to configure the address of the computer on which the *Main Module* is operating and the port to which the *Main Module* receives the transmitted data. By using Kinect it should be possible to track the movement of the user, to detect pre-defined static and dynamic gestures. Information on the detected gesture should be transferred to the *Main Module*, only if this gesture will be done entirely and correctly. The sensor module includes its own dictionary clearly explaining gestures that are to be detected. After detecting a gesture its identification information is sent to the *Main Module*, which based on its dictionary “gesture id – script” transfers the appropriate script for execution to the *plug-in*.

#### 4. COMPONENTS OF THE MAIN MODULE

Functions and objects required to implement:

**Server threads** – application for each connection with an external controller creates a separate, dedicated thread. This thread is responsible for establishing and maintaining a connection and exchange data with the external controller.

**Script and key queues** – each control signal will result in sending a command to be performed of a specific script to the *plug-in* or simulating a keyboard key being pressed. Scripts and keys are received from all threads and collected into two global FIFO queues, one for scripts and one for keys. Because queues are on the main application thread, a direct reference to them from many other threads will be impossible, therefore it is necessary to implement an asynchronous mechanism for inserting elements into queues.

**Communication thread with the VBS2 plug-in** – communication with the VBS2 system will also be done in a separate thread that will collect data from the queue of scripts and keys and send them directly to the *plug-in*.

**Dictionary parsing mechanism** – delivered in the form of an XML file dictionaries (e.g. Microsoft Kinect module) will have to be parsed to objects on the basis of which the *Main Module* will be mapping out the resulting identifier for the script or the key.

**The mechanism of “capturing” the VBS2 process** – to simulate the pressing of a key the VBS2 process needs to be “captured” and be told to listen to not only real keyboard keystrokes, but also those simulated by the *Main Module*.

**Keystroke simulation mechanism** – a feature that will act as a virtual keyboard for the main module. If a control signal will require simulation of pressing any key on the keyboard, the Main Module will transmit information to the VBS2 simulation on the key that was pressed, so that the simulator can perform the action assigned to the key.

**Communication** – communication between modules can be based on the mechanism of named links or sockets.

## 5. MICROSOFT KINECT MODULE

This module with the help of the Microsoft Kinect camera recognizes gestures performed by the user and sends the identifier assigned to them to the *Main Module*. An action will be performed based on the dictionary “action identifier” in the *Main Module*, i.e. simulating the pressing of a key or sending the SQF script for simulation.

The “*Simple Gesture Processing using the Kinect for Windows*” project was used for recognizing gestures. The main element of the project is the *Gesture Framework* – the framework provides the functionality of tracking user profiles and recognizing predefined gestures made by him/her. The framework enables the recognition of dynamic gestures.

## 6. COMMUNICATION MECHANISM BETWEEN MODULES BASED ON “NAMED PIPES”

Communication in the form of the “named pipes” mechanism was used in the presented solution. Named pipes is a mechanism of sharing memory allowing processes to communicate with each other. Any process that knows the name of the named link can access it. Implementation of the process in various technologies (C++/C#) does not impact the possibility or the quality of communication. A process that creates a link is called a server, and the process that requests access to the link is the client. In the case of the created module the server is the *main module*, and the clients the external controller modules and VBS2 *plug-in*.

Links for both servers and clients are created in separate threads, and only for the requested external controllers. Character tables are collected from key and script queues and transmitted through links.

The processing algorithm for the server thread (see Fig. 2):

1. Create a link with a specific name.
2. Await client connection.
3. If the client connects, then until the thread does not end or no communication error occurs:
  - 3.1. Check, if there are any character tables in the queue to be sent.
  - 3.2. If so:
    - 3.2.1. Remove one element from the queue and send it via the link.
    - 3.2.2. Return to point 3.1.
  - 3.3. If not, return to point 3.1.

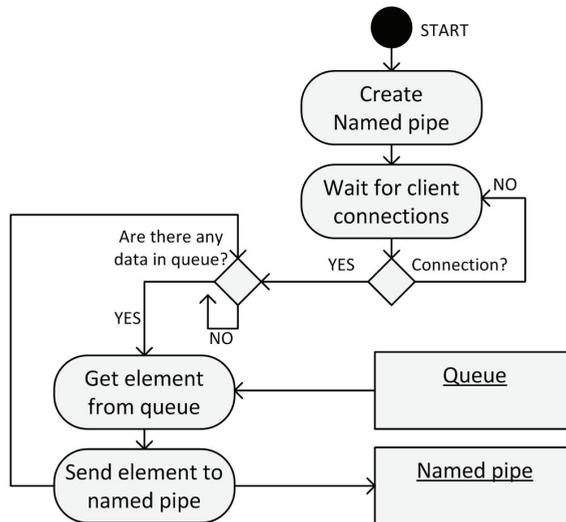


Fig. 2. Server thread algorithm

The processing algorithm for the client thread (see Fig. 3):

1. Create a link with a specific name.
2. Connect to link.
3. Until the thread does not end or no communication error occurs:
  - 3.1. Remove character tables from the link.
  - 3.2. If the table length is greater than zero then place it in an appropriate queue.
  - 3.3. Return to point 3.1.

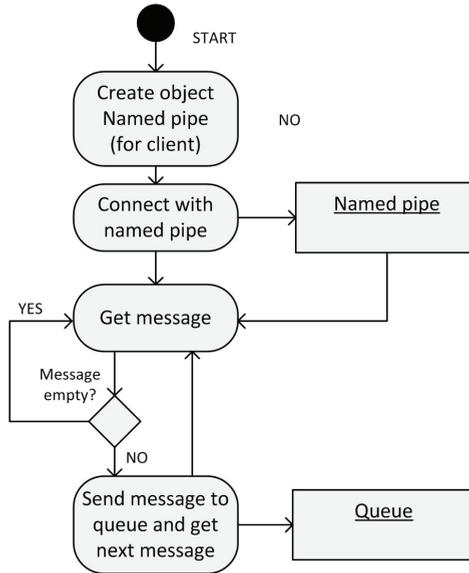


Fig. 3. Client thread algorithm

## 7. GESTURE FRAMEWORK

To implement the presented solution a *framework* was used, which is based on the observation of changes in coordinates of specific parts of the body in accordance with each other, at specified intervals. Two body parts should be given within the definition of the expected gesture in the configuration file, the relation they are to be placed in at the start of the gesture, the relation ending the gesture, and the maximum time executing the gesture. Allowed relations include:

- *None*
- *Above*
- *Below*
- *LeftOf*
- *RightOf*
- *AboveAndRight*
- *BelowAndRight*
- *AboveAndLeft*
- *BelowAndLeft*

The engine of the *framework* after reading the configuration file analyzes the data sent by Kinect “searching” for defined gestures.

The main classes of the framework:

*GestureComponents* – it includes a description of a single relation between two body parts. The description consists of two gesture stages: start and end position of body parts in relation to each other.

*Gesture* – *GestureComponents* type list consisting of a unique identifier as well as description.

*GestureMap* – a set of features that allow reading and operating gestures described in the configuration file.

*GesureComponentState* – enables tracking the state of individual gestures described in *GestureComponents*.

*GestureState* – manages the state of the carried out gesture and reports if one of the awaited gestures was carried out.

*GestureMapState* – creates and tracks the map of all started gestures, the status is continuously updated.

## 8. PREDICTION MECHANISM

The mechanism of prediction in virtual simulation can also be used to compensate for delays resulting from the applied system and hardware architecture of the simulation environment. The solution is of particular importance in case the simulation concerns highly dynamic processes of change and very short times of their operation such as rapid movement of an object, bullet shot, etc.. Currently, the most commonly used virtual simulation system architecture makes use of simulations with a stepping time (1 step =  $\sim 10\text{--}20$  ms). Such a solution means that in step  $x$  the simulator processes events taking place in the virtual world (e.g. location of objects and their interactions with other objects, phenomena of physics and behavior of objects, etc.). Then, the effects of this step are transferred to the other computer components (e.g. graphics module, graphic card, and monitor) for further processing, which in effect allows for an interaction with the user. In the case of rapidly occurring processes it means that what the user sees on the screen is delayed by  $d1 \geq 1$  steps, i.e.  $x - d1$  steps in relation to what the processor is currently processing in the virtual simulator memory. The user reaction to the presented situation makes it concern the visualized step  $x$ , while the CPU processes the step of at least  $x + d2$ , where  $d2 \geq 1$ . In addition, communication signals through technical devices from the user to the simulator are also delayed, in that after it reaches the virtual simulator it is not the actual time in relation to the step currently being processed e.g.,  $x + d3$ , where  $d3 \geq 1$ . As long as technical solutions will contribute to such high delays in data processing and transmission in relation to the phenomena taking place, then one of the methods that allow minimization is by using of prediction mechanisms. These mechanisms should be used in relation to the

most important processes of interaction between the simulator and the user, which delays negatively affect the efficient use of “immersive”-type virtual simulation.

In the first step the total delay time should be estimated that is caused in the simulator system by software and hardware components moving from simulator to user and from user to simulator. Then, for the selected events determine their adjustments resulting from the estimated delay times and applying it according to the technical possibilities or in the simulation environment or on an external device or its software.

## 9. CONCLUSION

A designed and implemented a mechanism for recognizing gestures is a universal and configurable tool for human-computer communication. The results of the carried out tests of recognizing different gestures and their transfer to the virtual world prove its usefulness, because they can reflect even more complex behavior of real people.

The use of the properties of an advanced gesture recognition device, the kind that Kinect is, enabled the introduction of a very intuitive communication between man and objects operating in a virtual simulation. The proposed mechanism of the gesture recognition of real people is a very natural way of communication and may become the base of different types of exercises and computer-aided training using virtual simulators.

## ACKNOWLEDGMENTS

This work was partially supported by the National Centre for Research and Development: ROB 0021/01/ID21/2 and UOD-DEM-1-501/001.

## REFERENCES

- [1] WANTOCH-REKOWSKI R. (scientific editor), *The programmable VBS2 virtual simulation environment*, 2013 PWN, ISBN 978-83-01-17323-4.
- [2] KOSZELA J., WANTOCH-REKOWSKI R., *The concept of a simulator for training drivers of PSP combat vehicles in terms of tasks carried out in the framework of the national rescue and fire-fightingsystem*, Safety & Fire Technique, pp.: 71–81, ISSN: 1895-8443, 2012.
- [3] KOSZELA J., DROZDOWSKI T., WANTOCH-REKOWSKI R., *Preparing field data for multiresolution simulation*, High-speed Track Vehicles (31) No. 3, 2012, pp.: 109–118, ISSN: 0860-8369.
- [4] KOSZELA J., DROZDOWSKI T., WANTOCH-REKOWSKI R., *Advanced methods of preparing field data for the VBS2 simulator tactical level*, Institute of Computer Science, Faculty of Cybernetics, Military University of Technology, Warsaw, (11) 2013, 1508–4183.
- [5] KOSZELA J., WRÓBLEWSKI P., SZYMAŃSKA A., WANTOCH-REKOWSKI R., *Design and implementation of mechanisms of artificial intelligence in the VBS2 simulation environment*, SPG (31), No. 3, 2012, pp.: 119–132, ISSN: 0860-8369.

- [6] KOSZELA J., WANTOCH-REKOWSKI R., *The use of simulators for training in crisis situations*, Safety & Fire Technique, 2013, pp.: 113–120, ISSN: 1895-8443.
- [7] ANTKIEWICZ R., KULAS W., NAJGEBAUER A., PIERZCHAŁA D., RULKA J., TARAPATA Z., WANTOCH-REKOWSKI R., *Modelling and simulation of C2 processes based on cases in the operational simulation system for CAX*, 1<sup>st</sup> Military Communication and Information System Conference MCC'2006, Gdynia, 2006, ISBN 83-920120-1-1.
- [8] ANTKIEWICZ R., KULAS W., NAJGEBAUER A., PIERZCHAŁA D., RULKA J., TARAPATA Z., WANTOCH-REKOWSKI R., *Some aspects of designing and using deterministic and stochastic simulators for military trainings and CAX'es*, Proceedings of the Military Communications and Information Systems Conference MCC'2008, ISBN 83-920120-5-4, 23–24 September 2008, Cracow, Poland.
- [9] ANTKIEWICZ R., NAJGEBAUER A., KULAS W., PIERZCHAŁA D., RULKA J., TARAPATA Z., WANTOCH-REKOWSKI R., *The Automation of Combat Decision Processes in the Simulation Based Operational Training Support System*, Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA) 2007, ISBN 1-4244-0698-6, Honolulu (Hawaii, USA), April 1–5, 2007.

Zbigniew BUCHALSKI\*

## **AN HEURISTIC SOLUTION PROCEDURE FOR SOLVING THE PROGRAMS SCHEDULING PROBLEM IN MULTIPROCESSING COMPUTER SYSTEM**

The paper presents results of research on the problem of time-optimal programs scheduling and primary memory pages allocation in multiprocessing computer system. We consider an multiprocessing computer system consisting of  $m$  parallel processors, common primary memory and external memory. The primary memory contains  $N$  pages of identical capacity. This system can execute  $n$  independent programs. Because our problem belongs to the class of  $NP$ -complete problems we propose an heuristic algorithm to minimize schedule length criterion, which employs some problem properties. Some results of executed computational experiments for basis of this heuristic solution procedure are presented.

### 1. INTRODUCTION

Scheduling problems can be understood very broadly as the problem of the allocation of resources over time to perform a set of tasks. By resources we understand arbitrary means tasks compete for. They can be of a very different nature, eg. energy, tools, money, manpower. Tasks can have a variety of interpretation starting from machining parts in manufacturing systems up to processing information in computer systems.

A structure of a set of tasks, reflecting precedence constraints among them, can be defined in different ways. Different criteria which measure the quality of the performance of a set of tasks can be taken into account [4, 6, 9–20]. The time-optimal problem of tasks scheduling and resources allocation are intensive developing, as in [9, 10, 18, 19]. The further development of the research has been connected with applications, among

---

\* Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wrocław, Poland, e-mail: zbigniew.buchalski@pwr.wroc.pl

other things, in many production processes and in multiprocessing computer systems, as in [1, 2, 3, 5, 7, 8, 21].

In multiprocessing computer systems is usually used common primary memory with limited capacity and external memory. The external memory has significantly longer access time and this is why minimization of the number of demands to the external memory during programs processing is necessary.

In this paper the problem optimization of programs scheduling and optimal allocation of primary memory pages to the processors are considered. These programs scheduling and primary memory pages allocation problems are very complicated problems and belongs to the class of  $NP$ -complete problems. Therefore in this paper we propose an heuristic algorithm for solving of a optimization problem. In the second section formulation of optimization problem is presented. In the third section an heuristic algorithm is given and in the fourth section several experimental results on the base this heuristic algorithm are presented. Last section contains final remarks.

## 2. FORMULATION OF THE PROBLEM

We consider an multiprocessing computer system (as shown in Fig.1) containing  $m$  processors, common primary memory and external memory. This system can execute  $n$  independent programs.

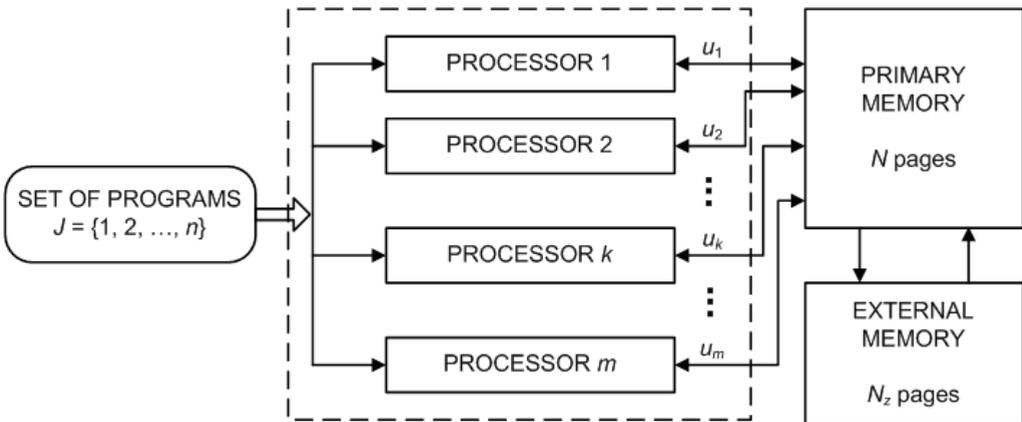


Fig.1. Multiprocessing computer system

This system can execute  $n$  independent programs. We assume about this system, that it is paged virtual memory system and that:

- the primary memory contains  $N$  pages of identical capacity,

- each the processor has access to every one of  $N$  primary memory pages and may execute every one of  $n$  programs,
- the external memory contains  $N_z$  pages (the external memory pages capacity is equal to the primary memory pages capacity),  $N_z > N$ ,
- during execution of all  $n$  programs, the number of  $u_k$  primary memory pages is allocated to the  $k$ -th processor;  $\sum_{k=1}^m u_k \leq N$ . Each processor may use only allocated to him the primary memory pages.

Let  $J = \{1, 2, \dots, n\}$  be the set of programs,  $U = \{1, 2, \dots, N\}$  – set of primary memory pages,  $P$  denotes the set of processors  $P = \{1, 2, \dots, m\}$ . Processing time of  $i$ -th program on  $k$ -th processor is given by following function:

$$T_i(u_k, k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in U, \quad 1 \leq k \leq m, \quad i \in J, \quad (1)$$

where  $a_{ik} > 0, b_{ik} > 0$  – parameters characterized  $i$ -th program and  $k$ -th processor.

This programs scheduling and primary memory pages allocation problem in multi processing computer system can be formulated as follows: find scheduling of  $n$  independent programs on the  $m$  processors running parallel and partitioning of  $N$  primary memory pages among  $m$  processors, that schedule length criterion is minimized.

Let  $J_1, J_2, \dots, J_k, \dots, J_m$  be defined as subsets of programs, which are processing on the processors  $1, 2, \dots, k, \dots, m$ . The problem is to find such subsets  $J_1, J_2, \dots, J_k, \dots, J_m$  and such pages numbers  $u_1, u_2, \dots, u_k, \dots, u_m$ , which minimize the  $T_{opt}$  of all set  $J$ :

$$T_{opt} = \min_{\substack{J_1, J_2, \dots, J_m \\ u_1, u_2, \dots, u_m}} \max_{1 \leq k \leq m} \left\{ \sum_{i \in J_k} T_i(u_k, k) \right\} \quad (2)$$

under the following assumptions:

- (i)  $J_s \cap J_t = \emptyset, \quad s, t = 1, 2, \dots, m, \quad s \neq t, \quad \bigcup_{k=1}^m J_k = J,$
- (ii)  $\sum_{k=1}^m u_k \leq N, \quad u_k \in U, \quad k = 1, 2, \dots, m,$
- (iii)  $u_1, u_2, \dots, u_m$  – positive integer.

The assumption (iii) is causing, that the stated problem is very complicated therefore to simplify the solution our problem we assume in the sequel that primary memory pages are continuous. The numbers of pages obtained by this approach are rounded to the integer numbers (look **Step 13** in the heuristic algorithm) and finally our problem can formulated as following minimizing problem:

$$T_{opt} = \min_{\substack{J_1, J_2, \dots, J_m \\ u_1, u_2, \dots, u_m}} \max_{1 \leq k \leq m} \left\{ \sum_{i \in J_k} \tilde{T}_i(u_k, k) \right\} \quad (3)$$

under the following assumptions:

$$(i) \quad J_s \cap J_t = \emptyset, \quad s, t = 1, 2, \dots, m, \quad s \neq t, \quad \bigcup_{k=1}^m J_k = J,$$

$$(ii) \quad \sum_{k=1}^m u_k \leq N, \quad u_k \geq 0, \quad k = 1, 2, \dots, m,$$

where  $\tilde{T}_i : [0, N] \times \{1, 2, \dots, m\} \rightarrow R^+$  is the extension of function  $T_i : \{1, 2, \dots, N\} \times \{1, 2, \dots, m\} \rightarrow R^+$  and formulated by function:

$$\tilde{T}_i(u_k, k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in [0, N], \quad 1 \leq k \leq m, \quad i \in J. \quad (4)$$

Taking into account properties of the function  $\tilde{T}_i(u_k, k)$ , it is easy to show the truth of the following theorem:

### Theorem 1

If the sets  $u_k^*, J_k^*, k = 1, 2, \dots, m$  are a solutions of minimizing problem (3), then:

$$(i) \quad \sum_{k=1}^m u_k^* = N; \quad u_k^* > 0, \quad k : J_k^* \neq \emptyset, \quad k = 1, 2, \dots, m;$$

$$u_k^* = 0, \quad k : J_k^* = \emptyset, \quad k = 1, 2, \dots, m;$$

$$(ii) \quad \sum_{i \in J_k^*} \tilde{T}_i(u_k^*, k) = \text{const}, \quad k : J_k^* \neq \emptyset, \quad k = 1, 2, \dots, m.$$

Condition (i) of **Theorem 1** shows, that all the primary memory pages are used in the optimal scheduling and condition (ii) shows, that processing time of each processor is the same. Proof of **Theorem 1** can be found in [8].

We define function  $F(J_1, J_2, \dots, J_m)$ , which value is solution following system of equations:

$$\left\{ \begin{array}{l} \sum_{i \in J_k} a_{ik} + \frac{\sum_{i \in J_k} b_{ik}}{u_k} = F(J_1, J_2, \dots, J_m), \quad k : J_k \neq \emptyset, \quad k = 1, 2, \dots, m \\ \sum_{k: J_k \neq \emptyset} u_k = N; \quad u_k > 0 \quad k : J_k \neq \emptyset, \quad k = 1, 2, \dots, m \end{array} \right. \quad (5)$$

On the basis of **Theorem 1** and (5), problem (3) will be following:

$$T_{opt} = \min_{J_1, J_2, \dots, J_m} F(J_1, J_2, \dots, J_m) \quad (6)$$

under the assumptions:

- (i)  $J_s \cap J_t = \emptyset; \quad s, t = 1, 2, \dots, m, \quad s \neq t$
- (ii)  $\bigcup_{k=1}^m J_k = J; \quad k = 1, 2, \dots, m,$

If  $J_1^*, J_2^*, \dots, J_m^*$  are solutions of problem (6), it  $u_k^*, J_k^* \quad k = 1, 2, \dots, m$  are solutions of problems (3), where:

$$u_k^* = \begin{cases} \frac{\sum_{i \in J_k^*} b_{ik}}{F(J_1^*, J_2^*, \dots, J_m^*) - \sum_{i \in J_k^*} a_{ik}}; & k : J_k^* \neq \emptyset, \quad 1 \leq k \leq m, \\ 0 & ; \quad k : J_k^* = \emptyset, \quad 1 \leq k \leq m. \end{cases} \quad (7)$$

### 3. THE HEURISTIC ALGORITHM

We assume that the first processor from the set  $P$  has highest speed and the last processor from the set  $P$  has least speed. We assume also if be of assistance in pages allocation

tion so-called partition of pages coefficient  $\alpha$ ;  $\alpha > 1$ . To the last  $m$  processor is allocated  $u_m$  pages according to the following formula:

$$u_m = \frac{N}{1 + \sum_{k=1}^{m-1} [(m-k) \cdot \alpha]} \quad (8)$$

To the remaining processors are allocated pages according to the formula:

$$u_k = (m-k) \cdot \alpha \cdot u_m; \quad k=1,2,\dots,m-1. \quad (9)$$

The proposed heuristic algorithm is as follows:

- Step 1.** For given  $u_k = \frac{N}{m}$  and random generate parameters  $a_{ik}, b_{ik}$  calculate the processing times of programs  $T_i(u_k, k)$  according to the formula (1).
- Step 2.** Schedule programs from longest till shortest times  $T_i(u_k, k)$  and formulate the list  $L$  of these programs.
- Step 3.** Calculate mean processing time  $T_{mean}$  every processors according to follows formula:

$$T_{mean} = \frac{\sum_{i=1}^n T_i(u_k, k)}{m}; \quad i \in J, \quad k \in P, \quad u_k = \frac{N}{m}.$$

- Step 4.** Assign in turn longest and shortest programs from the list  $L$  to the first empty processor for the moment, when the sum of all these programs processing time assigned to that processor would not be greater than  $T_{mean}$  and eliminate these programs from the list  $L$ .
- Step 5.** If there are also empty processors, which are not scheduled with programs, go to the **Step 4**. If there is not empty processor go to the **Step 6**.
- Step 6.** Schedule in turn shortest programs from the list  $L$  to the succeeding processor from first processor to the  $m$ -th processor for the moment, when the sum of processing times these programs to keep within the bounds of time  $T_{mean}$  and eliminate these programs from the list  $L$ . If list  $L$  is not empty go to the next step, if is empty go to the **Step 8**.
- Step 7.** Remainder of programs in the list  $L$  schedule to the processors according to the algorithm  $LPT$  (Longest Processing Time) to moment, when the list  $L$  will be empty.

**Step 8.** Calculate total processing time  $T_{opt}$  of all programs for scheduling  $J_1, J_2, \dots, J_m$ , which was determined in the **Steps 3÷7** and for given numbers of pages

$$u_k = \frac{N}{m}.$$

**Step 9.** For given partition of pages coefficient  $\alpha$  allot pages  $u_k, k \in P$  to succeeding processors as calculated according formula (8) and (9).

**Step 10.** For programs scheduling which was determined in **Steps 3÷7** and for numbers of pages  $u_k, k \in P$  allotted to processors in the **Step 9** calculate total processing time  $T_{opt}$  of all programs.

**Step 11.** Repeat the **Step 9** and **Step 10** for the next nine augmentative succeeding another values of coefficient  $\alpha$ .

**Step 12.** Compare values of total processing times  $T_{opt}$  of all programs calculated after all samples with different values of coefficient  $\alpha$  (**Steps 9÷11**). Take this coefficient  $\alpha$  when total processing time  $T_{opt}$  of all programs is shortest.

**Step 13.** Find the discrete numbers  $\hat{u}_k$  of pages,  $k = 1, 2, \dots, m$  according to follows dependence:

$$\hat{u}_{\beta(k)} = \begin{cases} \left\lfloor u_{\beta(k)} \right\rfloor + 1 & ; \quad k = 1, 2, \dots, \Delta \\ \left\lfloor u_{\beta(k)} \right\rfloor & ; \quad k = \Delta + 1, \Delta + 2, \dots, m \end{cases}$$

where  $\Delta = N - \sum_{j=1}^m \left\lfloor u_j \right\rfloor$  and  $\beta$  is permutation of elements of set  $P = \{1, 2, \dots, m\}$

such, that  $u_{\beta(1)} - \left\lfloor u_{\beta(1)} \right\rfloor \geq u_{\beta(2)} - \left\lfloor u_{\beta(2)} \right\rfloor \geq \dots \geq u_{\beta(m)} - \left\lfloor u_{\beta(m)} \right\rfloor$ .

#### 4. COMPUTATIONAL EXPERIMENTS

On the base this heuristic algorithm were obtained results of computational experiments for ten another values of coefficient  $\alpha = 3, 6, 9, \dots, 30$ . For the definite number of programs  $n = 60, 120, 180, 240, 300, 360$  number of processors  $m = 5, 10, 15, 20, 25, 30$  and number of primary memory pages  $N = 10.000$  were generated parameters  $a_{ik}, b_{ik}$  from the set  $\{0.2, 0.4, \dots, 9.8, 10.0\}$ . For each combination of  $n$  and  $m$  were generated 50 instances. The results of comparative analysis of heuristic algorithm proposed in this paper and the algorithm *LPT* are showed in the Table 1.

Table 1. The results of comparative analysis of heuristic algorithm and algorithm  $LPT$ 

$n/m$	Number of instances, when:			$\Delta^H$	$S^H$	$S^{LPT}$
	$T_{opt}^H < T_{opt}^{LPT}$	$T_{opt}^H < T_{opt}^{LPT}$	$T_{opt}^H < T_{opt}^{LPT}$	%	sec	sec
60/5	25	1	24	1,9	2,8	2,2
120/5	26	2	22	2,2	5,6	4,9
180/5	27	3	20	3,6	8,3	7,8
240/5	27	2	21	4,8	13,4	11,2
300/5	28	2	20	5,9	16,2	12,4
360/5	29	3	18	6,3	17,4	14,6
60/10	25	1	24	2,4	3,9	3,2
120/10	25	2	23	3,6	5,8	5,2
180/10	27	2	21	4,4	9,1	8,5
240/10	26	1	23	5,7	14,2	12,4
300/10	27	3	20	6,1	17,6	15,2
360/10	30	1	19	6,7	18,9	16,3
60/15	25	1	24	2,9	4,5	3,4
120/15	27	2	21	3,9	6,3	5,0
180/15	28	2	20	5,1	10,4	8,4
240/15	29	3	18	6,2	15,6	13,8
300/15	28	1	21	6,7	18,4	16,8
360/15	30	3	17	7,2	19,6	16,9
60/20	26	1	23	2,6	4,9	4,4
120/20	28	1	21	3,8	7,2	5,8
180/20	27	3	20	4,5	11,5	9,5
240/20	30	2	18	5,9	16,8	13,9
300/20	32	1	17	6,8	19,8	17,3
360/20	34	0	16	7,5	20,6	17,6
60/25	26	1	23	2,6	5,6	4,9
120/25	28	0	22	3,9	8,4	6,9
180/25	29	1	20	4,7	12,7	9,8
240/25	31	2	17	5,5	17,0	14,5
300/25	32	4	14	6,4	20,2	17,5
360/25	34	2	14	7,7	21,8	19,2
60/30	26	0	24	2,9	6,0	5,2
120/30	28	1	21	3,8	9,1	7,4
180/30	30	2	18	5,1	14,0	10,5
240/30	32	2	16	6,4	18,2	15,5
300/30	34	1	15	7,5	21,3	18,2
360/30	36	2	12	8,2	23,2	19,8

In the Table 1 there are the following designations:

- $n$  – number of programs,
- $m$  – number of processors,
- $T_{opt}^H$  – total processing time of all set of programs  $J$  for the heuristic algorithm,
- $T_{opt}^{LPT}$  – total processing time of all set of programs  $J$  for the algorithm  $LPT$ ,
- $\Delta^H$  – the mean value of the relative improvement  $T_{opt}^H$  in relation to  $T_{opt}^{LPT}$ :

$$\Delta^H = \frac{T_{opt}^{LPT} - T_{opt}^H}{T_{opt}^H} \cdot 100\%,$$

- $S^H$  – the mean time of the numerical calculation for the heuristic algorithm,
- $S^{LPT}$  – the mean time of the numerical calculation for the algorithm  $LPT$ .

## 5. FINAL REMARKS

Computational experiments presented above show, that quality of programs scheduling in parallel multiprocessing computer system based on the proposed in this paper heuristic algorithm increased in compare with simple  $LPT$  algorithm. The few percentages improvement of time  $T^H$  in compare with  $T^{LPT}$  can be the reason why heuristic algorithms researches will be successfully taken in the future.

Application of presented in this paper heuristic algorithm is especially good for multiprocessing computer systems with great number of programs because in this case the  $\Delta^H$  improvement is the highest. Proposed heuristic algorithm can be used not only to programs scheduling in multiprocessing computer systems but also to task scheduling in parallel machines or even to operations scheduling in workplaces equipped with production machines.

## REFERENCES

- [1] BIANCO L., BŁAŻEWICZ J., DELL'OLMO P., DROZDOWSKI M., *Preemptive scheduling of multiprocessor tasks on the dedicated processors system subject to minimal lateness*. Information Processing Letters, 46, 1993, 109–113.
- [2] BIANCO L., BŁAŻEWICZ J., DELL'OLMO P., DROZDOWSKI M., *Linear algorithms for preemptive scheduling of multiprocessor tasks subject to minimal lateness*, Discrete Applied Mathematics, 72, 1997, 25–46.
- [3] BŁAŻEWICZ J., DROZDOWSKI M., WERRA D., WĘGLARZ J., *Scheduling independent multiprocessor tasks before deadlines*. Discrete Applied Mathematics 65 (1–3), 1996, 81–96.

- [4] BŁAŻEWICZ J., ECKER K., SCHMIDT G., WĘGLARZ J., *Scheduling in Computer and Manufacturing Systems*. Springer-Verlag, Berlin-Heidelberg, 1993.
- [5] BŁAŻEWICZ J., LIU Z., *Scheduling multiprocessor tasks with chain constraints*. European Journal of Operational Research, 94, 1996, 231–241.
- [6] BOCTOR F., *A new and efficient heuristic for scheduling projects with resources restrictions and multiple execution models*. European Journal of Operational Research, Vol. 90, 1996, 349–361.
- [7] BRAH S.A., LOO L.L., *Heuristics for scheduling in a flow shop with multiple processors*, European Journal of Operational Research, Vol. 113, No. 1, 1999, 113–122.
- [8] BUCHALSKI Z., *Time-optimal memory allocation and programs scheduling in multiprocessing systems with common primary memory*, Raport serii PREPRINT No 14/83, Wydawnictwo Politechniki Wrocławskiej, Wrocław, 1983 (in Polish).
- [9] BUCHALSKI Z., *Application of heuristic algorithm for the tasks scheduling on parallel machines to minimize the total processing time*. Proceedings of the 15<sup>th</sup> International Conference on Systems Science, Vol. 2, Wrocław, 2004.
- [10] BUCHALSKI Z., *Minimising the Total Processing Time for the Tasks Scheduling on the Parallel Machines System*. Proc. of the 12<sup>th</sup> IEEE International Conference on Methods and Models in Automation and Robotics, Domek S., Kaszyński R. (Eds.), Międzyzdroje, Poland, MMAR 2006, 28–31 August 2006, 1081–1084.
- [11] CHENG J., KARUNO Y., KISE H., *A shifting bottleneck approach for a parallel-machine flow-shop scheduling problem*, Journal of the Operational Research Society of Japan, Vol. 44, No. 2, 2001, 140–156.
- [12] GUPTA J.N.D., HARIRI A.M.A., POTTS C.N., *Scheduling a two-stage hybrid flow shop with parallel machines at the first stage*, Annals of Operations Research, Vol. 69, No. 0, 1997, 171–191.
- [13] JANIĄK A., KOVALYOV M., *Single machine scheduling subject to deadlines and resources dependent processing times*. European Journal of Operational Research, Vol. 94, 1996, 284–291.
- [14] JÓZEF CZYK J., *Task scheduling in the complex of operation with moving executors*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1996 (in Polish).
- [15] JÓZEF CZYK J., *Selected Decision Making Problems in Complex Operation Systems*, Monografie Komitetu Automatyki i Robotyki PAN, t. 2, Oficyna Wydawnicza Politechniki Wrocławskiej, Warszawa–Wrocław, 2001 (in Polish).
- [16] JÓZEFOWSKA J., MIKA M., RÓŻYCKI R., WALIGÓRA G., WĘGLARZ J., *Discrete-continuous scheduling to minimize maximum lateness*, Proceedings of the Fourth International Symposium on Methods and Models in Automation and Robotics MMAR'97, Międzyzdroje, Poland, 1997, 947–952.
- [17] JÓZEFOWSKA J., MIKA M., RÓŻYCKI R., WALIGÓRA G., WĘGLARZ J., *Local search meta-heuristics for discrete-continuous scheduling problems*, European Journal of Operational Research, 107, 1998, 354–370.
- [18] JÓZEFOWSKA J., WĘGLARZ J., *Discrete-continuous scheduling problems – mean completion time result*, European Journal of Operational Research, Vol. 94, No. 2, 1996, 302–310.
- [19] JÓZEFOWSKA J., WĘGLARZ J., *On a methodology for discrete-continuous scheduling*, European Journal of Operational Research, Vol. 107, No. 2, 1998, 338–353.
- [20] NOWICKI E., SMUTNICKI C., *The flow shop with parallel machines. A Tabu search approach*. European Journal of Operational Research 106, 1998, 226–253.
- [21] WĘGLARZ J., *Multiprocessor scheduling with memory allocation – a deterministic approach*. IEEE Trans. Comput., C-29, 1980, 703–710.

**PART 4**  
**SECURITY AND PRIVACY**



Marek JASIŃSKI\* , Ryszard ANTKIEWICZ\*\*

## **THE HIDDEN PERT MODEL**

The main problem considered in the paper is the detection of projects realized secretly. Terrorist organizations activity, complex criminal activity, preparation of military operations, secret state programs of nuclear or biological weapon production are the examples of hidden projects. We assume that direct observation of a hidden project is impossible. But each action of a hidden project can generate observations, which could be collected. The problem is how we could identify current state of preparation to the action and evaluate the time remaining to the end of the action using gathered observations. The end of the action may mean terrorist attack. Such problem was considered in literature. Hidden Markov Model (HMM) was used there as a model of a hidden project. The drawback of this model is a fact, that Markov Model could not describe parallel realized actions. We propose PERT network as a model of hidden project with possible parallel actions.

### 1. INTRODUCTION

The PERT model has been used commonly since 1950s as the model of complex projects. In the PERT model weighted graph is used as a model of project, where nodes mean actions and edges describe sequence of actions. It is commonly assumed that duration of actions are random variables.

The main problem considered in the paper is the detection of projects realized secretly. Terrorist organizations activity, complex criminal activity, preparation of military operations, secret state programs of nuclear or biological weapon production are the examples of hidden projects. We assume that direct observation of a hidden project is impossible. But each action of a hidden project can generate observations, which could be collected. The problem is how we could identify current state of preparation

---

\* State Higher Vocational School in Nowy Sacz, Jagiellońska Str. 61, 33-300 Nowy Sacz.

\*\* Military University of Technology, Cybernetics Faculty, gen. S. Kaliskiego Str. 2, 00-908 Warsaw.

to the action and evaluate the time remaining to the end of the action using gathered observations. The end of the action may mean terrorist attack.

Such problem was considered in papers [2], [5], [7]. Hidden Markov Model (HMM) [6] was used there as a model of a hidden project. The drawback of this model is a fact, that Markov Model could not describe parallel realized actions. We propose PERT [3] network as a model of hidden project with possible parallel actions. In the rest of the paper we will present: hidden PERT model (HPERT) and methods of current state of hidden activity identification, method of probability distribution of time to the end of considered activity calculation.

## 2. HIDDEN PERT MODEL

The PERT model is three-tuple:

$$S_p = \langle G, \Phi, \Psi \rangle \quad (1)$$

where:

$$G = \langle C, L \rangle \quad (2)$$

- $G$  – directed graph without cycles and loops [3],
- $C = \{c_1, c_2, \dots, c_{N_c}\}$  – set of nodes, nodes represent actions of project,
- $L \subset C \times C, L$  – set of arcs, arcs represent sequence relations between actions represented by the nodes,
- $\Phi = \{f \mid f: C \rightarrow R\}$  – set of functions defined on the network nodes, in this model  $\Phi = \{\tau_c, F_c, Pred(c), c \in C\}$ , where:
  - $\tau_c$  – random variable, which value means duration of action  $c \in C$ ,
  - $F_c(t) = P\{\tau_c < t\}$  – distribution function of random variable  $\tau_c$ ,
  - $Pred(c) = \{e \in C \mid (e, c) \in L\}$  – set of nodes (actions) which are direct predecessor of a node (action)  $c$ ,
- $\Psi = \{g \mid g: L \rightarrow R\}$  – set of functions defined on the set of arcs. In model we assume that  $\Psi = \emptyset$ .

Additionally we assume, that:

- duration of actions are independent random variables,
- duration of actions are geometrically distributed. It means that:

$$P\{\tau_{c_i} = k\} = (1 - p_i)^{k-1} \cdot p_i, \text{ for } k = 1, 2, \dots; c_i \in C. \quad (3)$$

We assume, that actions of hidden project generate observations according to the following rules:

- action active at time  $k$  generates only one observation,
- active actions generate observations independently of each other,
- observations are generated randomly, it means that a fixed action can generate fixed observation with a defined probability.

Taking into account these definitions and assumptions, we define hidden PERT model as follows:

$$HPM = \langle S_p, W, E \rangle \quad (4)$$

where:

- $W = \{w_1, w_2, \dots, w_{N_W}\}$  – set of all possible types of observations,
- Matrix  $E$  describes relations between actions and observations:

$$E = \left[ e_{ij} \right]_{\substack{i=1, \dots, N_C \\ j=1, \dots, N_W}} \quad (5)$$

and  $e_{ij}$  is a probability that active action  $c_i$  generates observation of type  $w_j \in W$  in one unit of time. We assume, that:

$$\sum_{j=1}^{N_W} e_{ij} = 1, \text{ for } i = 1, \dots, N_C. \quad (6)$$

Taking into consideration the assumptions of the HPM, the realization of hidden project can be described by the following stochastic process:

$$H(k) = (S(k), O(k)), k = 1, 2, \dots \quad (7)$$

where:

$S(k)$  – process, named further hidden process, which state at time  $k$  means subset of model PERT actions active at time  $k$ ,  
 $O(k)$  – process, named further process of observations, which state at time  $k$  means subset of observations generated by the actions active at time  $k$ . States of process  $O(k)$  will be called a complex observations. Graphical illustrations of process  $H(k)$  is given at the Fig. 1.

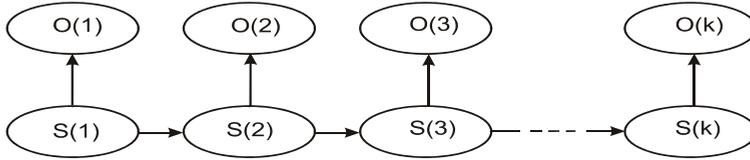


Fig. 1. Model of a hidden project realization

Considering assumptions and definition of *HPM* and definition of  $H(k)$  process it could be proved that process  $H(k)$  is a two-dimensional Markov chain. Probability description of process  $H(k)$  contains the following elements:

- $\bar{S} = \{s_1, s_2, \dots, s_{N_S}\}$ , where  $s_i \subseteq C$ , for  $i = 1, \dots, N_S$ ,  $S$  – set of hidden process  $S(k)$  states,
- $\bar{O} = \{o_1, o_2, \dots, o_{N_O}\}$ , for  $i = 1, \dots, N_O$ ,  $\bar{O}$  – set of all complex observations,
- $A = [a_{ij}]_{i,j=1,\dots,N_S}$ ,  $a_{ij} = P\{S(k+1) = s_j \mid S(k) = s_i\}$ ,  $i, j \in \{1, 2, \dots, N_S\}$ ,  $A$  matrix of transition probabilities of hidden process  $S(k)$ ,
- $B = [b_{ij}]_{i=1,\dots,N_S, j=1,\dots,N_O}$ ,  $b_{ij} = P\{O(k) = o_j \mid S(k) = s_i\}$ ,  $i \in \{1, 2, \dots, N_S\}$ ,  $j \in \{1, \dots, N_O\}$ ,  $B$  – matrix of observations emission (generation);
- $\Pi(k) = [\pi_i(k)]_{i=1,\dots,N_S}$ ,  $\pi_i(k) = P\{S(k) = s_i\}$ ,  $\Pi(k)$  – vector of process distribution at time  $k$ .

Assuming that we know values of all parameters of *HPM* model, we could compute values of all elements of process  $H(k)$  probability description.

Set  $\bar{S}$  could be determined using the following algorithm:

### **Algorithm 1**

$i = 1$ ;

$k = 2$ ;

$S' = \emptyset$ ;

**Create list  $L$ ;**

$L := L \oplus (s_1, z_1)$ , where  $s_1 = C_s$ ,  $z_1 = \emptyset$ ;

**While  $L \neq \emptyset$  do**

For element  $(s_i, z_i)$  from the list  $L$  create all subsets  $z_i^j$  of the set  $s_i$  without empty set;

**For  $j = 1$  to  $2^{|s_i|} - 1$  do**

**If  $\neg \exists (\bar{s}, \bar{z}) \in S^i: \bar{z} = z_i \cup z_i^j$  then**

$z_k = z_i \cup z_i^j$ ;  
 $s_k = \{c_r \in C : Pred(c_r) \subset z_k \wedge c_r \notin z_k\}$ ;  
 $L := L \oplus (s_k, z_k)$ ;  
 $k := k + 1$ ;  
**end If**  
**end For**  
 $L := L \setminus (s_i, z_i)$ ;  
 $i := i + 1$ ;  
**end While.**

Element  $z_i$  of vector the  $(s_i, z_i)$  can be interpreted as a set of closed nodes (actions), when  $s_i$  is a set of active nodes (actions)

Set  $\bar{S}$  is created as follows:

$$\bar{S} = \{s_i \subset C \mid i \in \{1, 2, \dots, N_S\} \wedge ((s_i, z_i) \in S')\} \quad (8)$$

Matrix  $A = [a_{ij}]_{i,j=1,\dots,N_S}$  of transition probabilities is computed using the following formulas:

- For  $i, j \in \{1, 2, \dots, N_S\} \wedge i > j$   
 $a_{ij} = 0$ , it results from method of numeration of states from set  $\bar{S}$  applied in **Algorithm 1**;
- For  $i, j \in \{1, 2, \dots, N_S\} \wedge i = j$   
 $a_{ij} = \prod_{c_k \in s_i} (1 - p_k)$
- For  $i, j \in \{1, 2, \dots, N_S\} \wedge i < j$   
 $a_{ij} = \begin{cases} 0 & \text{for } z_i \cap s_j \neq \emptyset \vee z_j \setminus (s_i \cup z_i) \neq \emptyset \\ t_1^{ij} \cdot t_2^{ij} & \text{for } z_i \cap s_j = \emptyset \wedge z_j \setminus (s_i \cup z_i) = \emptyset \end{cases}$

where:

$$t_1^{ij} = \begin{cases} \prod_{c_k \in s_i \cap z_j} p_k & \text{for } |s_i \cap z_j| > 0 \\ 0 & \text{for } s_i \cap z_j = \emptyset \end{cases}$$

$$t_2^{ij} = \begin{cases} \prod_{c_k \in s_i \cap s_j} (1 - p_k) & \text{for } |s_i \cap s_j| > 0 \\ 1 & \text{for } s_i \cap s_j = \emptyset \end{cases}$$

Set  $s_i \cap z_j$  we could interpret as a set of actions which ended with transition from state  $s_i$  to state  $s_j$ .

Set  $s_i \cap s_j$  we could interpret as a set of actions which are continued with transition from state  $s_i$  to state  $s_j$ .

According to the definition (7) of the process  $H(k)$ , state of the process  $O(k)$  means subset of observations generated by actions active at time  $k$ . Because types of observations generated by different active actions could be repeated, therefore we define state of process  $O(k)$  as a multiset:

$$\begin{aligned} \bar{O} = \{o_j = \langle W, m_j \rangle \mid W = \{w_1, w_2, \dots, w_{N_W}\}, m_j = (m_j^1, m_j^2, \dots, m_j^{N_W}), \\ m_j^k : W \rightarrow N, \text{ for } j = 1, \dots, N_O\} \end{aligned} \quad (9)$$

where  $m_j^k : W \rightarrow N$  is a function which value means number of occurrence of  $k$ -th type observation in  $j$ -th complex observation.

Elements of matrix  $B$  of observations emission we calculate using formulas given lower:

$$b_{ij} = \begin{cases} \sum_{u_{jk} \in U_j} \prod_{r=1}^{|o_j|} e_{I_C(s_i^r), I_W(w_{jk}^r)} & |o_j| = |s_i| \\ 0 & |o_j| \neq |s_i| \end{cases} \quad (10)$$

where:

- $U_j = \{u_{jk}\}$  – set of permutations of  $j$ -th complex observation,
- $u_{jk} = (w_{jk}^1, w_{jk}^2, \dots, w_{jk}^{|o_j|})$  –  $k$ -th permutation of  $j$ -th complex observation,
- $|o_j| = \sum_{k=1}^{N_W} m_j^k$ ,
- $\bar{s}_i = (s_i^1, s_i^2, \dots, s_i^{|s_i|})$  – vector of actions active in the state  $s_i$  ordered by number,
- $I_C : C \rightarrow \{1, 2, \dots, N_C\}$ ,  $I_C(c_i) = i$
- $I_W : W \rightarrow \{1, 2, \dots, N_W\}$ ,  $I_W(w_k) = k$

As an initial probability of the hidden process  $S(k)$  we take the vector:

$$\begin{aligned} \Pi(0) &= [\pi_i(0)]_{i=1, \dots, N_S} = [\delta_{li}]_{i=1, \dots, N_S} \\ \delta_{ij} &= \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \end{aligned} \quad (11)$$

### 3. HIDDEN PERT MODEL ANALYSIS

The Hidden PERT Model can be applied for solving the following problems:

- problem 1.a. – to find most probable state of the process  $S(k)$  (set of currently active actions in the hidden project) at the time of registration of the last observation, given that we have a sequence of observations (sequence of realizations of process  $O(k)$ ),
- problem 1.b. – to find probability distribution of the process  $S(k)$  at the time of registration of the last observation, given that we have a sequence of observations,
- problem 2.a. – to find probability distribution of the time from the moment of the last observation to the end of the hidden project, given that the most probable state of the process  $S(k)$  at the moment of last observation is known (as the solution of the problem 1.a.),
- problem 2.b. – to find probability distribution of the time from the moment of registration of the last observation to the end of the hidden project, given that the probability distribution of the process  $S(k)$  at the moment of the last observation is known (as the solution of the problem 1.b.).

In order to solve the problem 1.a, we formulate auxiliary problem, which can be described as a search for such sequence  $\bar{s}_n^*$  of process  $S(k)$  states, that for given  $\bar{o}_n$  we have:

$$P\{\bar{S}_n = \bar{s}_n^* \mid \bar{O}_n = \bar{o}_n\} = \max_{\bar{s}_n \in S_r^n} P\{\bar{S}_n = \bar{s}_n \mid \bar{O}_n = \bar{o}_n\} \quad (12)$$

where:

- $\bar{S}_n = (S(1), S(2), \dots, S(n))$  – vector of states of process  $S(k)$ ,
- $S_r^n = \{\bar{s}_n = (s_1, s_2, \dots, s_n) \mid s_i \in \bar{S}, i = 1, 2, \dots, n; P\{\bar{S}_n = \bar{s}_n\} > 0\}$  – set of possible realizations of vector  $\bar{S}_n$ ,

- $\bar{O}_n = (O(1), O(2), \dots, O(n))$  – vector of states of process  $O(k)$ ,
- $\bar{o}_n = (o_n^1, o_n^2, \dots, o_n^n)$  – realization of vector  $\bar{O}_n$  (vector of complex observations).

It is important to note, that the process  $H(k)$  can be interpreted as an Hidden Markov Model. Then we can use Viterbi algorithm in order to solve problem given by (12) [1], [6]. We find a solution of the problem 1.a by taking element  $s_n^*$  from the last position of the sequence  $\bar{s}_n^* = (s_1^*, s_2^*, \dots, s_n^*)$ .

The problem 1.b we can solve using the following algorithm:

### **Algorithm 2**

**For**  $i = 1, 2, \dots, N_S$  **do**

$v[0, i] := \pi_i(0)$ ;

**end For**;

**For**  $k = 1, 2, \dots, n$  **do**

**For**  $j = 1, 2, \dots, N_S$  **do**

$$v[k, j] := \sum_{i=1}^{N_S} v[k-1, i] \cdot a_{i,j} \cdot b_{j, I_O(\bar{o}_n^k)};$$

**end For**;

$$S_v := \sum_{j=1}^{N_S} v[k, j];$$

**For**  $j = 1, 2, \dots, N_S$  **do**

$$v[k, j] := \frac{v[k, j]}{S_v};$$

**end For**;

**end For**;

$$s_n^* = \arg \max_{1 \leq i \leq N_S} v[n, i]$$

**end Algorithm 2.**

Probability distribution of process  $S(k)$  at time  $n$  (time of registration the last observation) is given by the formula:

$$P\{S(n) = s_i \mid \bar{O}_n = \bar{o}_n\} = v[n, i], \quad \text{for } i = 1, \dots, N_S \quad (13)$$

Let us define the following random variable:

$$T_{s_n^*}^{S_{N_S}} = \min \{k \geq 0 : S(k) = s_{N_S} \mid S(0) = s_n^*\} \quad (14)$$

It could be noted, that value of  $T_{s_n^*}^{S_{N_S}}$  means time from the moment of last observation to the end of hidden project, given that  $s_n^*$  is the most probable state of the process  $S(k)$  at the moment of last observation. Then, the problem 2.a could be formulated, as problem of finding probability distribution of the random variable  $T_{s_n^*}^{S_{N_S}}$ .

Taking into account (14), we could write:

$$F_{s_n^*}^{S_{N_S}}(k) = P\{T_{s_n^*}^{S_{N_S}} \leq k\} = P\{S(k) = s_{N_S} \mid S(0) = s_n^*\} = [A^k]_{I_{S(s_n^*), N_S}}, \text{ for } k = 1, 2, \dots \quad (15)$$

Let us define the random variables  $\tau_n^{S_{N_S}}$  which value means time from moment n to the moment of reaching state  $s_{N_S}$  by the process  $S(k)$ . Therefore, problem 2.b consists in an evaluation of probability distribution of  $\tau_n^{S_{N_S}}$ . We could write, that:

$$\bar{F}_n(k, \bar{o}_n) = P\{\tau_n^{S_{N_S}} \leq k \mid \bar{O}_n = \bar{o}_n\} = [\Pi^*(0)A^k]_{N_S} \quad (16)$$

where:  $\Pi^*(0) = [\pi_i(0)]_{i=1, \dots, N_S}$ ,  $\pi_i(0) = P\{S(n) = s_i \mid \bar{O}_n = \bar{o}_n\}$  (17)

#### 4. NUMERICAL EXAMPLE

Let us consider example of application HPM to analysis hidden operations. We would like to illustrate methods for solving problems 1.a, 1.b and 2.b. Hidden PERT Model of collection resources operation is shown in Fig. 2. This model is inspired by Markov Chain Model presented in Fig. 10 in [2]. Hidden process  $S(k)$  from process  $H(k)$  is presented in Fig. 3.

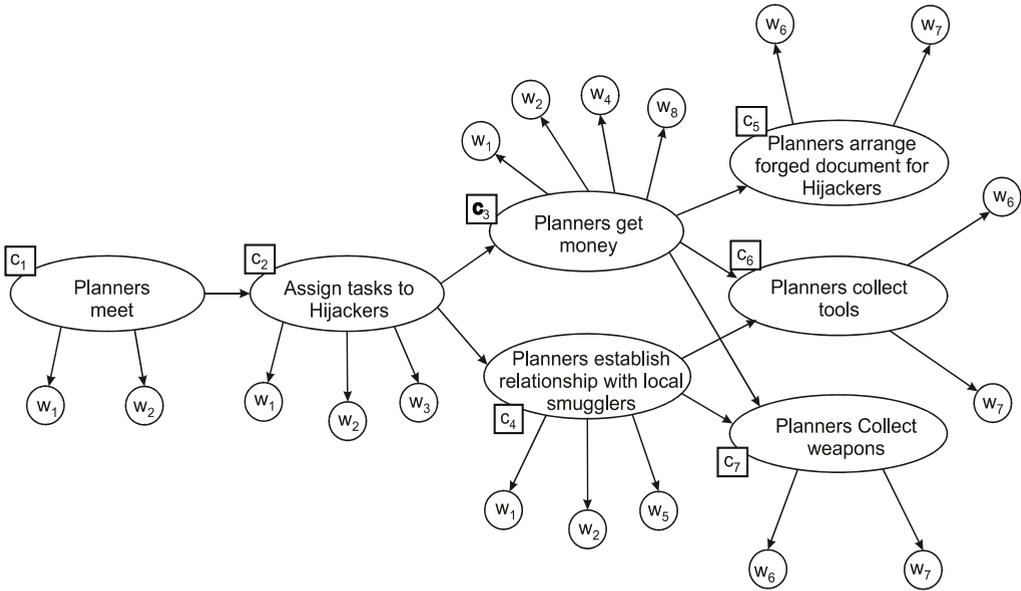


Fig. 2. HPM for Collect Resources

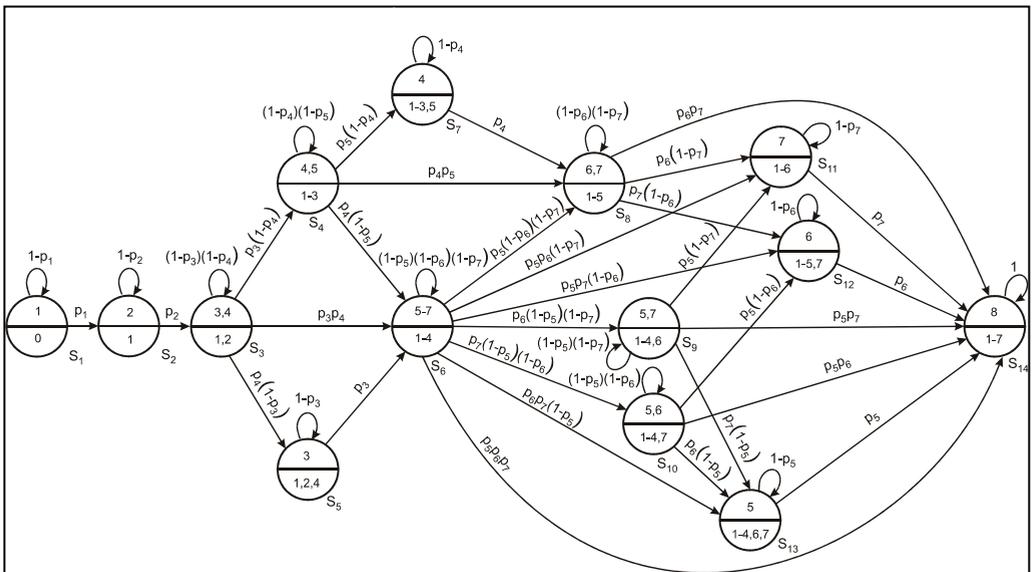


Fig. 3. Hidden process  $S(k)$  for process of collection resources

Probabilities of observations emission are given in the Table 1. The Table 2 contains parameters of actions duration distribution.

Table 1. Matrix of observation emission probabilities

	$w_1$	$w_2$	$w_3$	$w_4$	$w_5$	$w_6$	$w_7$	$w_8$
$c_1$	0.5	0.5	0	0	0	0	0	0
$c_2$	0.2	0.4	0.4	0	0	0	0	0
$c_3$	0.6	0.1	0	0.2	0	0	0	0.1
$c_4$	0.2	0.3	0	0	0.5	0	0	0
$c_5$	0	0	0	0	0	0.2	0.8	0
$c_6$	0	0	0	0	0	0.3	0.7	0
$c_7$	0	0	0	0	0	0.4	0.6	0

Table 2. Parameters of actions duration distribution

$i$	1	2	3	4	5	6	7
$p_i$	0.5	0.45	0.25	0.6	0.8	0.8	0.8

Let us assume that we have the following sequence of observations:  $\bar{o}_5 = (o_1, o_3, o_9, o_{31}, o_{27})$ , where  $W = (w_1, w_2, \dots, w_8)$  and  $m_1 = (1, 0, 0, 0, 0, 0, 0, 0)$ ,  $m_3 = (0, 0, 1, 0, 0, 0, 0, 0)$ ,  $m_9 = (2, 0, 0, 0, 0, 0, 0, 0)$ ,  $m_{27} = (0, 0, 0, 0, 0, 1, 1, 0)$ ,  $m_{31} = (0, 0, 0, 0, 0, 1, 2, 0)$ .

Applying Viterbi algorithm we obtain the most probable sequence of process  $S(k)$  states which is the following  $\bar{s}_5^* = (s_1, s_2, s_3, s_6, s_8)$  and the most probable state of the process  $S(k)$  at the time  $k = 5$  is  $s_5^* = s_8$  (solution of the [problem 1.a.](#)).

Using **Algorithm 2** we obtain the probability distribution of the process  $S(k)$  at the time  $k=5$  (solution of the [problem 1.b.](#)):

$$P\{S(5) = s_8 \mid \bar{O}_5 = \bar{o}_5\} = 0,3594, \quad P\{S(5) = s_9 \mid \bar{O}_5 = \bar{o}_5\} = 0,3437,$$

$$P\{S(5) = s_{10} \mid \bar{O}_5 = \bar{o}_5\} = 0,2969, \quad P\{S(5) = s_j \mid \bar{O}_5 = \bar{o}_5\} = 0 \text{ for } j = 1-7, 11-14,$$

Taking into account above distribution we could find that the most probable state of the process  $S(k)$  at the time  $k = 5$  is equal to  $s_8$ . Identical result was obtained by application of Viterbi algorithm.

Distribution function  $\bar{F}_5(k, \bar{o}_5)$  we could obtain using a formula (16) (solution of the [problem 2.b.](#)). Approximation of this distribution is given in the table:

$k$	1	2	3	4	5	6	7
$\bar{F}_5(k, \bar{o}_5)$	0.64	0.9216	0.9841	0.9968	0.9994	0.9998	0.9999

## 5. SUMMARY

The Hidden PERT Model (HPM) is presented in the paper. This model allows to describe realization of a hidden project with parallel actions. We propose methods for finding most probable states and distribution of time to the end of the project given that some sequence of observations is known.

We plan to develop HPM. It will be considered: null observation, continuous distribution of action duration, other than geometric and exponential distribution of action duration.

## ACKNOWLEDGMENTS

This work was partially supported by the National Centre for Research and Development: ROB 0021/01/ID21/2 and DOBR/0069/R/ID1/2012/03.

## REFERENCES

- [1] FORNEY G.D., *The Viterbi Algorithm*, Proc. of the IEEE, Vol. 61, No. 3, pp. 268–278, March 1973.
- [2] HAIYING T., ALLANACH J., SINGH S., PATTIPATI K.R., WILLETT P., *Information integration via hierarchical and hybrid bayesian networks*, IEEE, Vol. 36, No.1, January 2006.
- [3] KORZAN B., *Grafy, hipergrafy i sieci*, Wojskowa Akademia Techniczna, Warszawa, 1980.
- [4] KULKARNI V.G., ADLAKHA V.G., *Markov and Markov-regenerative PERT networks*, Operations Research, 34:769–781, 1986.
- [5] LEE H., SINGH S., AN W., GOKHALE S.S., PATTIPATI K., KLEINMAN D.L., *Rollout Strategy for Hidden Markov Model (HMM)-based. Dynamic Sensor Scheduling*, IEEE International Conference on Systems, Man and Cybernetics, Montreal, Canada, October 2007.
- [6] RABINER L., JUANG B., *Fundamentals of Speech Recognition*, Prentice Hall International, New Jersey, 1993.
- [7] SINGH S., HAIYING T., DONAT W., PATTIPATI K., WILLETT P., *Anomaly Detection via Feature-Aided Tracking and Hidden Markov Models*, IEEE, Vol. 39, No.1:144–159, January 2009.

Agnieszka DURAJ\*

## **PRESERVATION OF PRIVACY IN ASSOCIATION RULES**

Access to the vast amount of data results in a variety of their abuse or modifications of the data. It can also become the reason for extorting sensitive data. The article addresses the issue of privacy protection in the rules of association. The study used a well-known algorithms, namely the Apriori and FP-Growth algorithms. In the experiments, special emphasis was placed on assessing the impact of privacy protection methods on the detection of association rules. In order to preserve the privacy of data the following were used blurring of association rules basing on perturbations of centralized data, the blurring of association rules based on blockage of centralized data, cryptography, securing the data divided vertically or horizontally in the discovery of association rules, the reconstruction methods. Tests have shown that the privacy protection algorithms have an impact on the detection of association rules. The simultaneous selective perturbation of one column and selective blocking of the other too strongly affects the data set, since both algorithms have found only 30% of the frequent set and not a single associative rule. The FP-Growth algorithm copes with the modified data much better. It also showed significantly better results in terms of time.

### **1. INTRODUCTION**

Currently we collect, store and analyze massive amounts of data. These data come from government institutions, research centers, the Internet, corporations and businesses. The main task of the data mining domain is primarily to provide methods that enable and facilitate the process of data analysis in terms of searching for the most relevant information in large data sets. On the other hand, the availability and ownership of data causes all kinds of abuse, modification of data and leads to extortion of sensitive data. It is therefore essential that algorithms to protect data privacy be formed. Data mining is a complex process that uses a very large range of algorithms

---

\* Institute of Information Technology, Lodz University of Technology, 90-924 Lodz, Poland, ul. Wolczanska 215.

for classification, clustering, detecting all kinds of exceptions or detecting dependencies between data.

Algorithms for discovering association rules are a very popular tool for detecting dependencies. These rules are in the form of conditional sentences because their construction is based on the logical implications “if sentence  $Z_1$ , then sentence  $Z_2$ ” ( $Z_1 \Rightarrow Z_2$ ). In the mathematical sense, we should talk here about the distribution of events both in time and space, and therefore about the structural and cause-and-effect relationships. Association rules are an example of learning without a teacher, which can be divided according to their application or utility. In the first group we distinguish the following rules: dissociative – formulated as ‘if A and NOT B, then C’; recurring – include repeatability and are profiled with regard to time; sequencing – examine the relationships between transactions in a certain period of time; valuable – take into account information on quantity and value; substitutive – require supplementing by expert knowledge. Another division of rules according to usefulness distinguish: useful (which discover previously unknown patterns), trivial (which discover classic, commonly known patterns), unexplained (which discover unexpected patterns, which do not translate into marketing activities).

Privacy protection is to secure two types of data. Firstly, sensitive raw data such as account numbers, personal information, addresses should be modified or deleted prior to exploration. Secondly, sensitive knowledge extracted from the database should also be excluded from search results. The main task of privacy preservation is to protect sensitive data both raw and extracted from the database using the algorithms appropriately modifying the original data set.

Privacy protection is considered in several dimensions. The first dimension is the distribution of data, which, in particular, draws attention to the dispersion of data (data centralized and dispersed horizontally and vertically). The second dimension is the modification of the data, where we can distinguish: perturbations – renaming of the attribute value to the new value or adding noise; blocking – the replacement of value by the sign?; aggregation – change of the value of individual attributes in one category; conversion – refers to the conversion of values for specific records; sampling – pruning the database only to random records. The third dimension is a data mining algorithms. In this case, the division of privacy protection algorithms depends on the purposes for which the data will be used. This is important due to the possibility of modifying the data, which must be adjusted so that the data mining algorithm has no problems with their processing. The fourth dimension is the hiding of data that is processing the original database in order to remove raw sensitive data or hide rules. It is much more difficult than in the case of raw data and thus forced the development and use of heuristic methods. The fifth and at the same time the most important dimension is the selective modification. It is designed to hide sensitive data from the database so

that it is useful in further studies. There are several methods used, for example: heuristic methods – these are techniques allowing the use selective modification; methods based on cryptography – they solve the problem of protection of privacy of input and output data in operations involving multiple users; and the reconstruction methods – that make it possible to reconstruct the original data based on the random data.

Data Storage and especially discovering relationships between them various methods of data mining provides important problem in years of research. Another problem is the sensitive data that allow identification of individuals or data characteristics. The aim of the study was to build a tool that allows the use of data mining algorithms while maintaining the privacy. This is according to the author of an important research process because even for medical or financial data. This study is an initial step to create more sophisticated applications related to the detection of exceptions in databases of privacy. Prepared and presented results provide a basis for optimizing detection algorithms exceptions of privacy.

Privacy of data mining algorithms is very important. Especially hide the aggregated data in the form of rules is difficult. Is the basis for the development of new methods of using heuristic rules. Reduction of publicly available information results in the formation of weak filtering rules to prevent database. The protection of privacy, as shown above, is of great importance in data mining. The conducted studies are therefore fully justified. The article has the following structure. In section 2 there is a brief description of privacy protection algorithms that have been used for the detection of association rules. The following were considered: the blurring of association rules based on perturbations of centralized data, the blurring of association rules based on blocking centralized data, securing data divided vertically or horizontally in the discovery of association rules, and methods of reconstruction. In section 3, there is analysis and evaluation of the applied privacy protection algorithms for association rules defined by the Apriori and FP-Growth algorithms.

## 2. PRIVACY PROTECTION ALGORITHMS USED IN ASSOCIATION RULES

The association rules algorithms of are adapted to the needs of a specific task. Most of the rules is based on the Apriori algorithm. Also numerous modifications are used e.g. AprioriTID. Another approach to detecting rules is presented by the FP-Growth algorithm. It is based on FP-trees, also called acyclic graphs.

The first and quite often used method for privacy protection is a blurring of association rules basing on perturbations of centralized data. Supposing that  $D$  is a source

database,  $R$  is a set of significant rules extracted from  $D$ , then let  $R_h$  be a set of sensitive rules. How should base  $D$  be modified into  $D'$ , so that it is possible to find all the rules from the set  $R$  with the exception of those from the set  $R_h$ ? A heuristic method that solves the above problem is to use a selective perturbation of data into the opposite value, so that support for vulnerable rules has been reduced and the usefulness of the data from the set  $D'$  has been maintained at the highest level. Here the measure of utility is the number of insensitive rules which were lost during the modification and the number of emerging insignificant rules in the set of significant rules. Centralized data perturbation based association rule presented in [4, 6].

Another method is the blurring of association rules based on blockage of centralized data. The essence of this method is replacing sensitive data by a question mark. This method is preferable in the processing of for example medical databases, where inserting unknown value is better than its falsification.

Entering the new value of the variable has an effect on the value of support and confidence, for which a defined range becomes the minimum value. If the designated support or confidence of sensitive rules assumes the value below the average of the value within the range, it can be assumed that the confidentiality of the data has not been compromised. Centralized data blocking based association rule presented for example in [7–10].

The third method is based on cryptography [3, 11]. It is fully understandable that a company or a user conducting certain activities does not want to share their output data. The solution to this problem can be provided by methods based on the safety of operations involving multiple users, namely: secure sum, secure merge, secure size of intersection, scalar product.

The fourth method is to secure the data divided vertically or horizontally in the discovery of association rules. The search for sensitive association rules from data vertically divided, where the value of each attribute for the data set are divided between the sides, is possible by finding support for a given set. If the counter of the support of set can be calculated safely, it is possible to check whether the support is higher than the threshold value, which allows specifying whether the collection is frequent. The key element in this case is the calculation of the support data set and calculation of the scalar product of vectors representing a subset of, the particular part of the side. There are algorithms for the calculation of the scalar product, so as to hide the true values by placing them in the equation and masking by random data. In the horizontal division of database it is the transactions that are distributed across multiple sides. Global support of a data set is the sum of all local values of support. For example, the set  $X$  is supported if the support of the global value of  $X$  is greater than  $s\%$  of all transactions from a database. (see in [13]).

Another group are the reconstruction methods, which are, in a sense, an improvement to heuristic methods based on data perturbation. Data mining is not carried out on the data set resulting from perturbations but on aggregated data reconstructed on their basis. Popular reconstruction methods are, for example, techniques reconstructing numerical data.

Reconstruction of numerical data presented for example in [1, 2, 14] refers to the problem of building a decision tree classifier from training data, where the value of the selected records have been subjected to perturbations. When it is not possible to accurately estimate original values for each record, the author proposes a reconstructive method for estimating the distribution of values of the original data. Using the reconstructed data deployments, it is possible to build a classifier whose accuracy is comparable to the classifier constructed from the original data. For the distorted data the author considered the use of discretization or falsification of values. To reconstruct the original distribution of the data the Bayesian approach was used and three algorithms for the construction of the decision tree based on reconstructions of distribution were proposed.

A solution for data reconstruction which is better than using Bayesian methods is the method offered by Agrawal D. and Aggarwal Charu C. in [1] the use the EM algorithm (Expectation Maximization), which is maximizing the expected value for the reconstruction of distribution.

The EM algorithm determines the maximum probability of the original distribution of the data subjected to perturbation. The higher the number of the data, the closer the approximation to the original data can be observed. In the context of association rules with the preservation of privacy of data also the techniques of reconstruction of binary and labeled data are used. It has been proven that the sampling of a database preserves the privacy of the data while maintaining a high quality of the data set.

Another method is the use of the so-called hash functions. Hashing is the creation of a short number on the basis on of arbitrarily long message, so it is an invertible abbreviation, the so-called hash. This is done by using the hash functions, that is, one-way mixing function. Hash functions must satisfy two conditions: irreversibility (it is not possible to reconstruct the input data on the basis of the output), and collision resistance (two hash values cannot be identical for two different input messages). In addition, the hash function should not give the possibility of deducing any useful information based on its abbreviation. Therefore, hash function should behave like a random function as far as it is possible. The most popular hash functions are, among others, MD2, MD3, SHA, MD5, SHA-1, SHA-2.

Review of privacy preserving algorithm are presented in e.g. [5, 12, 15].

### 3. COMPARATIVE ANALYSIS – RESEARCH

The aim of this study was to examine how the methods of privacy affect the discovery of rules. The application was written in Java. As input accepts data in the form of \*. csv. The results are representative and provide the basis for an algorithm to detect the exceptions of privacy. The conducted studies took into account three basic comparative criteria, ie the rules found and frequent sets and their usefulness as well as execution time. The comparative analysis was based on the Apriori and FP-Growth algorithms. Additionally, in the experiments special emphasis was placed on assessing the impact of privacy protection methods in the case of the detection of association rules. In order to preserve the privacy of data the following were used:

- blurring of association rules basing on perturbations of centralized data,
- the blurring of association rules based on blockage of centralized data,
- Cryptography,
- securing the data divided vertically or horizontally in the discovery of association rules,
- the reconstruction methods.

Methods of protection of privacy were analyzed in terms of the following criteria. The first criterion is the execution time of the algorithm, i.e. the time that the algorithm needs to hide the assumed collection of sensitive data. The second is the utility of the data, it determines the rate of the loss of information after the execution of the algorithm or the loss of functionality of the data. The third is the level of uncertainty – the indicator of the amount of sensitive information that can still be detected. The last criterion is the resistance provided by the technique of privacy protection to various data mining algorithms.

For both algorithms used, namely Apriori and FP-Growth, in each test the support was set to 0.7 and confidence to 0.85. For the unmodified data, both algorithms detected the same frequent itemsets, but the FP-Growth algorithm had shorter execution time (Table 1). Deleting selected columns greatly accelerated the performance of the Apriori algorithm.

Table 1. Results of Apriori algorithm and FP-Growth algorithm for file without modification and file with deleted columns

		APriori	FP-Growth
File without modification	Total time	579	254
	Frequent itemsets count	18	11
File with deleted columns	Total time	172	197
	Frequent itemsets count	12	11

Modification by sampling significantly reduced the size of the database, which had a positive effect on search time for both algorithms. Both algorithms failed to find one frequent set. The algorithm FP-Growth proved to be more efficient because it found all association rules properly. The Apriori algorithm found false frequent sets as well as the false frequent rule. Sampling of the data does not interfere with the work of the data mining algorithms while maintaining privacy. The data are not distorted, which makes it possible to reconstruct the entire database.

In the subsequent tests two variants of blocking method were used. In the first one the method of blocking all records in the case of one particular column was used, and in the second one random number of records was blocked. A non-significant delay in the performance of the algorithm FP-Growth was noted. Blocking the entire column had a significant impact on the functioning of the Apriori algorithm. Blocking all the records in one column caused the Apriori algorithm to mistakenly detect 11 frequent sets and 18 false rules. Blocking of random records gives better results for the Apriori algorithm, but the result is not satisfactory. Search efficiency for frequent itemsets is approximately 60%, while for association rules it is 40%. The performance of the FP-Growth algorithm is much better, as it did not add any false data and found all association rules that were possible to find in both block variants. It should be noted that in the case of blocking the data are heavily modified and there is no possibility of their reconstruction.

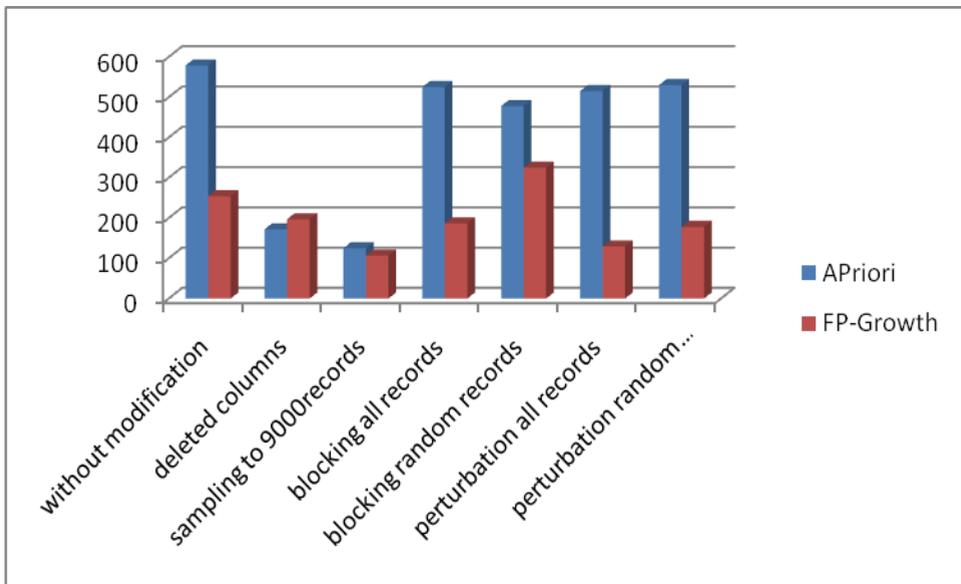


Fig. 1. Time of execution of the Apriori and FP-Growth algorithms using particular privacy protection algorithms

In the subsequent studies the method of perturbations of all records and randomly selected records was used. The perturbation does not influence significantly the duration of the execution of the Apriori algorithm, and the execution of the FP-Growth algorithm slightly accelerates. The obtained efficiency in finding frequent itemsets and association rules was on the similar level as in the case of blocking. In contrast, perturbations do not interfere with the execution of any of the algorithms. The data are adequately masked, so that the user of the modified database is not able to tell if it is the original or the modified data. Summary of the results are given in Table 2. Time of execution of the Apriori and FP-Growth algorithms using particular privacy protection algorithms are shown in Figure 1. Number of frequent itemsets and association of rules determined by the Apriori and FP-Growth algorithms using particular privacy protection algorithms are given in Figure 2.

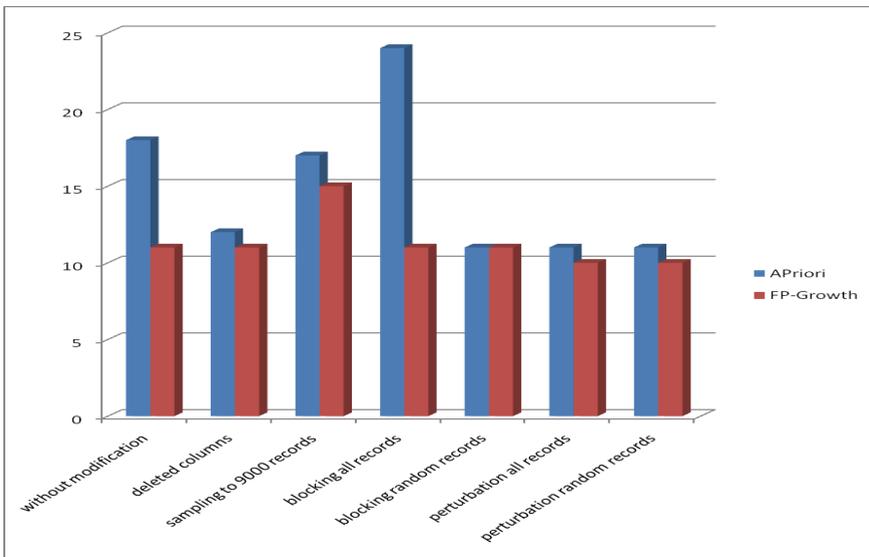


Fig. 2. Number of frequent itemsets and association of rules determined by the Apriori and FP-Growth algorithms using particular privacy protection algorithms

Table 2. Summary of results for privacy protection

		APriori	FP-Growth
Sampling to 9000 records	Total time	125	107
	Frequent itemsets count	17	15
Blocking all records	Total time	524	187
	Frequent itemsets count	24	11
Blocking random records	Total time	478	325
	Frequent itemsets count	11	11
perturbation all records	Total time	515	129
	Frequent itemsets count	11	10
random perturbation	Total time	530	178
	Frequent itemsets count	11	10

#### 4. SUMMARY

The conducted tests showed that the simultaneous selective perturbation of one column and selective blocking of the other too strongly affects the data set, since both algorithms have found only 30% of the frequent set and not a single associative rule. The FP-Growth algorithm copes with the modified data much better. It also showed significantly better results in terms of time. Blocking is a good method of modification if you want highly secure data, but it is advisable to use the FP-Growth algorithm to discover the association rules.

On the other hand, perturbation yields satisfactory results for both algorithms. It is also a good way to protect the privacy of sensitive data, because the user prepares very real results for the modified database. The most effective method of modification due to its subsequent impact on the performance of the algorithms searching association rules is sampling. However, this is the weakest method for privacy protection of sensitive data, because the user still operates on actual data but in their truncated versions.

Thus, the studies have shown and evaluated the performance of privacy protection algorithms for detecting frequent itemsets and association rules. In further experiments privacy protection will be taken into account in the case of algorithms searching for exceptions in databases.

#### REFERENCES

- [1] AGRAWAL D., AGGARWAL CHARU C., *On the design and quantification of privacy preserving data mining algorithms*, In Proceedings of the 20th ACM Symposium on Principles of Database Systems (2001), 247–255.
- [2] AGRAWAL R., SRIKANT R., *Privacy preserving data mining*, In Proceedings of the ACM SIGMOD Conference on Management of Data (2000), 2000, 439–450.
- [3] CLIFTON C., MARKS D., *Security and privacy implications of data mining*, In ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, May 1996, 15–19.
- [4] ATALLAH M.J., BERTINO E., ELMAGARMID A.K., IBRAHIM M., VERYKIOS V.S., *Disclosure Limitation of Sensitive Rules*, In Proceedings of the IEEE Knowledge and Data Engineering Workshop (1999), 45–52.
- [5] VERYKIOS V.S., BERTINO E., FOVINO I.N., PROVENZA L.P., SAYGIN Y., THEODORIDIS Y., *The State-of-the-art in Privacy Preserving Data Mining*, SIGMOD Record, Vol. 33, No. 1, March 2004, 50–57.
- [6] DASSEN E., VERYKIOS V.S., ELMAGARMID A.K., BERTINO E., *Hiding Association Rules by using Confidence and Support*, In Proceedings of the 4th Information Hiding Workshop (2001), 369–383.
- [7] CHANG LiWu, MOSKOWITZ I.S., *Parsimonious downgrading and decision trees applied to the inference problem*, In Proceedings of the 1998 New Security Paradigms Workshop (1998), 82–89.
- [8] CHANG Li. Wu, MOSKOWITZ I.S., *An integrated framework for database inference and privacy protection*, Data and Applications Security (2000), Kluwer, IFIP WG 11.3, The Netherlands, 161–172.

- [9] SAYGIN Y., VERYKIOS V., CLIFTON CH., *Using unknowns to prevent discovery of association rules*, SIGMOD Record 30 (2001), no. 4, 45–54.
- [10] SAYGIN Y., VERYKIOS V., ELMAGARMID A.K., *Privacy preserving association rule mining*, In Proceedings of the 12th International Workshop on Research Issues in Data Engineering (2002), 151–158.
- [11] WENLIANG DU., ATTALAH M.J., *Secure multi-problem computation problems and their applications: A review and open problems*, Tech.Wenliang Du and Zhijun Zhan, Building decision tree classifier on private data, In Proceedings of the IEEE ICDM Workshop on Privacy, Security and Data Mining (2002).
- [12] CLINTON Ch., KANTARCIOGLOU M., LIN X., ZHU M.Y., *Tools for privacy preserving distributed data mining*, SIGKDD Explorations (2002), no. 2, Vol. 4, 28–34.
- [13] LINDELL Y. PINKAS B., *Privacy preserving data mining*, In Advances in Cryptology – CRYPTO 2000 (2000), 36–54.
- [14] RIZVI S.J., HARITSA J.R., *Maintaing data privacy in association rule mining*, In Proceedings of the 28th International Conference on Very Large Databases (2002), 225–234
- [15] EVFIMIEVSKI A., SRIKANT R., AGRAWAL R., GEHRKE J., *Privacy preserving mining of association rules*, In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2002).

## BIBLIOTEKA INFORMATYKI SZKÓŁ WYŻSZYCH

- Information Systems Architecture and Technology. Advances in Web-Age Information Systems*, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2009
- Information Systems Architecture and Technology. Service Oriented Distributed Systems: Concepts and Infrastructure*, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2009
- Information Systems Architecture and Technology. Systems Analysis in Decision Aided Problems*, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2009
- Information Systems Architecture and Technology. IT Technologies in Knowledge Oriented Management Process*, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2009
- Information Systems Architecture and Technology. New Developments in Web-Age Information Systems*, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2010
- Information Systems Architecture and Technology. Networks and Networks Services*, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2010
- Information Systems Architecture and Technology. System Analysis Approach to the Design, Control and Decision Support*, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2010
- Information Systems Architecture and Technology. IT TModels in Management Process*, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2010
- Information Systems Architecture and Technology. Web Information Systems Engineering, Knowledge Discovery and Hybrid Computing*, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2011
- Information Systems Architecture and Technology. Service Oriented Networked Systems*, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2011
- Information Systems Architecture and Technology. System Analysis Approach to the Design, Control and Decision Support*, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2011
- Information Systems Architecture and Technology. Information as the Intangible Assets and Company Value Source*, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2011
- Information Systems Architecture and Technology. Web Engineering and High-Performance Computing on Complex Environments*, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2012
- Information Systems Architecture and Technology. Networks Design and Analysis*, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2012
- Information Systems Architecture and Technology. System Analysis Approach to the Design, Control and Decision Support*, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2012
- Information Systems Architecture and Technology. The Use of IT Models for Organization Management*, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2012
- Information Systems Architecture and Technology. Intelligent Information Systems, Knowledge Discovery, Big Data and High Performance Computing*, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2013
- Information Systems Architecture and Technology. Networks Architecture and Applications*, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2013
- Information Systems Architecture and Technology. Knowledge Based Approach to the Design, Control and Decision Support*, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2013
- Information Systems Architecture and Technology. Models of Decision Making in the Process of Management in a Risky Environment*, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2013

**Wydawnictwa Politechniki Wrocławskiej  
są do nabycia w księgarni  
plac Grunwaldzki 13, 50-377 Wrocław  
budynek D-1 PWr., tel. 71 320 29 35  
Prowadzimy sprzedaż wysyłkową  
zamawianie.ksiazek@pwr.wroc.pl**

**ISBN 978-83-7493-856-3**