# Performance Modelling and Evaluation of Heterogeneous Networks

PROCEEDINGS OF

6TH WORKING INTERNATIONAL CONFERENCE

# HET- NETs 2010

**EDITOR**

**Tadeusz Czachórski**

**Gliwice 2009**

## PUBLISHED BY

**Institute of Theoretical and Applied Informatics**

**of the Polish Academy of Sciences**

**Bałtycka 5, 44-100 Gliwice, POLAND**

**www.iitis.gliwice.pl**

## TECHNICAL PROGRAM COMMITTEE



## ORGANIZING COMMITTEE

Chair: **Krzysztof Grochla**

pts@iitis.gliwice.pl,

Phone: +48 32 231 73 19 ext 218; Fax: +48 32 231 70 26

**Joanna Domańska**

**Sławomir Nowak**

Cover designed by Krzysztof Grochla

## ISBN: 978-83-926054-4-7

# CONTENTS

## Keynote Talks



## Network Management & Optical Networks



## Flow control and buffer management

## Traffic measurements and models



## Analytical models

## Wireless Networks

## PREFACE

The HET-NETs conferences aim to motivate fundamental theoretical and applied research into the performance modelling, analysis and engineering of evolving and converging multi-service networks of diverse technology and the next generation Internet (NGI). It's goal is to bring together the networking scientist and engineers from both academia and the industry, to share the knowledge on performance evaluation of heterogeneous networks. The conferences are concentrating mainly on the following topics:

- Traffic Modelling, Characterisation and Engineering
- Experimental Performance Validation and Measurement Platforms
- Broadband Access and TCP Performance Prediction
- Numerical, Simulation and Analytic Methodologies
- QNMs with Blocking, Switching QNMs
- Performance Modelling and Congestion Control
- Optimal Broadcasting and Multicasting Schemes
- Performance issues in Optical and Sensor Networks
- Performance Evaluation and Mobility Management in Wireless Networks
- Overlay Networks
- Wireless and Multihop Ad-Hoc Networks (MANETs)
- End-to-End Quality of Service (QoS) in Heterogeneous Networks
- Quality Feedback for Perceived Service Dependability
- Performance Related Security Mechanisms

The proceedings of Sixth International Working Conference on Performance, Modelling and Evaluation of Heterogeneous Networks HET-NETs 2010, 14 - 16 January 2010, Zakopane, Poland contain 34 contributions grouped in five parts corresponding to the conference sessions and presented in chronological order. The articles are proceeded by abstracts of 6 keynote lectures which our eminent Colleagues kindly agreed to present. These talks present the state of the art of various important performance evaluation issues.

We thank all Authors, Program Committee members and everyone involved in HET-NETs 2010 preparation.

The conference was technically sponsored by EuroNF Network of Excellence and the Institute of Theoretical and Applied Informatics of Polish Academy of Sciences.

*Tadeusz Czachórski*

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 9 – 10

# Steps towards self-aware networks

EROL GELENBE

Professor in the Dennis Gabor Chair
Imperial College London

**Abstract:** We present the design of a packet network where paths are dynamically selected based on quality of service (QoS) metrics that can be specified by the users of the network or by the network's access points. The approach uses on-line measurement associated with the traffic itself, and reinforcement learning throughout the nodes of the network, to try to satisfy the users' QoS objectives. The talk will present numerous measurement studies of this approach on a large laboratory test-bed. The seminar is based on our recent paper in the Communications of the ACM (July 2009), which integrates different aspects of our research including network design, performance evaluation models and methods, and probability models of neuronal networks.

**About the speaker:** Erol Gelenbe graduated from the Middle East Technical University (Ankara, Turkey) and was elected to professorial chairs successively at the University of Liege (Belgium) at the age of 27, then the University of Paris Sud-Orsay, the University of Paris V, Duke University, University of Central Florida, and Imperial College. He is a member of the French National Academy of Engineering, the Turkish Academy of Sciences, and of Academia Europaea. He has graduated some 50 PhD students, many of whom are prominent in France, Greece, China, Turkey, and North and South America. A Fellow of IEEE and ACM, he won the ACM SIGMETRICS Life-Time Achievement Award for his contributions to probability models of computer and network performance in 2008. He was the first Computer Scientist to win the Grand Prix France-Telecom of the French Academy of Sciences in 1996, and has received several Doctorates Honoris Causa. The President of Italy appointed him Commander of Merit and Grande Ufficiale in the Order of the Star of Italy. He is an Officer of Merit of France and has received several "honoris causa" doctorates.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 11 –12

# Mobility models for mobility management

VICENTE CASARES-GINER

Professor at Universidad Politecnica de Valencia
E. T. S. I. Telecomunicacion
Departamento de Comunicaciones

**Abstract:** The main goals of today's wireless mobile telecommunication systems is to provide both, mobility and ubiquity to mobile terminals (MTs) with a given satisfactory quality of service. By mobility we understand as the ability that the MT be connected to the network via a wireless radio telecommunication channel. By ubiquity we understand as the ability that the MT be connected to the network anytime, anywhere, regardless of the access channel's characteristics. In this talk we deal with mobility aspects. More precisely, we provide some basic backgrounds on mobility models that are being used in performance evaluation of relevant mobility management procedures, such as handover and location update. For handover, and consequently for channel holding time, we revisit the characterization of the cell residence time (also named as cell dwelling time or cell sojourn time). Then, based on those previous results, models about the location area residence time are built. Cell residence time can be seen as a micro-mobility parameter while the second can be considered as a macro-mobility parameter. Both have a significant impact on the handover and location update algorithms. Also, we overview some gravitation models, based on transportation theory, which are useful for mobility management procedures.

**About the speaker:** Dr. Vicente Casares-Giner (http://www.girba.upv.es/english.htm) obtained the Telecommunication Engineering degree in October 1974 from Escuela Técnica Superior de Ingenieros de Telecomunicación (ETSIT-UPM) de Madrid and the Ph.D. in Telecommunication Engineering in September 1980 from ETSIT-UPC de Catalunya (Barcelona). During the period from 1974 to 1983 he worked on problems related to signal processing, image restoration, and propagation aspects of radio-link systems. In the first half of 1984 he was a visiting scholar at the Royal Institute of Technology (Stockholm) dealing with digital switching and concurrent EUCLID for Stored Program Control telephone exchanges. Since then he has been involved in traffic theory applied to telecommunication systems. He has published papers in international magazines and conferences: IEEE, Electronic Letters, Signal Processing, IEEE-ICASSP, IEEE-ICC, EURASIP-EUSIPCO, International Teletraffic Conference (ITC) and its ITC Special Seminars, IC on Digital Signal Processing, Wireless Conference, ICUPC, WCNC, etc. From 1992 to 1994, and 1995, he worked in traffic and mobility models of MONET (R-2066) and ATDMA (R-2084) European RACE projects. From September 1, 1994 till august 31, 1995, Professor Casares Giner was a visiting scholar at WINLAB Rutgers University-, working with random access protocols applied to wireless environment, wireless resource management, and land mobile trunking system. During the editions of 1996, 1997 and 1998 he also participated as a lecturer in the Master of Mobile Telecommunication, sponsored by VODAFONE (Spanish branch),

where he was in charge of lecturing the mobility and traffic models for dimensioning. Professor Casares Giner also has participated in the OBANET project (2001– 2003, FP5 of the EC and other related project). He has participated in the Network of Excellence (NoE) Euro-NGI, Euro-FGI and Euro-NF within the FP6 and FP7 (http://euronf.enst.fr/en_accueil.html). Since September 1996 he is at Escuela Técnica Superior de Ingenieros de Telecomunicación (ETSIT-UPV) de Valencia. His main interest is in the area of wireless systems, in particular random access protocols, voice and data integration, systems capacity and performance on mobility management.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 13 – 14

# On the notion of max-min fairness and its applications in network design

MICHAŁ PIÓRO

Department of Data Networks and Switching
Institute of Telecommunications, Warsaw University of Technology

Department of Electrical and Information Technology
Lund University

**Abstract:** The notion of fairness is a natural means for characterizing objectives of various design problems in communication network design, For example, routing of elastic traffic in the Internet should be fair in terms of bandwidth allocated to individual traffic demands. One fairness principle that can be applied is called max-min fairness (MMF) and requires that the worst bandwidth allocation is maximized and the solution is then extended with maximization of the second worst allocation, the third one, and so on. Due to lexicographic maximization of ordered objectives, the MMF solution concept cannot be tackled by the standard optimization model, i.e., a mathematical program. However, a sequential lexicographic optimization procedure can be formulated for that purpose. The basic procedure is applicable only for convex models, thus it allows to deal only with relatively simple optimization problems but fails if practical discrete restrictions commonly arising in the communications network context are to be taken into account. Then, however, alternative sequential approaches allowing to solve non-convex MMF problems can be used. In the presentation we discuss solution algorithms for basic convex and non-convex MMF optimization problems related to fair routing and resource restoration in communications networks. The presented material is not commonly known to the community and therefore can be helpful in developing relevant optimization models for researchers working in network design.

**About the speaker:** Michał Pióro (http://ztit.tele.pw.edu.pl/en/head.html) completed his Ph.D. degree in 1979, his habilitation degree in 1990, and obtained the Polish State Professorship in 2002 (all in telecommunications). He is a full professor and Head of Department of Data Networks and Switching at the Institute of Telecommunications, Warsaw University of Technology (Poland). At the same time he is a professor at the Department of Electrical and Information Technology, Lund University (Sweden). Professor Pióro has lead many national and international research projects in telecommunications network modeling, optimization and performance analysis. He is an author of more than 150 research papers in the field. He also wrote several books, including the monograph "Routing, Flow, and Capacity Design of Communication and Computer Networks", Morgan Kaufmann Publishers (imprint of Elsevier), 2004. Prof. Pióro is a member of international research bodies and technical program committees of several major conferences. He is a technical editor of IEEE Communications Magazine.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 15–18

# Modeling and Analysis of Triple Play Services in Multimedia Internet

UDO R. KRIEGER [a] TIEN VAN DO [b] NATALIA M. MARKOVICH [c]

[a]Corresponding author's address: Otto-Friedrich University, D-96052 Bamberg, Germany, Email: udo.krieger@ieee.org

[b]Department of Telecommunications, Budapest University of Technology and Economics, Magyar tudósok körútja 2, H-1117 Budapest, Hungary

[c]Institute of Control Sciences, Russian Academy of Sciences, 117997 Moscow, Russia

## Abstract

In recent years, multimedia Web and real-time applications such as Skype, YouTube, Facebook, Zattoo, PPLive, or World-of-Warcraft supporting voice-over-IP (VoIP), video-on-demand (VoD), live-streaming of videos, IPTV and multi-party on-line game playing have created a rich set of fast growing services in multimedia Internet. They are enabled by new portals featuring Web2.0 and peer-to-peer (P2P) transport technology. Furthermore, they indicate a way towards next generation networks and substantially contribute to new triple play portfolios of competitive Internet and application service providers. The use of adaptive variable bitrate encoding schemes like iSAC or MPEG-4/AVC and the packet-based transfer of multimedia streams have raised new teletraffic issues regarding traffic characterization, traffic control and the dimensioning of the underlying server and transport infrastructure. It is the main task of such a multi-tier infrastructure comprising application, database and media servers to process efficiently the received media requests of a dynamic client population demanding popular objects from the attached databases and to return the latter quickly by streams of related multimedia messages to all requesting sources.
In our presentation we report on a joint research effort with T. van Do and N.M.

Markovich to analyze and model important components of these new multimedia service infrastructures by appropriate teletraffic techniques and new statistical methods.
Following a top-down approach of teletraffic engineering, we first study the session performance of a multi-tier service infrastructure and the access to the multimedia content using an Apache Web server as gateway. In particular, we consider the related software architecture with a non-threaded multi-processing module. Considering the used resource pool of available HTTP service processes, we propose a tractable multi-server model to approximate the performance of its load-dependent dynamic behavior. Secondly, we show how VoD and live-streaming media servers can be modelled by appropriate multi-class loss systems taking into account the selection behavior of customers and the popularity of the selected objects.
Finally, we consider the packetized transport of multimedia streams by overlay networks and present a statistical characterization of the traffic streams following an operational modeling approach. Considering measured traffic of IPTV sessions, video and voice streams and applying results of extreme-value theory, we sketch a versatile methodology for VBR traffic characterization in the presence of non-stationarity and dependence in the data which may generate burstiness, long-range dependence and heavy tailed marginal distributions of the underlying traffic characteristics. Using the bivariate process of inter-arrival times between packets and the packet lengths in combination with a bufferless fluid model, it allows us to derive the capacity required by a packet stream on a virtual link, to determine the loss characteristics of the flow and to provide indicators on the user's satisfaction.

Finally, some open issues regarding the characterization and control of multimedia traffic in future Internet are pointed out.

**References**

1. T. van Do, U.R. Krieger, R. Chakka: Performance Modeling of an Apache Web Server with a Dynamic Pool of Service Processes. Telecommunication Systems, 39(2-3), 117–129, November 2008.
2. U.R. Krieger, T. van Do: Performance Modeling of a Streaming Media Server. Proc. Fourth Euro-FGI Workshop on "New Trends in Modelling, Quantitative Methods and Measurements", Ghent, Belgium, May 31 - June 1, 2007.
3. N.M. Markovich, U.R. Krieger: Statistical Analysis and Modeling of Peer-to-Peer Multimedia Traffic. In D. Kouvatsos, ed., Performance Handbook - Next Generation Internet, Lecture Notes in Computer Science, LNCS 5233, Springer, Heidelberg, to appear 2010.

## About the speaker

**Udo Krieger** is head of the computer networks group at the Faculty Information Systems and Applied Computer Science of Otto-Friedrich University Bamberg. He has graduated from Technische Hochschule Darmstadt, Germany, receiving an M.Sc. degree in Applied Mathematics and a Ph.D. degree in Computer Science, respectively. Before joining Otto-Friedrich University in 2003 he has been working for 18 years as a senior scientist and technical project manager at the Research and Technology Center of Deutsche Telekom in Darmstadt. From 1994 until 2003 he has also been affiliated as a lecturer to the Computer Science Department of J.W. Goethe University, Frankfurt.

Udo Krieger was responsible for the participation of Deutsche Telekom in the EU projects COST 257, 279 and Eurescom P1112. From 2003 until 2008 he has served as workpackage leader of the activity "IP Traffic Characterization, Measurements and Statistical Methods" in the IST-FP6 NoEs EuroNGI/EuroFGI. Currently, he is engaged in the ESF action COST IC0703 "Data Traffic Monitoring and Analysis (TMA): theory, techniques, tools and applications for the future networks" (www.tma-portal.eu/).

Udo Krieger is a member of the editorial board of "Computer Networks" (www.elsevier.com/locate/comnet) and has been serving in numerous programme committees of international conferences dealing with Internet research including Infocom '98 and Infocom 2007. His research interests include traffic management of wired and wireless IP networks, teletraffic theory, and numerical solution methods for Markov chains.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 19 –20

# Stochastic bounds for Markov chains and how to use them for performance evaluation

JEAN-MICHEL FOURNEAU

Professor of Computer Science
University of Versailles St Quentin, France

**Abstract:** We survey several algortihms and applications of stochastic comparison of Discrete Time Markov Chains. These techniques lead to an important reduction on time and memory complexity when one is able to compare the initial rewards with rewards on smaller chains. We also show that stochastic comparison and stochastic monotonicity can provide intervals for rewards when models are not completly known (some parameters are missing). Finally we can prove some qualitative properties due to the stochastic monotonicity or the level crossing ordering or Markov Chains.

In this lecture we present some applications of stochastic comparison of Discrete Time Markov Chains. The approach differs from sample path techniques and coupling theorem applied to models as we only consider Markov chains and algorithms on stochastic matrices. These algorithms build stochastic matrices which finally provide bounds on the rewards. We consider DTMC but we also show how we can apply the method to Continuous Time Markov Chains. We consider three typical problems we have to deal with when consider a Markovian models.

The most known problem is the size of the state space. Even if the tensor representation provides an efficient manner to store the non zero entries of the transition rate matrix, it is still difficult to solve the model. We are interested in performance measures defined as reward functions on the steady-state or transient distribution or by first passage time (for instance to a faulty state in dependability modeling). Thus the numerical computation of the analysis is mainly the computation of the steady-state or transient distributions or the fundamental matrix. This is in general difficult because of the memory space and time requirements. Sometimes it is even impossible to store a vector of states. We also have a problem related to the parametrization of the model. Finding realistic parameters for a Markovian model may be a very difficult task. Sometimes we know that some transition exist but we do not know their probabilities. Instead we can find an interval of probability for all the transitions. Thus the model is associated to a set of Markov chains. Some models are also described by a set of distributions rather than a single one. Finally, we sometimes need qualitative properties rather than numerical results. For instance we may want to prove that the reward we analyze is non decreasing with a parameter. Numerical analysis does not help here.

We show in that tutorial that stochastic comparison of Markov chains may lead to an efficient solution for these three problems. While modeling high speed networks or dependable systems, it is often sufficient to satisfy the requirements for the Quality of Service we expect. Exact values of the performance measures are not necessary in this case and bounding some reward functions is often sufficient. So, we advocate the use of stochastic comparison to prove that QoS requirements are satisfied. We can derive bounds for a family of stochastic matrices. Thus we are able to give an interval

for rewards on models which are not completely specified. The stochastic monotonicity and the level crossing ordering of Markov chains allow to prove some qualitative properties of the models.

**About the speaker:** Jean-Michel Fourneau is Professor of Computer Science at the University of Versailles St Quentin, France. He was formerly with Ecole Nationale des Telecommunications, Paris and University of Paris XI Orsay as an Assistant Professor. He graduated in Statistics and Economy from Ecole Nationale de la Statistique et de l'Administation Economique, Paris and he obtained is PHD and his habilitation in Computer Science at Paris XI Orsay in 87 and 91 respectively. He is an elected member of IFIP WG7.3 the IFIP working group on performance evaluation.

He is the Head of the Performance Evaluation team within PRiSM laboratory at Versailles University and his recent research interests are algorithmic performance evaluation, Stochastic Automata Networks, G-networks, stochastic bounds, and application to high speed networks, and all optical networks.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 21 – 22

# On the generalisation of the Zipf-Mandelbrot distribution and its application to the study of queues with heavy tails

DEMETRES KOUVATSOS

Networks and Performance Engineering Research
School of Computing, Informatics and Media
University of Bradford
Bradford BD7 1DP West Yorkshire, UK
*D.Kouvatsos@Bradford.ac.uk*

**Abstract:** Many studies of Internet traffic revealed that it is characterised by burstiness, self-similarity (SS) and/or long range dependence (LRD) that can be attributed to the long tails of various distributions with power law behaviour, such as those associated with interevent times and queue lengths. These traffic patterns have adverse impact on network's operation leading to queues and congestion and, thus, are of significant importance towards the prediction of performance degradation and the capacity planning of the network. Long–tailed distributions have been employed to generate traffic flows in simulation studies, which, however, tend to be rather inflexible and computationally expensive. Thus, there is a need to devise new and cost-effective analytic methodologies for the credible assessment of the effects of long tailness in network traffic patterns.

The talk will describe a novel analytic framework, based on the optimisation of generalised entropy measures, such as those proposed in the fields of Information Theory, Statistical Physics and Quantification Theory, subject to appropriate mean value system constraints. In this context, a generalisation of the Zipf-Mandelbrot (Z-M) distribution will be devised and interpreted as the steady-state probability distribution of queues with long tails and power law behaviour. Moreover, related analytic algorithms will be devised and typical numerical tests will be presented to assess the credibility of the generalised Z-B queue length distribution and the adverse impact of SS and LRD traffic flows on queue performance.

**About the speaker:** Prof. Demetres Kouvatsos is the Head of NetPEn, Networks & Performance Engineering Research, University of Bradford, UK and a Visiting Professor at CTIF - Center for TeleInFrastruktur, University of Aalborg, Denmark. He pioneered new and cost-effective analytic methodologies, based on queueing, information and graph theoretic concepts, for the approximate analysis of arbitrary queueing network models (QNMs) and their performance modelling applications into the multiservice networks of diverse technology. Professor Kouvatsos acted as the chair of seven IFIP Working Conferences on the 'Performance Modelling and Evaluation of ATM & IP Networks' (ATM & IP 1993-1998, 2000). More recently, under the auspices of the EU Network of Excellence Euro-NGI, he acted as the Chair of four HET-NETs 2003-2006 International Working Conferences on the 'Performance Modelling and

Evaluation of Heterogeneous Networks' and also directed PERFORM-QNMs - the first EU NoE Euro-NGI PhD Course in 'Performance Engineering and Queueing Network Models' (Sept. '06). Some of his latest scholarly activities include the editing and publication of three research volumes by River Publishers (July '09) focusing on traffic engineering, performance analysis, mobility management and quality-of-service of heterogeneous networks. Moreover, he is currently editing a performance handbook to be published by Springer, based on tutorial papers concerning the 'Next generation Internet: Performance Evaluation and Applications'. Prof. Kouvatsos is a Recipient of the IFIP Silver Core Award (1997) and served as a member of the Jury of international experts for the Board of Trustees of the Research Foundation – Flanders (FWO-Vlaanderen), Research Council, Belgium (2004). His professional associations include memberships with the EPSRC College, UK and the IFIP Working Groups IFIP WG6.2 on 'Network and Inter-network Architectures' and IFIP WG 6.3 on the 'Performance of Computer Networks'.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 23–34

# On the Use of *bzip2* for Network Anomaly Detection

CHRISTIAN CALLEGARI STEFANO GIORDANO
MICHELE PAGANO TERESA PEPE

Dept. of Information Engineering,
University of Pisa, ITALY
{*c.callegari,s.giordano,m.pagano,t.pepe*}*@iet.unipi.it*

**Abstract:** In the last few years, the number and impact of security attacks over the Internet have been continuously increasing. Since it seems impossible to guarantee complete protection to a system by means of the "classical" prevention mechanisms, the use of Intrusion Detection Systems has emerged as a key element in network security. In this paper we address the problem considering the use of bzip2, a well-known compression algorithm, for detecting anomalies in the network traffic running over TCP. The proposed method is based on the consideration that the entropy represents a lower bound to the compression rate that we can obtain, and the presence of anomalies should affect the entropy of the analyzed sequences.

**Keywords:** The performance analysis, presented in this paper, demonstrates the effectiveness of the proposed method.

## 1. Introduction

In the last few years Internet has experienced an explosive growth. Along with the wide proliferation of new services, the quantity and impact of attacks have been continuously increasing. The number of computer systems and their vulnerabilities have been rising, while the level of sophistication and knowledge required to carry out an attack have been decreasing, as much technical attack know-how is readily available on Web sites all over the world.

Recent advances in encryption, public key exchange, digital signature, and the development of related standards have set a foundation for network security. However, security on a network goes beyond these issues. Indeed it must include security of computer systems and networks, at all levels, top to bottom.

Since it seems impossible to guarantee complete protection to a system by means of prevention mechanisms (e.g., authentication techniques and data encryption), the use of an Intrusion Detection System (IDS) is of primary importance to reveal intrusions in a network or in a system.

State of the art in the field of intrusion detection is mostly represented by misuse based IDSs. Considering that most attacks are realized with known tools, available on the Internet, a signature based IDS could seem a good solution. Nevertheless hackers continuously come up with new ideas for the attacks, that a misuse based IDS is not able to block.

This is the main reason why our work focuses on the development of an anomaly based IDS. In particular our goal is to reveal intrusions carried out exploiting TCP bugs. To this purpose in [1] we proposed a novel method, based on the use of a compression algorithm to "model" the normal behavior of the TCP connections. The method is based on the consideration that the entropy represents a lower bound to the compression rate that we can obtain, and that the more redundant the data are the better we can compress them.

Indeed, if we append the observed sequence to the training sequence, and we compress the whole sequence, the compression rate that will be higher when the observed sequence is more "similar" to the training sequence; vice-versa the compression rate is low when the observed sequence is anomalous.

These observations, which represent the starting point for our proposed methods, are also at the base of several methods which aim at automatically finding out in which language a given plain text is written. Examples of such methods are provided in [2] and [3].

This paper somehow extends a previous work of the authors [1], by taking into account a new compression algorithm, namely bzip2, which is considered to be much more efficient than those considered in their previous work [4].

The remainder of this paper is organized as follows: section II describes the bzip2 compression algorithm, while section III provides a description of the implemented system, detailing both the training phase and the detection phase. Then in section IV we provide the experimental results and finally section V concludes the paper with some final remarks.

## 2. Compression algorithms

In computer science and information theory, data compression is the process of encoding information using fewer bits (or other information-bearing units) than an unencoded representation would require through use of specific encoding schemes, without having any information loss. Lossless compression is possible because

most real-world data have statistical redundancy.

Before analyzing in more detail the compression algorithm used in this work, we briefly present some elements of the information theory, which will help the understanding of the implemented systems.

Information theory [5][6] aims at defining, in a rigorous way, the notion of information and a way to measure it. Intuitively, the information can be considered as a reduction of the quantity of uncertainty; it is obvious that the information do not necessarily coincide with the quantity of transmitted data. Indeed, if the latter have a lot of redundancy, the information will be less than in the case of non-redundant data. The sentence "it is going to rain" will differ if we are talking about a desertic region or about Ireland. Indeed, in the first case we will have a big quantity of information, since it is a rare event, while in the second case the quantity of information will be lower since it is a more common event.

One of the most important elements of the information theory is the notion of Entropy, introduced by C.E. Shannon in the late 40s [7][8].

The entropy $H$ of a discrete random variable $X$ is a measure of the amount of uncertainty associated with the value of $X$.

The entropy is measured in bit (BInary digiT), thus a bit is the fundamental quantum of information, necessary to distinguish between two equiprobable hypotheses. Referring to an alphabet composed of $n$ distinct symbols, respectively associated to a probability $p_i$, the mean information taken by each symbol is

$$H = -\sum_{i=1}^{n} p_i \cdot log_2 p_i \text{ bit/symbol} \tag{1}$$

which is usually referred to as single-symbol entropy or alphabet entropy. This quantity obviously decreases if we consider the correlation among the alphabet symbols.

It is worth noticing that $H$, being a measure of the uncertainty, in case of absence of uncertainty ($p_1 = 1, p_i = 0\ i = 2,3,\cdots,n$) is:

$$H = H_{min} = 0 \tag{2}$$

while in case of equiprobable symbols ($p_i = 1/n\ i = 1,2,\cdots,n$) we have:

$$H = H_{MAX} = -\sum_{i=1}^{n} 1/n \cdot log_2 1/n = log_2 n \tag{3}$$

Lossless compression algorithms are usually classified into three main categories:

- Model based algorithms: each symbol or group of symbols is encoded with a variable length code, according to some probability distribution. The efficiency of such algorithms depends on the choice of an appropriate probability model and on the way such model is used. According to this last characteristic, the algorithms are divided into:
  - Static coders (e.g., Morse code)
  - Semi-adaptive coders: they build the translation table starting from the data; the table has to be sent together with the compressed data (e.g., static Huffman Coding)
  - Dynamic coders: the translation table is directly built during both the encoding/decoding phase, thus it has not to be sent (e.g., DMC)
- Dictionary based algorithms: they are based on the use of a dictionary, which can be static or dynamic, and they code each symbol or group of symbols with an element of the dictionary (e.g., LZW)
- Block-sorting algorithms: the basic idea is to perform a transformation of the data, so as to obtain a format which can be easily compressed

In the following we detail the bzip2 algorithm that has been used in the implemented system. Since it is one of the most efficient compression algorithms, the compression ratio should lead to a good estimate of the real entropy of data.

### 2.1. Bzip2

Bzip2 is an open source, patent free, high-quality data compressor [4]. Indeed, the compression rate that it achieves is generally considerably better than that achieved by more conventional LZ77/LZ78-based compressors, but, on the other hand, it is also much more complex and considerably slower [4].

Bzip2 uses a combination of techniques to compress data in a lossless way. In more detail, data compression with bzip2 involves three reversible transformations:

- Burrows-Wheeler transform
- Move to Front transform
- Huffman coding

In practice, an input file is divided into fixed sized block that are compressed independently, by applying the three transformations in the given order.

In the following we will focus on the three algorithms, providing for each of them a brief description (for a more detailed discussion we report to the original papers).

### 2.1.1. Burrows-Wheeler transform

The Burrows-Wheeler transform (BWT) [9] is a block-sorting lossless transformation. BWT does not perform any compression, but only modifies the data so as to simplify the compression, performed in the other two phases (Move to Front coding and then Huffman). Indeed, the transformed block contains exactly the same characters as the original block, but in a form that is easier to compress.

In more detail, given a string $S$ of $n$ characters, the algorithm works as follows:

- transform $S$ in $n$ new strings $S_1, S_2, \cdots S_n$ that are $n$ rotations (cyclic shifts) of $S$
- sort the $n$ strings lexicographically
- extract the last character of each rotation

Then, the output is a string $L$ where the $i-th$ character is the last character of the $i-th$ sorted rotation. To each character is also associated an index $I$ that is the position of the character in the original string $S$.

To be noted that, $L$ and $I$ are sufficient to compute the original string $S$ during the decompression phase.

### 2.1.2. Move to Front transform

Move to Front transform (MTF) [10] is an algorithm that allows the reduction of the redundancy of data so as to improve the compression algorithm performance. To be noted that also this algorithm does not perform any compression of the data.

MTF works by using a coding table, that is a list of all possible characters given in a specific order, typically ascending order. This table does not remain constant in the coding process but is updated at each step.

In more detail, the algorithm works as follows. When a new symbol has to be processed

- the encoder outputs a number that is the position of the symbol in the coding table
- the coding table is modified by moving the symbol on top of the coding table

Thus, the final output is a series of numbers that denotes the position of the original characters in the continuously evolving coding table.

It is worth noticing that the decoding phase starts with the original table, which evolves again in a similar manner.

### 2.1.3. Huffman coding

Huffman coding [11] (developed by David A. Huffman while he was a Ph.D. student at MIT) is based on the use of a variable-length code table for encoding a source symbol, where the variable-length code table has been derived from a binary tree built from the estimated probability of occurrence for each possible value of the symbol source.

Huffman coding uses a specific method for choosing the representation for each symbol, resulting in a prefix-free code[1] that expresses the most common characters using shorter strings of bits.

Huffman coder is the most efficient compression method of this type: no other mapping of individual source symbols to unique strings of bits will produce a smaller average output size when the actual symbol frequencies agree with those used to create the code. Because of this Huffman coding belongs to the category of the "minimum-redundancy coders" or "optimal coders".

The algorithm works by creating a binary tree of nodes. First of all, it is necessary to estimate the occurrence frequency of each symbol in the data to encode, then the tree is built in the following way:

- the tree contains as many leaves as there are symbols
- all leaf nodes are sorted in increasing order according to the corresponding probabilities
- while there is no root node (not all the nodes are "linked")
  - link the two nodes with the lowest weight and create a new node with probability equal to the sum of the two symbol probabilities
- the remaining node is the root node; the tree has now been generated

Then, iteratively starting from the root node, each right branch is labelled with a "1" and each left branch with a "0". To generate the compressed data, each symbol is replaced by the corresponding string, obtained by consecutively reading the branch labels from the root to the leaf node (corresponding to the symbol).

Figure 1 shows the Huffman tree for a source with 4 symbols (a1, a2, a3, and a4) respectively characterized by the following occurency frequencies 0.4, 0.35, 0.2, and 0.05.

It is worth noticing that the compression rate, obtained by the algorithm, is low if the symbols are almost equiprobable, and is high if there are big differences among the symbols appearance probabilities.

[1] The bit string representing some particular symbol is never a prefix of the bit string representing any other symbol

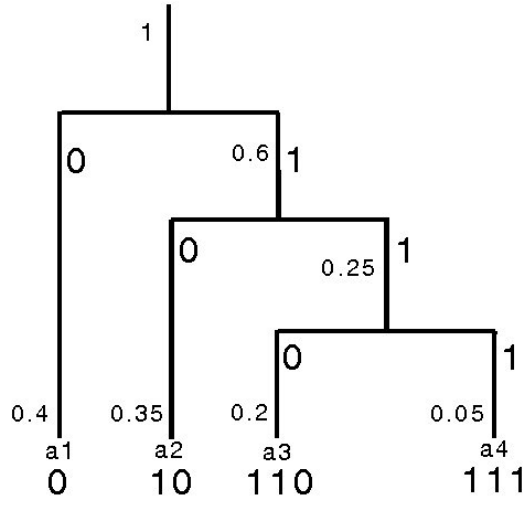



Fig. 1. Huffman tree

## 3. System architecture

In this section we detail how the implemented system works: the next subsection describes the system input, while the subsequent one focuses on the description of the training and detection phase of the system.

### 3.1. System input

The system input is given by raw traffic traces in libpcap format [12], the standard used by publicly available packet sniffer software, as Tcpdump [13] or Wireshark [14][15]. First of all, the IDS performs a filtering phase so that only TCP packets are passed as input to the detection block.

The IDS only considers some fields of the packet headers, more precisely the 5-tuple (source and destination addresses, source and destination ports, and protocol) plus the TCP flags. The 5-tuple is used to identify a connection, while the value of the flags is used to build the "profile". Experimental results have shown that the stochastic models associated to different applications strongly differ one from the other. Thus, before constructing the model, the system isolates the different services, on the basis of the server port number, and the following procedure is realized once for each port number.

A value $s_i$ is associated to each packet, according to the configuration of the TCP flags:

$$s_i = SYN + 2 \cdot ACK + 4 \cdot PSH + 8 \cdot RST + 16 \cdot URG + 32 \cdot FIN \quad (4)$$

Thus each "mono-directional" connection is represented by a sequence of symbols $s_i$, which are integers in $\{0, 1, \cdots, 63\}$.

The sequences obtained during this "filtering" phase represent the system input for both the training and the detection phase.

### 3.2. Training phase

The bzip2 compression algorithm has been modified so that the "learning phase" is stopped as the training phase is over. In this way the detection phase will be performed with a compression scheme that is "optimal" for the training data and that could be suboptimal for the detection phase, especially in case of anomalous connections.

In the following we will refer to the TCP flags sequence extracted from the whole training dataset (of a single port number) as $A$, while $b$ will be the flags sequence of a single connection (in the detection phase) and finally $B$ will be the flags sequence extracted from the whole detection dataset.

### 3.3. Detection phase

During the training phase the considered algorithm implicitly builds a model of the analyzed data. For our purpose, this model can be considered as the profile of the "normal" behavior of the network. Thus, the system, given an observed sequence $(c_1, c_2, \cdots, c_T)$, has to decide between the two hypotheses:

$$\begin{aligned} H_0 &: \{(c_1, c_2, \cdots, c_T) \sim computed\ model\} \\ H_1 &: \{anomaly\} \end{aligned} \quad (5)$$

The problem is to choose between a single hypothesis $H_0$, which is associated to the estimated stochastic model, and the composite hypothesis $H_1$, which represents all the other possibilities.

In more detail, after the training phase has been performed, the system appends each distinct connection $b$, of the detection dataset $B$, to $A$ and for each of them

decides if there is an anomaly or not, by computing the "compression rate per symbol":

$$\frac{dim([A|b]^*) - dim([A]^*)}{Length(b)} \tag{6}$$

where $[X]^*$ represents the compressed version of $X$, as the anomaly score (AS) for the proposed system

Finally, the presence of an anomaly is decided on the basis of a threshold mechanism, where the threshold is set as a tradeoff between detection rate and false alarm rate.

## 4. Experimental results

In this section we discuss the performance achieved by the system when using the proposed system over the DARPA data set.

The DARPA evaluation project [16][17][18] represents the first formal, repeatable, and statistically-significant evaluation of IDSs and is considered the reference benchmark for testing this kind of systems.

Such project was carried out in 1998 and 1999, and the results shown in this paper have been obtained with the 1999 data set. This data set consists of 5 weeks of traffic, collected over a network composed by about 40 computers.

The first and the third weeks do not contain any attack, thus they are suitable to perform the training of anomaly detection systems. The second week of traffic contains some labelled attacks and can be used to analyse how they modify network behavior. Finally traffic from the fourth and the fifth weeks are the *test data*. There are 201 instances of 56 types of attacks distributed throughout these two weeks. Information about the attack instances (e.g. where they are located in week 4 and 5 data) is found in the 1999 "Attack Truth list", which is provided along with the data set. Moreover two distinct data sets are provided, respectively collected on the external and the internal side of the LAN gateway. In this work we have considered the latter, where the total number of attack instances is 177.

In more detail, we have taken into account the traffic related to the following port number:

- TCP 21 (FTP)
- TCP 22 (SSH)
- TCP 23 (Telnet)

To be noted that, even though in general it is not true, in the DARPA data-set we can trust the port number for classifying the traffic.

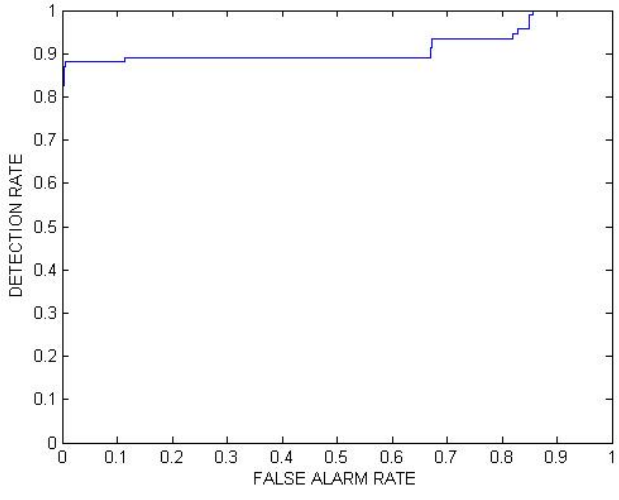



Fig. 2. ROC curve obtained using FTP-data traffic

As a performance parameter we consider the ROC curve, which plots the detection rates versus the false alarm rate. Figure 2 we present the ROC curve for the FTP-data traffic.
As it can be seen from the graph, the proposed system achieves very good performance revealing the 88.6% of the total attacks with a rate of false alarm limited to the 0.5%. To be noted that if we want to reveal more attacks (up to the 95%) the false alarm rate increases up to the 68%.

As far as SSH and telnet traffic are concerned, performance are even better, as we can see from Figure 3 and 4, respectively. In both cases we can reveal more than the 99% of the attacks with a false alarm rate lower than 1%.

As a general observation we can state that the proposed system is able to achieve excellent performance with very low false alarm rates.

## 5. Conclusions

In this paper we have presented a novel anomaly detection method, which detects anomalies by means of a statistical characterization of the TCP traffic, by using the bzip2 compression algorithm, to estimate the entropy of the traffic data.

To assess the validity of the proposed solution, we have tested the system over the 1999 DARPA dataset, which represents the *standard de facto* for IDS evaluation. The performance analysis has highlighted that the implemented system obtain very good results.

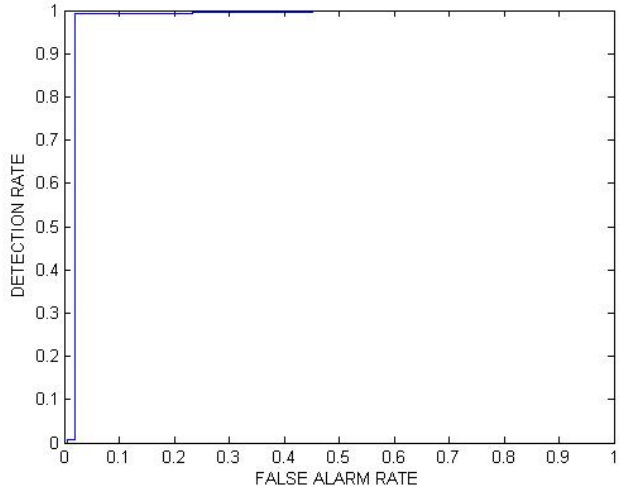



Fig. 3. ROC curve obtained using SSH-data traffic

## References

[1] C. Callegari, S. Giordano, and M. Pagano, "On the use of compression algorithms for network anomaly detection," in *Proc. of the International Conference on Communications (ICC)*, 2009.

[2] D. Benedetto, E. Caglioti, and V. Loreto, "Language trees and zipping," *Physical Review Letters*, vol. 88, January 2002.

[3] A. Puglisi, "Data compression and learning in time sequences analysis," 2002.

[4] "Bzip2." http://www.bzip.org/ (accessed on 2009/11/24).

[5] T. Cover and J. Thomas, *Elements of information theory*. USA: Wiley-Interscience, 2nd ed., 2006.

[6] R. Gallager, *Information Theory and Reliable Communication*. USA: Wiley, 1968.

[7] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.

[8] C. Shannon and W. Weaver, *A Mathematical Theory of Communication*. USA: University of Illinois Press, 1963.

[9] B. Balkenhol and S. Kurtz, "Universal data compression based on the burrows-wheeler transformation: Theory and practice," *IEEE Transactions on Computers*, vol. 49, no. 10, pp. 1043–1053, 2000.

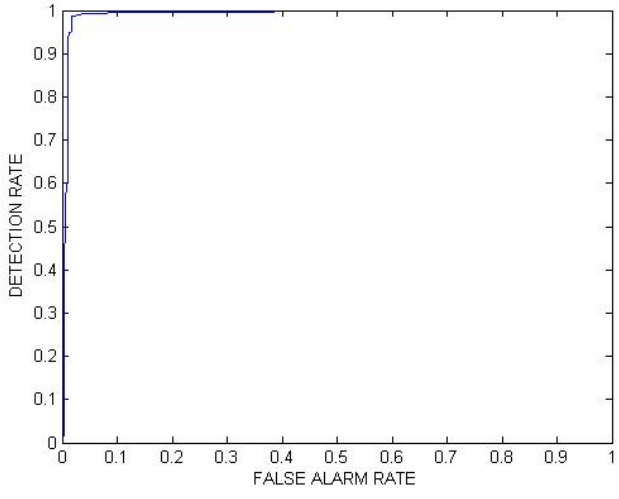



Fig. 4. ROC curve obtained using Telnet-data traffic

[10] J. L. Bentley, D. D. Sleator, R. E. Tarjan, and V. K. Wei, "A locally adaptive data compression scheme," *Commun. ACM*, vol. 29, no. 4, pp. 320–330, 1986.

[11] D. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the Institute of Radio Engineers*, vol. 40, no. 9, 1952.

[12] "pcap (format)." http://imdc.datcat.org/format/1-002W-D=pcap (accessed on 2009/11/24).

[13] "Tcpdump." http://www.tcpdump.org/ (accessed on 2009/11/24).

[14] "Wireshark." http://www.wireshark.org/ (accessed on 2009/11/24).

[15] A. Orebaugh, G. Ramirez, J. Burke, and L. Pesce, *Wireshark & Ethereal Network Protocol Analyzer Toolkit (Jay Beale's Open Source Security)*. USA: Syngress Publishing, 2006.

[16] "MIT, Lincoln laboratory, DARPA evaluation intrusion detection." http://www.ll.mit.edu/IST/ideval/ (accessed on 2009/11/24).

[17] R. Lippmann, J. Haines, D. Fried, J. Korba, and K. Das, "The 1999 DARPA off-line intrusion detection evaluation," *Computer Networks*, vol. 34, no. 4, pp. 579–595, 2000.

[18] J. Haines, R. Lippmann, D. Fried, E. Tran, S. Boswell, and M. Zissman, "1999 DARPA intrusion detection system evaluation: Design and procedures," Tech. Rep. 1062, MIT Lincoln Laboratory, 2001.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 35–48

# QoS-aware infrastructure resources allocation in systems based on service-oriented architecture paradigm

ADAM GRZECH [a] PIOTR RYGIELSKI [a] PAWEŁ ŚWIĄTEK [a]

[a]Institute of Computer Science
Wroclaw University of Technology, Poland
{adam.grzech, piotr.rygielski, pawel.swiatek}@pwr.wroc.pl

**Abstract:** In this paper the task of communication and computational resources allocation in systems based on SOA paradigm is considered. The task of resources allocation consists in assigning resources to each incoming service request in such a way, that required level of the quality of service is met. Complex services performance time in distributed environment is assumed as the quality of service measure. Models of the system and analysis of service response time presented in this paper allowed to formulate several tasks of resource allocation in terms of well known quality of service assurance models: *best effort*, *IntServ* and *DiffServ*. For each formulated task solution algorithms were proposed and their correctness was evaluated by means of simulation.

**Keywords:** quality of service (QoS), service-oriented architecture (SOA), resource allocation, response time guaranties

## 1. Introduction

Resource allocation and quality of service management in systems based on service-oriented architecture (SOA) paradigm are very important tasks, which allow to maximize satisfaction of clients and profits of service provider [1]. In nowadays SOA systems, which utilize Internet as the communication bus the problem of service response time guaranties arises. Since overall service response time consists of communication and computational delays the task of delivering requested service response time requires proper management of both communication and computational resources [4].

In this work three methods for satisfying quality of service requirements based on well known quality of service assurance models (i.e.: *best effort*, Integrated Services (*IntServ*) and Differentiated Services (*DiffServ*)) [8] are presented.

Paper is organized as follows. In section 2 models of system, complex service and network traffic generated service requests are presented. Section 3 covers qualitative analysis of service response time in the considered system. Basing on assumed models and performed analysis three tasks and solution algorithms for resource allocation for the purpose of quality of service assurance are presented in section 4. Exemplary results of performed simulations are presented in section 5. Finally conclusions are drawn and directions for future research are given in section 6.

## 2. Model of the system

It is assumed that the considered system delivers complex services composed of atomic services; the latter is defined as a service with an indivisible functionality offered by known and well-specified place or places in the system. Moreover, it is also assumed that each atomic service is available in several versions; different versions of the particular atomic service offer exactly the same functionality and are differentiated by values of various attributes assign to each atomic service [4].

### 2.1. Complex service composition

Let us assume that $AS$ is the set of all atomic services and contains $m$ separate subsets $AS_i$ ($i = 1, \ldots, m$); $AS = \{AS_1, \ldots, AS_m\}$. Subset $AS_i$ contains all already existing and ordered versions of the particular $i$-th atomic service $as_{ij}$; $AS_i = \{as_{i1}, \ldots, as_{in_i}\}$ where $n_i$ is the number of all distinguishable versions of the $i$-th atomic service. Different versions of all $i$-th atomic services $as_{ij_i}$ ($j_i = 1, \ldots, n_i$) may be labeled using values of many different attributes (e.g. location within the system, attached security mechanisms, frequency of use, computational complexity, completing time, quality of interface, etc.).

Let us also assume that atomic services from set $AS$ are used to complete complex services $s_k$ ($s_k \in S, k = 1, \ldots, K$) in such a way, that each $k$-th complex service ($k$-th path through the serially ordered set of atomic services sets) is composed of exactly $m$ different atomic services $as_{ij_i}$ executed one-by-one following increasing values of indexes $i$ ($i = 1, \ldots, m$). Each complex service path $s_k$ is precisely defined by a sequence of indices $j_i$ of particular versions of atomic services used by complex service $s_k$:

$$s_k = (j_1, \ldots, j_m). \tag{1}$$

The set $S$ of all possible complex services is defined as:

$$S = \{(j_1, \ldots, j_m) : j_1 \in \{1, \ldots, n_1\}, \ldots, j_m \in \{1, \ldots, n_m\}\}. \tag{2}$$

Such defined set of atomic services as well as assumed way of complex service composition means that the available set of $m$ atomic services in various versions allows to obtain $K$ different complex services where $K = n_1 \times \ldots \times n_m$ (see fig. 1).

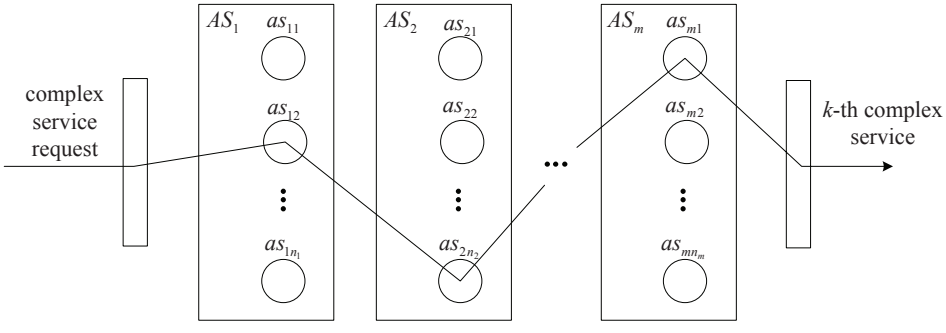



Fig. 1. Complex service composed of serially executed atomic services.

### 2.2. Network traffic

In the SOA paradigm it is assumed, that atomic services providing certain functionalities are performed in a distributed environment [6]. Let $req_l$ $(l = 1, 2, \ldots)$ represents certain $l$-th complex service request incoming to the system. In order to deliver requested complex functionality service request $req_l$ has to pass through all $m$ atomic services along certain service path $s_k$. Obviously atomic services belonging to chosen service path may be placed in remote locations. Choosing certain service path $s_k$ for service request $req_l$ means that request $req_l$ needs to travel from service requester $SR$ through each link and atomic service on service path $s_k$ and back to service requester $SR$ (see fig. 2).

Denote by $c_{(i-1)j_{i-1}j_i}$ $(i = 1, \ldots, m, j_{i-1} = 1, \ldots, n_{i-1}, j_i = 1, \ldots, n_i)$ the capacity of a communication link connecting two consecutive atomic services $as_{(i-1)j_{i-1}}$ and $as_{ij_i}$. Parameters $c_{01j_1}$ $(j_1 = 1, \ldots, n_1)$ and $c_{mj_m1}$ $(j_m = 1, \ldots, n_m)$ denote respectively: capacities of links between service requester $SR$ and atomic services in the first stage and atomic services in last ($m$-th) stage and service requester $SR$.

Execution of each atomic service changes the size of service request in proportional manner:

$$u_{out} = \alpha_{ki} \cdot u_{in}, \tag{3}$$

where $u_{in}$ and $u_{out}$ denote input and output size of service request and proportional coefficient $\alpha_{ki}$ depends on the service path $s_k$ $(k = 1, \ldots, K)$ of service request

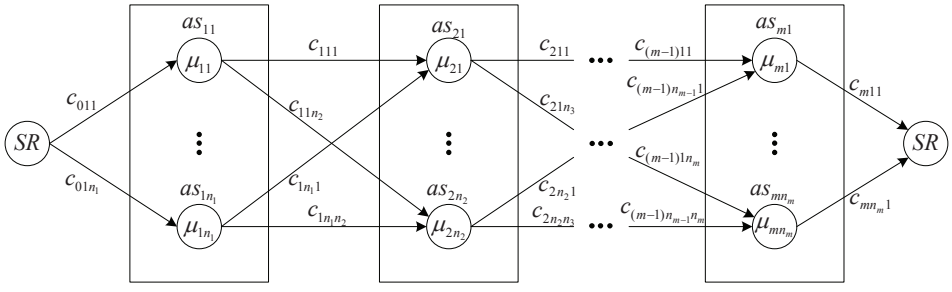



Fig. 2. Atomic services in distributed environment.

and service request processing stage $AS_i$ ($i = 1, \ldots, m$). Input and output size of request can be interpreted as the amount of input and output data necessary to execute and being a result of execution of an atomic service on certain service path. Therefore, each service path $s_k$ is described by vector $\boldsymbol{\alpha}_k = [\alpha_{k1} \ldots \alpha_{km}]^T$ of proportional coefficients concerning size of data generated by atomic services along this path.

Let $p_k$ ($k = 1, \ldots, K$) denote the probability, that certain service request $req_l$ is served along $k$-th service path $s_k$. Moreover, denote by $\overline{u}$ average size of incoming requests, by $\lambda$ average request arrival rate, and by $\mu_{ij}$ average service rate of each atomic service $as_{ij}$.

Average of traffic $f_{(i-1)j_{i-1}j_i}$ flowing between two consecutive atomic services $as_{(i-1)j_{i-1}}$ and $as_{ij}$ is a sum of traffic generated on service paths passing through atomic services $as_{(i-1)j_{i-1}}$ and $as_{ij}$:

$$f_{(i-1)j_{i-1}j_i} = \lambda\overline{u} \sum_{k \in K_{(i-1)j_{i-1}j_i}} p_k \prod_{n=1}^{i-1} \alpha_{nk}, \tag{4}$$

where $K_{(i-1)j_{i-1}j_i}$ is a set of indices of service paths passing through atomic services $as_{(i-1)j_{i-1}}$ and $as_{ij}$ and is defined as:

$$K_{(i-1)j_{i-1}j_i} = \{k \in K : as_{(i-1)j_{i-1}}, as_{ij_i} \in s_k\}. \tag{5}$$

Average traffic incoming to certain atomic service $as_{ij_i}$ is a sum of traffic on links incoming to $as_{ij_i}$:

$$f_{ij_i} = \sum_{j_{i-1}=1}^{n_{i-1}} f_{(i-1)j_{i-1}j_i} = \lambda\overline{u} \sum_{j_{i-1}=1}^{n_{i-1}} \sum_{k \in K_{(i-1)j_{i-1}j_i}} p_k \prod_{n=1}^{i-1} \alpha_{nk}, \tag{6}$$

Average size of traffic $f_{k^*}$ flowing through each $k^*$-th service path can be calculated as a sum of traffic sizes flowing between consecutive atomic services $as_{ij_i}$ $(i = 1, \ldots, m)$ along $k^*$-th path:

$$f_{k^*} = \sum_{i=1}^{m+1} f_{(i-1)j_{i-1}j_i} = \lambda \overline{u} \sum_{i=1}^{m+1} \sum_{k \in K_{(i-1)s_{k^*}(i-1)s_{k^*}(i)}} p_k \left(1 + \prod_{n=1}^{i-1} \alpha_{nk}\right), \quad (7)$$

where $s_{k^*}(i)$ denotes index $j_i$ of atomic service on $i$-th stage along path $s_{k^*}$.

An exemplary system, which consists of two atomic services ($m = 2$), each having two versions ($n_1 = n_2 = 2$) is presented on figure 3. All four possible service paths are enumerated as follows: $s_1 = (as_{11}, as_{21})$, $s_2 = (as_{11}, as_{22})$, $s_1 = (as_{12}, as_{21})$, $s_1 = (as_{12}, as_{22})$. Amount of traffic flowing through each link and each service path is presented on figure 3.

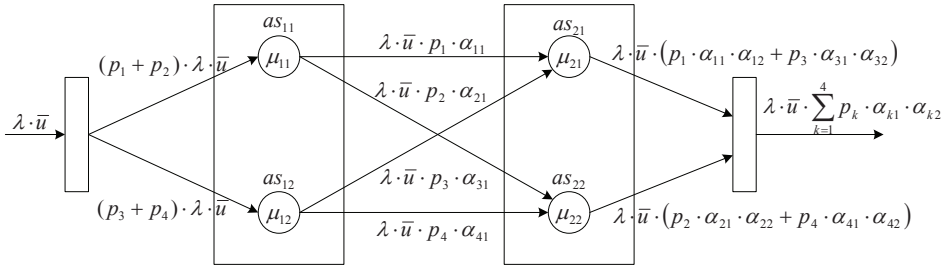



Fig. 3. Amount traffic flowing through considered system.

## 3. Service request response time

Each of atomic services $as_{ij_i}$ $(i = 1, \ldots, m, j_i = 1, \ldots, n_i)$ and communication links $c_{(i-1)j_{i-1}j_i}$ $(i = 1, \ldots, m, j_{i-1} = 1, \ldots, n_{i-1}, j_i = 1, \ldots, n_i)$ between atomic services $as_{(i-1)j_{i-1}}$ and $as_{ij_i}$ can be treated as single queue single processor queuing systems with average service rate $\mu_{ij_i}$ and $c_{(i-1)j_{i-1}j_i}$ respectively. Assuming, that incoming stream of requests is a Poisson stream and that atomic services and links service rates are characterized by exponential distributions with respective parameters $\mu_{ij_i}$ and $c_{(i-1)j_{i-1}j_i}$, average response times of atomic services and links are given by [3]:

$$\overline{d}_{ij_i} = (\mu_{ij_i} - f_{ij_i})^{-1} \quad (8)$$

and

$$\widehat{d}_{(i-1)j_{i-1}j_i} = (c_{(i-1)j_{i-1}j_i} - f_{(i-1)j_{i-1}j_i})^{-1} \quad (9)$$

respectively, where $f_{ij_i}$ is the average intensity of traffic incoming to atomic service $as_{ij_i}$ (eq. 6) and $f_{(i-1)j_{i-1}j_i}$ is the average traffic intensity on the link connecting atomic services $as_{(i-1)j_{i-1}}$ and $as_{ij_i}$ (eq. 4). It can be shown (e.g. [2, 3]) that under assumption of Poisson request arrivals and exponential service rates probability distributions of atomic services and links response times are exponential with parameters $\overline{d}_{ij_i}$ and $\widehat{d}_{(i-1)j_{i-1}j_i}$ respectively.

Denote by $r_k$ random variable representing response time of certain complex service request $req_l$ serviced along $k$-th service path $s_k$. Random variable $r_k$ is a sum of random variables representing response times of atomic services $as_{ij_i}$ ($\overline{r}_{ij_i}$) and links $c_{(i-1)j_{i-1}j_i}$ ($\widehat{r}_{(i-1)j_{i-1}j_i}$) belonging to service path $s_k$:

$$r_k = \sum_{i=1}^{m} \overline{r}_{ij_i} + \sum_{i=1}^{m+1} \widehat{r}_{(i-1)j_{i-1}j_i}. \tag{10}$$

Denote by $\boldsymbol{\delta}_k = [\delta_{k1} \dots \delta_{k2m+1}]$ vector of parameters $\overline{d}_{ij_i}$ and $\widehat{d}_{(i-1)j_{i-1}j_i}$ of probability distributions of random variables $\overline{r}_{ij_i}$ and $\widehat{r}_{(i-1)j_{i-1}j_i}$ such that $\delta_{k1} = \overline{d}_{1j_1}, \dots, \delta_{km} = \overline{d}_{mj_m}, \delta_{km+1} = \widehat{d}_{(0)j_0j_1}, \dots, \delta_{k2m+1} = \widehat{d}_{(m)j_mj_{m+1}}$. Since $\overline{r}_{ij_i}$ and $\widehat{r}_{(i-1)j_{i-1}j_i}$ are exponentially distributed with different parameters, probability distribution of $r_k$ is given as [2]:

$$f_{rk}(r_k) = \left[\prod_{i=1}^{2m+1} \delta_{ki}\right] \sum_{i=1}^{2m+1} \frac{e^{-\delta_{ki} r_k}}{\prod_{j \neq i} (\delta_{kj} - \delta_{ki})}, r_k > 0 \tag{11}$$

Cumulative distribution function of complex service response time is given by integral:

$$F_{rk}(r_k) = \int_0^{r_k} f_{rk}(x)dx = \left[\prod_{i=1}^{2m+1} \delta_{ki}\right] \sum_{i=1}^{2m+1} \frac{1 - e^{-\delta_{ki} r_k}}{\delta_{ki} \prod_{j \neq i} (\delta_{kj} - \delta_{ki})}. \tag{12}$$

Functions $f_{rk}(r_k)$ and $F_{rk}(r_k)$ denote respectively probability and cumulative distribution functions of complex service response time $r_k$ for requests served along $k$-th service path $s_k$.

## 4. Task of resource allocation

Models presented in section 2 and service response time analysis presented in section 3 allows to formulate various resource allocation tasks, the aim of which is to deliver required level of the quality of services (QoS) measured as complex service response time. In this paper we focus on three basic tasks: task of minimization of the average service response time, task of delivering required service response

time, and task of delivering average service response time for different classes of service requests. Presented tasks of resource allocation can be better understood in terms of quality of service assurance models known from computer communication network theory [7], namely: best effort, IntServ and DiffServ models.

### 4.1. Average service response time minimization (best effort)

Average response time $\overline{d}_k$ experienced by service requests on $k$-th service path $s_k$ is the sum of average delays experienced on each link and atomic service belonging to service path $s_k$:

$$\overline{d}_k = \sum_{i=1}^{m} \overline{d}_{ij_i} + \sum_{i=1}^{m+1} \widehat{d}_{(i-1)j_{i-1}j_i}, \tag{13}$$

where $j_i \in s_k$ for $i = 0, \ldots, m$. Under assumption of Poisson request arrivals and exponential link and atomic service response times average response time $\overline{d}_k$ can be calculated more precisely as the expected value of complex service response time $E[r_k]$:

$$\overline{d}_k = E[r_k] = \int_0^\infty r_k f_{rk}(r_k) dr_k, \tag{14}$$

where $f_{rk}(r_k)$ is defined by equation (11).

Average response time $\overline{d}$ experienced by service requests in whole system can be calculated as the weighted average over response times of each path $s_k$ ($k = 1, \ldots, K$):

$$\overline{d} = \sum_{k=1}^{K} p_k \overline{d}_k, \tag{15}$$

where $p_k$ is the probability, that certain service request will be served along $k$-th service path $s_k$.

The aim of the task of minimization of the average service response time is to find such a vector $\mathbf{p} = [p_1 \ldots p_K]$ of probabilities of choosing different service paths $s_k$ for which average service response time is minimized:

$$\mathbf{p}^* = \arg\min_{\mathbf{p}} \sum_{k=1}^{K} p_k \overline{d}_k, \tag{16}$$

with respect to constraints on probabilities $\mathbf{p}$:

$$\sum_{k=1}^{K} p_k = 1 \quad \text{and} \quad p_k \geq 0 \ \text{ for } \ k = 1, \ldots, K. \tag{17}$$

Since average response time $\overline{d}_k$ of each service path $s_k$ depends on request arrival intensity $\lambda$, average request size $\overline{u}$ and probabilities $\mathbf{p}$, which change over time, optimization task (16) has to be solved iteratively in consecutive time steps. Resource allocation consists in assigning incoming request to service paths in such a way, that intensities of requests served along each service path are proportional to calculated probabilities $\mathbf{p}^*$. For large number $K$ of service paths this approach may be inefficient due to high computational complexity of optimization task (16). In such a case one can approximate optimal allocation by application of greedy approach, which for each new service request chooses service path $s_{k^*}$ with the lowest average delay $\overline{d}_{k^*}$:

$$k^* = \arg\min_k \overline{d}_k. \tag{18}$$

### 4.2. Service response time guaranties (IntServ)

Denote by $S(t_l)$ state of the system at the moment $t_l$ of arrival of new request $req_l$. State $S(t_l)$ contains information concerning moments of arrival, assigned service paths and location of all service request present in the system at moment $t_l$. Given system state $S(t_l)$ it is possible to calculate exact service response time $d_k(req_l)$ for request $req_l$ for each service path $s_k$:

$$d_k(req_l) = d(S(t_l), req_l, k), \tag{19}$$

where function $d(S(t_l), req_l, k)$ (presented in [4]) represents an iterative algorithm for calculation of response time of service request $req_l$ along $k$-th service path.

In the task of delivering quality of service it is assumed that each incoming request $req_l$ contains a vector $\mathbf{q}_l$ of requirements concerning values of various parameters describing quality of service. Besides required service response time $d_l^*$ vector $\mathbf{q}_l$ may contain parameters describing: security, cost, availability, etc.

The aim of the task of guarantying service response time is to find such a service path $s_{k^*}$ for which service response time requirements are satisfied:

$$k^* = \arg\max_k \left\{ d_k(req_l) \right\}. \tag{20}$$

with respect to:

$$d_k(req_l) \leq d_l^*.$$

It is possible that there does not exist such a path for which response time requirements are met. In this case requirements can be renegotiated, for example by suggesting minimal possible service response time $d_k^*(req_l)$:

$$d_k^*(req_l) = \min_k \{ d_k(req_l) \}. \tag{21}$$

When required service path $s_{k^*}$ is found (by solving either task (20) or (21)) in order to be able to guarantee requested service response time, resources on service path $s_{k^*}$ have to be reserved.

### 4.3. Average service response time guaranties (DiffServ)

Assume, that each incoming service requests $req_l$ belongs to certain class $c_l$ ($c_l = 1, \ldots, C$). Each class $c$ ($c = 1, \ldots, C$) is characterized by probability $q_c$, that response time requirements of requests from this class are met:

$$P\{d_l \leq d_l^*\} = q_{c_l}, \tag{22}$$

where $d_l$ and $d_l^*$ are respectively: request $req_l$ response time and request $req_l$ response time requirement.

The aim of the task of delivering average service response time guaranties is to assign each incoming service request $req_l$ to such a service path $s_{k^*}$ for which equation (22) holds. Since probability $P\{d_l \leq d_l^*\}$ for each service path $s_k$ can be calculated by means of cumulative distribution function $F_{rk}(d_l)$ (see eq. (12)), the task of delivering average service response time guaranties can be formulated as follows:

$$k^* = \min_k \{F_{rk}(d_l)\}, \tag{23}$$

with respect to:

$$F_{rk}(d_l) \geq d_l^*.$$

Similarly to the task of delivering strict guaranties, it is possible that neither of service paths allow to obtain required probability of meeting response time requirements. In such a case service path with the highest probability of meeting response time requirements may be suggested:

$$k^* = \max_k \{F_{rk}(d_l)\}. \tag{24}$$

## 5. Simulation study

In order to illustrate presented tasks of resource allocation and evaluate performance of proposed algorithms simulation study was carried out. In simulations an example system was set up. It consisted of three serially ordered atomic services ($m = 3$), each having three different versions ($n_1 = n_2 = n_3 = 3$).

In the system example it is assumed, that there are three distinguished request classes, each of which has predefined method of quality of service assurance. First request class is served according to *best effort* model (described in section 4.1.)

in which average service request response time is minimized. Requests from second class are served according do *IntServ* model (section 4.2.) which allows to deliver strict guaranties for maximal response time. Requests from the third class are served according to *DiffServ* model (section 4.3.) in which guaranties on average response time are delivered. Third request class consists of four subclasses, each characterized by different probability ($q_1 = 0,8, q_2 = 0,7, q_3 = 0,6, q_4 = 0,5$) of meeting required service response time $d^* = 0,5s$.

System was fed with a Poisson stream of requests with average stream intensity $\lambda = 50$. The share of each request class in overall stream was as follows: best effort - 50%, IntServ - 10% and DiffServ - 40%. Each subclass of DiffServ requests had 10% share in overall stream. The ratio of amount of requests from different requests classes was chosen to be similar to the ratio of traffic volume in real computer communication networks.

The aim of simulation was to evaluate performance of proposed resource allocation algorithms measured as response time guaranties delivered to distinguished traffic classes for increasing value of request arrival intensity. Exemplary results of performed simulations are presented on figures 4 and 5. On figure 4 the influence

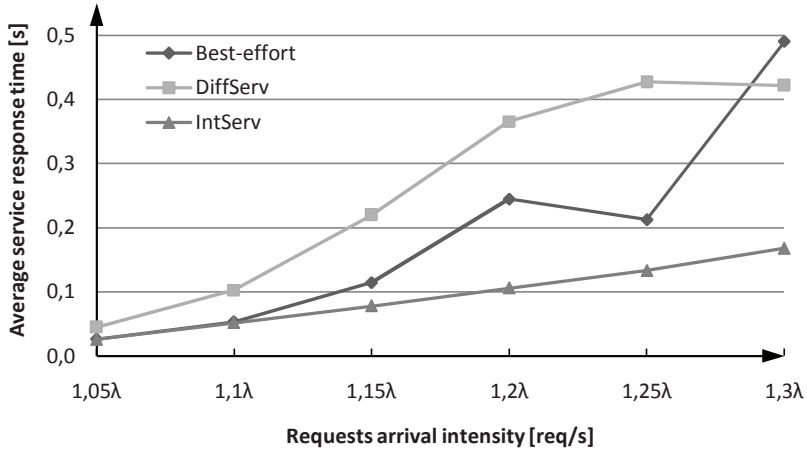



Fig. 4: Influence of increasing request arrival intensity $\lambda$ on average service response time for three main request classes: best effort, IntServ, DiffServ.

of increasing request arrival intensity on average service response time for three main request classes are presented. Requests from both best effort and IntServ classes are assigned resources such that average response time is minimized. There are two main differences between these two classes, namely resources for IntServ requests are reserved, what allows to provide strict guaranties on service response

times. Moreover, IntServ requests have higher priority then best effort requests. In fact best effort requests priority is the lowest among all classes, therefore requests scheduling algorithms in atomic services assign to best effort requests only such amount of computational resources which are not consumed by other classes.

It is obvious, that for increasing request arrival intensity average service response time should increase as well for all request classes. An interesting situation occurs when request intensity reaches $\lambda = 1,25\lambda_0$. Average response time of requests from DiffServ class approaches its requirement $d^* = 0,5s$ and stops increasing. At the same moment response time of best effort request starts to decrease and afterwards rapidly increases. This is caused by the fact, that when DiffServ class reached its requirement it did not need as much resources as earlier. Excess resources were assigned to best effort class, what resulted in decreased response time. When request intensity increased DiffServ class needed more resources to provide required response time guaranties. Necessary resources were taken from best effort class, what caused rapid growth of best effort average response time. Each

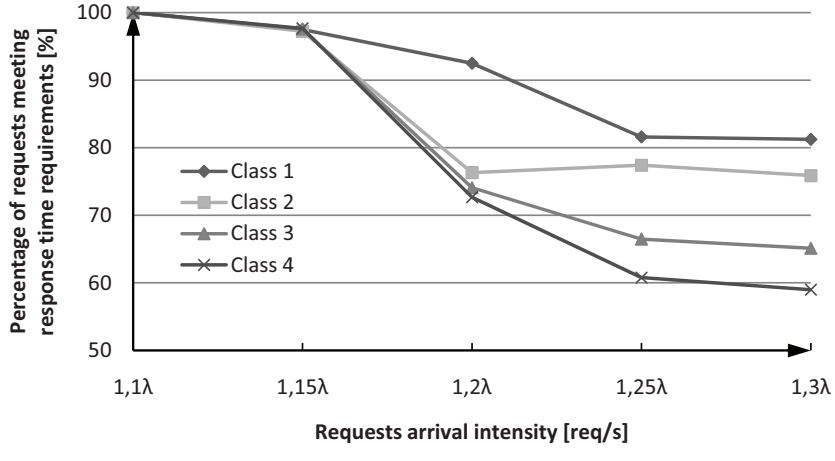



Fig. 5: Increasing request arrival intensity on the percentage of requests from each subclass of DiffServ meeting response time requirements.

subclass of DiffServ class have different requirements on percentage of requests meeting response time requirements. Exemplary results of quantitative analysis of the influence of increasing request arrival intensity on the percentage of requests from each subclass of DiffServ meeting response time requirements is presented on figure 5. One can notice, that as request arrival rate grows percentage of requests not violating response time guaranties approaches required values, which in presented study were set to $q_1 = 0,8, q_2 = 0,7, q_3 = 0,6, q_4 = 0,5$ for corresponding

subclasses.

## 6. Final remarks

Research presented in this paper shows, that it is possible to deliver required level of quality of service and differentiate it between distinguished request classes by application of commonly known quality of service assurance approaches. It is worth noting, that presented resource allocation algorithms utilize only few methods (resource reservation, request scheduling) from classical QoS assurance models. Application of all QoS mechanisms (e.g.: traffic shaping and conditioning, request classification, contract renegotiation, congestion control, etc.) as well as knowledge engineering methods [5] (e.g.: prediction of client behavior, adaptive scheduling, atomic services load prediction, etc.) to management of systems resources may allow to significantly improve delivered quality of service.

## Acknowledgements

The research presented in this paper has been partially supported by the European Union within the European Regional Development Fund program no. POIG.01.03.01-00-008/08.

## References

[1] S. Anderson, A. Grau, C. Hughes: Specification and satisfaction of SLAs in service oriented architectures, *5th Annual DIRC Research Conference*, pp. 141–150, 2005.

[2] N. Chee-Hock, S. Boon-Hee: *Queueing modelling fundamentals with application in Communication Networks, 2nd Edition*, Wiley and Sons, 2008, England.

[3] A. Grzech: *Teletraffic control in the computer communication networks*, Wroclaw University of Technology Publishing House, 2002, Wroclaw (in Polish).

[4] A. Grzech, P. Świątek: Modeling and optimization of complex services in service-based systems, *Cybernetics and Systems* , 40(08), pp. 706–723, 2009.

[5] A. Grzech, P. Świątek: Parallel processing of connection streams in nodes of packet-switched computer communication networks. *Cybernetics and Systems*, 39(2) pp. 155–170, 2008.

[6] N. Milanovic, M. Malek: Current Solutions for Web Service Composition, *IEEE Internet Computing* 8(6), pp.51–59, 2004.

[7] R. Rajan, D. Verma, S. Kamat, E. Felstaine, S. Herzog: A policy framework for Integrated and Differentiated Services in the Internet, *IEEE Network*, pp. 34–41, 1999.

[8] Z. Wang: *Internet QoS: architecture and mechanisms for Quality of Service*, Academic Press, 2001.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 49 – 58

# Fast Transition integrated into Media Independent Handover compliant network.

KRZYSZTOF ŁOZIAK

AGH - University of Science and Technology
*kloziak @kt.agh.edu.pl*

**Abstract:** This paper describes the possibility of integration a technology dependent solution of fast BSS transition (handover) into a heterogeneous, media independent handover environment. The proposed integration addresses the possibility of a seamless handover within a IEEE 802.11 access domain by decrease of handover latency, as well as ability of the access network to redistribute traffic in order to perform a local optimisation and to fulfil the QoS requirements from the Mobile Terminal perspective.
**Keywords:** Wireless LAN, local optimisation, handover, 802.11r, 802.21.

## 1. Introduction

Observed trends in a modern telecommunications networks show a constant growth of a heterogeneity level within a local access domain area. Combination of a different radio access techniques enables providers to offer a subscriber a variety of services, providing him in the same time the experience of a seamless mobility. Beside the major benefits coming from introduction of different radio technologies such as: multimode enabled mobile terminals and extended radio coverage due to overlapping areas of access points of a different techniques, providers have to face the fact of increased complexity of maintenance and management of heterogeneous domains. Seamless mobility in that sense requires a unified management platform, both for the network and users mobile terminals, as well, to guarantee a smooth services operation. Such a platform could be based on the IEEE 802.21 standard [1], which main goal is to create media independent handover environment. The IEEE 802.21 introduces a media independent architecture and allows to perform mobility procedures in the same way for a different radio techniques. However, unified framework does not take advantages of a technology specific features to decrease a total latency of a handover procedure in a specific radio access network. This paper presents

a possibility of integration of IEEE 802.21 and IEEE 802.11r standard [2] within an optimised wireless local access network, based on the intra-technology and intra-domain Network Initiated HandOver (NIHO) scenario. The IEEE 802.21 is used as a communication framework for all network modules taking part in the local optimisation scenario and preparation phase of handover procedure of a Mobile Terminal (MT), while IEEE 802.11r is used for the handover process execution.

## 2. IEEE 802.21 – communication framework

The ongoing work of IEEE 802.21 working group is focused on formulating a standard specification which is able to provide a generic interface between link layer users and existing media-specific link layers, such as those specified by 3GPP, 3GPP2 and the IEEE 802.x family of standards and to define media access mechanisms that enable possibility of performing handovers across heterogeneous networks. An abstraction layer MIH Function is a key concept of IEEE 802.21 standard. It provides services to the upper layers through a technology-independent interface (the MIH Service Access Point, MIH_SAP) and obtains services from the lower layers (driver layers) through a variety of technology-dependent interfaces (media-specific, MIH_LINK_SAPs). Its main role is to act as an abstraction layer between different links, physical layers and IP-layer hiding the particularities if transmission technologies. The IEEE 802.21 standard defines three basic Media Independent services: the Event Service, the Command Service and the Information Service. Detailed use cases and deployment of each of them are described in [1]. However, existing signalling does not include a QoS related operations and thus requires some extensions in order to have a common signalling platform for variety of technologies which are already QoS enabled (IEEE 802.11e, GPRS, EDGE, UMTS, HSxPA). Fig. 1. presents the general architecture of IEEE 802.21 and location of MIHF and services it provide.

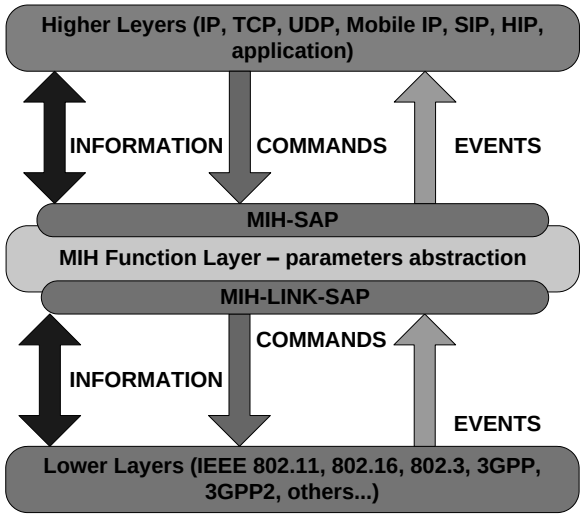



Fig. 1. IEEE 802.21 architecture

## 2. IEEE 802.11r – wireless LAN handover execution

The IEEE 802.11r Fast BSS Transition standard is devoted to support the handover process within a single BSS (Basic Service Set) domain with main emphasis put on decrease of a transition latency within the Wireless LAN local domains. The main goal of Fast BSS Transition services is to minimize or eliminate connectivity loss (to minimize it below 50 ms of overall delay) during performing a transition (handover) between Access Points. The fast BSS transition introduces a roaming capability for IEEE 802.11 networks similar to that of cellular ones. VoIP services within IEEE 802.11 networks have been thought as a primary application area.

Handover latency components in IEEE 802.11 consists of a re-association and a security binding. The IEEE 802.11r effort is focused on reducing the latency of security binding (pre-authentication). Moreover, an advance resource allocation is also considered by IEEE 802.11r, mainly due to the fact of QoS requirements of real-time applications.

Fast BSS Transition base mechanism commences just after the network discovery and target selection procedure is completed. It also supports resource requests as part of the re-association and this is called as the Fast BSS Transition reservation mechanism.

During the transition process Mobile Terminal passes through following three stages:

- discovery – MT locates and decides to which Point of Attachment (PoA) it will perform a transition,
- authentication and resource reservation – MT authenticates with target PoA and may reserve resources required to transit active sessions,
- (re)association – MT drops its current association and establishes a new one with target PoA.

During the discovery process the MT develops a list of potential transition candidates – list of neighbouring target Access Points and then based on this is able to perform a transition to a target PoA when and where it „decides" to. The fast BSS transition services provide mechanisms for MT to communicate with target PoA prior to performing transition, directly via radio interface or through existing MT's association

During the authentication phase, the fast BSS provides mechanisms for MT to reserve resources at the target PoA after authentication but prior to association.

The process of re-association starts when the best suited transition target has been found, and the MT can drop its active association and associates with a new one.

There are two ways of IEEE 802.11r operation (mechanisms of signalling exchange during transition process):

- over-the-air – a mechanism where the MT communicates directly with target PoA using 802.11 Authentication frames with Authentication Algorithm set to Fast BSS Transition,
- over-the-DS – a mechanism in which the MT communicates with target PoA using the Remote Request Broker (RBB) in the current PoA (communication between MT and current PoA is carried by fast BSS transition Action frames and between current PoA and target PoA via remote request encapsulation).

## 3. Integration – local optimisation scenario.

### 3.1 System architecture

An integrated architecture of the system proposed is based on the framework of IEEE 802.21 as a communication bearer during the handover preparation and the IEEE 802.11r as a tool of a mobile terminal handover execution within a Wireless LAN technology domain. Integration framework covers the scenario of intra-domain network initiated handover due to performance reason – the Local Optimisation case, which means that the network is able to detect a traffic overload at a single Point of Attachment and has an availability to overcome a congestion by moving Mobile Terminals associations across the domain. A sample architecture of such a system is presented in the Fig. 2.

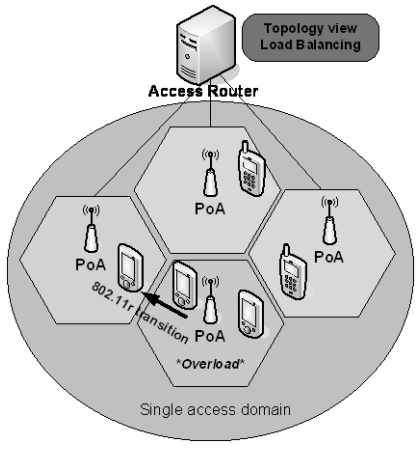



Fig. 2. Local Optimisation architecture.

The major innovation of presented approach is that Fast BSS Transition procedure is triggered by the network side and the list of possible handover targets within a single BSS domain is optimised by the Load Balancer Module (LBM) residing at the Access Router (network side) and not by a Mobile Terminal itself, as it is defined within IEEE 802.11r standard.

The main assumption of the Local Optimisation scenario is that network side nodes – especially LBM – are aware of the access network topology, i.e. list of connected Points of Attachment which are IEEE 802.11r compliant, their capabilities, such as: nominal operational data transmission rates, radio channel the Point of Attachment operates on, number of associated Mobile Terminals and most important parameter – current load per a single terminal. The topology awareness is essential during the optimisation process. Basing on the knowledge gathered, the LBM is able to redistribute traffic among intra-domain PoAs. Some static information (PoA's capabilities, such as: its MAC address of wireless interface, supported frame transmission rates, SSID and virtual SSID, radio channel, Tx power level, etc.) can be directly obtained by the LBM module from the IEEE 802.21 MIH Information Service.

Using the IEEE 802.21 *MIH_Get_Information.Request/Response* mechanism and information gathered by the MIH Information Service during the network nodes registering, the LBM can obtain data covering general and more detailed access network information elements, such as: network type, service provider and network operator identifiers, PoA L2 and L3 addresses, PoA location, data and channels range, cost, etc.

Moreover, to optimise the handover procedure and eliminate a possibility of a handover failure a signal strength measurements are required. Such kind of radio monitoring can be done both from the network and the terminal side. The architecture proposed assumes that signal level measurements are performed by

the Mobile Terminal and sent periodically or on-demand to the network. By receiving such kind of information LBM as a network module forms a network topology image including the ability of a Mobile Terminal to receive a signal from a specific Point of Attachment. Based on this, Points of Attachment which cannot be "heard" by a Mobile Terminal are not taken into account during the optimisation phase, and a number of handover failures is then strongly limited. Signal strength measurements are conveyed to the network side using the MIH Function layer events reporting mechanism. The LBM module acting as a MIHF layer user can register for events reporting for each Mobile Terminal which has registered itself into MIHF during the initial network association.

Integration of QoS requirements for a real-time based traffic can be done in parallel within a presented architecture. Resources reservation at the PoA can be achieved using the internal QoS support of IEEE 802.11r or by introducing additional QoS monitoring modules at the PoAs and AR. The IEEE 802.11r standard offers a simple mechanism for resources reservation at target PoA based on sending of a message containing resources reservation request and receiving in response information if such a reservation is possible or not. While the additional QoS monitoring could be combined with a traffic admission control mechanism.

Mobile Terminals traffic redistribution is the most important function of the LBM module. To achieve full operational level a constant traffic monitoring mechanism should be introduced, which allows to detect a current load per channel and a single terminal attached to the wireless network. The LBM is located within an Access Router and is responsible for generation and keeping up-to-date state of local access domain and detection and avoidance of PoAs overload. The LBM algorithm outcome is a new MTs association map generation by mean of network triggered handovers in order to eliminate PoAs overload. A lot of different optimisation algorithms have been proposed [3, 4], however most of them are based on access control and access restrictions. Some of them are focused only on selection of best-suited Access Point which in long-term approach can result with non-optimal traffic distribution. Majority of proposed algorithms are MT-based, which mean that all intelligence is located at the MT side and thus cannot be acceptable in real service provider networks. More reliable solutions are centralized and allow for handover triggering, which allows to eliminate the scanning part of handover procedure [4]. The LBM algorithm is similar to one proposed in [5], however it has been adopted to fit a multi-rate environment. The LBM optimisation algorithm is triggered by events observed within an access domain: new MT registration, PoA overload, new QoS contract registration or MT detachment. In order to ensure the proper conditions for LBM algorithm to run, some additional information related to MTs and access point are required. Information related to L2 operation of PoA are used to create an image of their utilisation and QoS realisation, while the MT itself is a source of

knowledge of an radio coverage of the access network. Mobile terminals are reporting the status of transmission quality, such as: received signal strength, frames rate ratio and observed delay, to the network using the MIHF event registration mechanism. When the optimisation procedure starts, the LBM generates a series of handover triggers for selected mobile terminals, which are directly conveyed by PoAs to MT's L2 drivers using the remote MIH communication.

### 3.2 Sample operation

In a Fig. 3. a sample message sequence chart of Local Optimisation algorithm output is depicted. The presented sequence starts when the algorithm is finished and set of handover triggers is to be sent to Mobile Terminals. The diagram shows the message exchange between the LBM module, single Mobile Terminal and its current and target Point of Attachment. Procedure of sending a handover trigger to a specific MT can be organised in two or just one step, called respectively: Soft and Hard NIHO. Considering the Soft NIHO procedure a Mobile Terminal receives a *MIH_Handover_Initiate. Request* message containing a list of possible handover targets, especially ID of target PoAs – i.e. MAC address and number of a radio channel PoA operates on. Based on this a Mobile Terminal has an opportunity to choose the best fit handover target which is then sent back to the LBM within the *MIH_Handover_Initiate.Response*.

At this point resources pre-reservation can be done using the *MIH_Handover_Prepare* messages if optional reservation mechanism if IEEE 802.11r is not used. However, additional QoS negotiations between QoS Broker and QoS Client residing at the Mobile Terminal can be taken also into account.

The usage of Soft and Hard NIHO procedure strongly depends on the optimisation time line requirements. In case of fast optimisation only the Hard NIHO procedure is executed. It relies on sending of a single selected handover target to the Mobile Terminal using the *MIH_Handover_Commit.Request* message.

After receiving the *MIH_Handover_Commit.Request* the MIHF layer at the MT sends a link layer two message WLAN-*_Link_Handover.Request* directly to the layer two driver.

Mobile Terminal, upon receiving the handover trigger runs the Fast BSS Transition procedure. The presented messages sequence covers the most complicated IEEE 802.11r scenario which is based on double hop communication between the MT and the target PoA using the current association point, and what is more using the mechanism of resources reservation. The communication process between the current serving PoA and target PoA is done

using the Remote Request Brokers (RRB) located at the PoA. RRB modules act as forwarding agents sending re-association requests from MT through an existing association from current to target PoA.

Resources reservation is described by Accepted and Active states. Accepted can be interpreted that the target PoA is able to guarantee required resources, while Active is synonymous with allocation of resources.

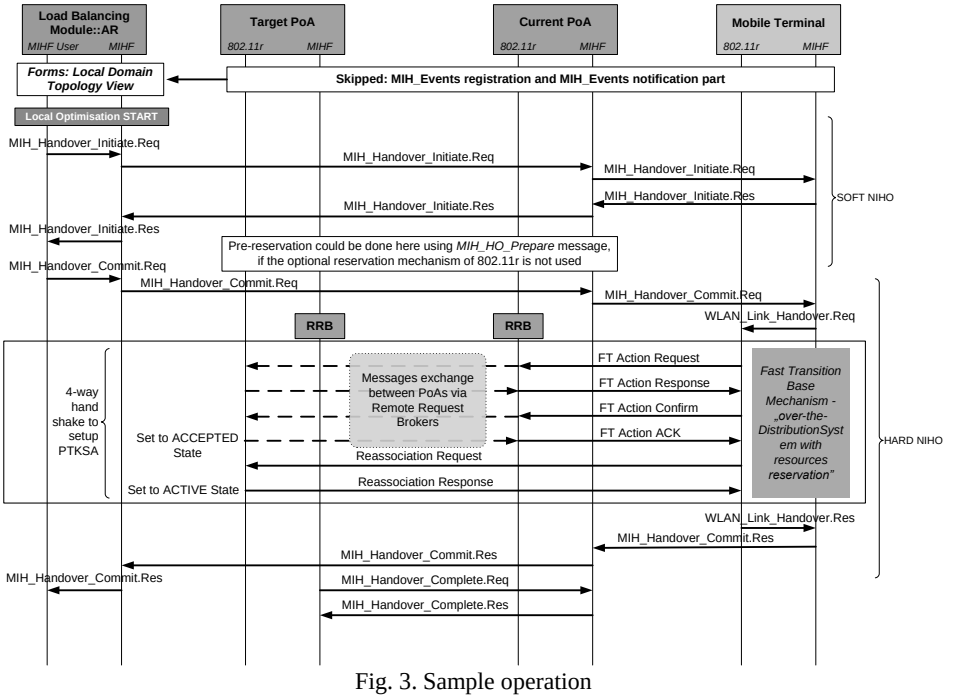



Fig. 3. Sample operation

Whole procedure of Local Optimisation NIHO scenario finishes with *MIH_Handover_Complete.Request/Response* messages exchange between former and current PoA which is a signal for previous one to forward all of buffered packets to a new PoA where the MT is actually associated.

The depicted scenario does not cover the MIH events registration and reporting part which is multiplied for each MT which is associated within the coverage area of single access domain. The notification part is essential from the LBM point of view to form a topology view of the optimised area.

### 3.3 Improving the handover latency.

The overall IEEE 802.11 layer two handover process latency consists of following phases: detection and recognition of conditions when a handover

procedure must be performed, scanning for available target access points, authentication procedure and finally a re-association to a selected access point. The values of each handover procedure component strongly depends on network architecture, hardware implementation and radio environment conditions [6, 7]. A lot of research has been done in area of experimental studies to derive the handover latency timings. Typical values for each of handover component show that the detection phase could last from 250 up to 600 ms, scanning procedure strongly depends on channel mask used and frequency band (IEEE 802.11a/b/g) and could last from 50 up to 400 ms for 2,4GHz band. The authentication and re-association phases should not exceed more than 10 ms, each.

The integration of IEEE 802.11r and IEEE 802.21 allows to decrease the handover latency by shortening or elimination of scanning procedure and detection phase. What is more, the existing pre-authentication mechanism of IEEE 802.11r should also improve the handover latency. The most efficient, from the perspective of whole handover procedure latency, is the Hard NIHO scenario, because its completely eliminates the scanning procedure triggering the MT with a single (network selected) handover target. The Soft NIHO adds a sequence of choosing by a MT the best-suited PoA. Both scenarios do not include the detection phase, because it is assumed that network decides on handover execution timeline.

## 4. Conclusion

The integrated architecture of heterogeneous media independent handover network enhanced with a strongly technology dependent handover mechanism has been presented. The main benefit of a proposed integration is the handover procedure latency decrease in case of local optimization scenario by usage of network initiated handover trigger, as well as current (up-to-date) access network state image processing and resources utilization analysis.

# References

[1] IEEE P802.21- Media Independent Handover Services, http://www.ieee802.org

[2] IEEE P802.11r/D3.0, Fast BSS Transition

[3] Shiann-Tsong Sheu and Chih-Chiang Wu, *Dynamic Load Balance Algorithm (DLBA) for IEEE 802.11 Wireless LAN*, Tamkang Journal of Science and Engineering, VOL 2, pp 45-52, 1999

[4] Anand Balachandran et al "Hot-Spot Congestion Relief in Public-area Wireless Networks," in Proc. of Fourth IEEE Workshop on Mobile Computing Systems and Applications, p. 70, June 2002

[5] Yigal Bejerano et al, *Fairness and Load Balancing in Wireless LANs Using Association Control*, in Proc. Of International Conference on Mobile Computing and Networking, pp. 315 – 329, Philadelphia, 2004

[6] Mishra, A., Shin, M., *An empirical analysis of the IEEE 802.11 MAC layer handoff process*, ACM SIGCOMM Computer Communication Review 2003, p. 93–102.

[7] Ramani, I., Savage, S., *SyncScan: Practical fast handoff for 802.11 infrastructure networks*, IEEE INFOCOM, 2005

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 59 – 64

# Different approaches of NGN and FGI.

IVAN KOTULIAK EUGEN MIKÓCZY

Slovak University of Technology in Bratislava
Ilkovičova 3, 812 19 Bratislava, Slovakia
*{Ivan.Kotuliak, mikoczy}@ktl.elf.stuba.sk*

**Abstract:** In this article we focus on comparison of main characteristics of the Next Generation Networks and Future Generation Internet. From the architectural point of view, we compare selected properties of both approaches to explain commonalities and differences. Based on this comparison, we propose to work areas to more closely evaluate possible convergence in the approaching of the two architectures to Future Network concept.
**Keywords:** NGN, NGI, telco services.

## 1. Introduction

Words with meaning "Future" and "Upcoming" are very popular as they design something what will be available and most recent or future proof. Anybody can usually extend such a future in its own way. Communication technology has borrowed these terms in two very popular technologies: NGN and NGI.

NGN means Next Generation Networks and its meaning is to the Telco approach in building multimedia services using IP networks. It comes from the Telco industry and standardization organizations like ETSI, ITU-T and 3GPP. Traditional Telco has proposed to throw out SS7, signalization protocol and various SDH hierarchies for transport and replace it by all IP networks with standardized service control platform called IP Multimedia subsystem (IMS).

NGI (Next Generation Internet) also known as Future Generation internet (FGI) has born as an approach how to deliver new services upon "Internet 2" or "Beyond IP". These networks can be designed in evolution way or in clean slate approach on "green field". FGI has been driven by philosophical thoughts, what we would do design of Internet differently, if we could design Internet today from scratch.

The aim of this article is to discuss two approaches to the upcoming networks and services: NGN and NGI. We present both frameworks from the services

point of view as they are delivered to the end-user. This article is more philosophical one, which should improve the discussions about future networks.

The rest of the article is organized as follows: the basic idea of both NGN and FGI is presented in second section. The third Section focuses on the aims of the FGI and how NGN fulfils some of these requirements and where the NGN does not still gives any response. In final Section, we conclude the article with ideas for future work.

## 2. Future Networks

This section provides overview of the NGN and FGI concepts and main goals for both approaches.

### 2.1 Next Generation Networks

ITU-T define NGN [1] as a network based on packet transfer, enabling to provide services, including telecommunication services, and is capable of using several broadband transmission technologies allowing guaranteeing QoS. The functions related to services are at the same time independent of the basic transmission technologies. NGN provides unlimited user access to different service providers. It supports general mobility providing the users with consistency and availability of services.

That is what definitions say, but probably most important in this discussion are original requirements for NGN that should be fulfilling as following:

- High-capacity packet transfer within the transmission infrastructure, however, with a possibility to connect existing and future networks (be it the networks with packet switching, circuit switching, connection-oriented or connectionless, fixed or mobile).
- Separation of managing functions from transmission features. Separation of service provisioning from the network and ensuring the access via an open interface and thus a flexible, open and distributed architecture.
- Support for a wide range of services and applications by using the mechanisms based on the modular and flexible structure of elementary service building blocks
- Broadband capabilities, while complying with the requirements for QoS (Quality of Services) and transparency. Possibility of a complex network management should be available.
- Various types of mobility (users, terminals, services). Unlimited access to a variety of service providers.

- Various identification schemes and addressing which can be translated to the target IP address for the purposes of routing in the IP network. (Flexible addressing and identification, authentication).
- Converged services between fixed and mobile networks (as well as voice, data and video convergence). Various categories of services with the need of different QoS and classes of services (CoS).
- Conformance to the regulation requirements, such as emergency calls and security requirements in terms of personal data protection.
- Cheaper and more effective technologies if compared to the current technologies.

### 2.2 Future Generation Internet

Future Generation Internet has born to resolve drawbacks of the current Internet. The Internet [2] as we know it today has evolved from the original interconnection sponsored by DARPA research. The IETF, which has covered Internet-related standards, targeted the layer approach. Each layer of the protocol stack should provide only necessary functionality on this layer and do it as efficient as possible. All other functionalities not implemented on certain layer or on any underlying should be implemented on higher layer.

This approach was successful and yielded in the protocol stack as we know it today. In the centre, there is the famous Internet Protocol (IP). Under IP protocol, we can find device and physical medium specific layers. Typically, there is Ethernet, WiFi, but can by any other protocol. IP protocol was originally designed provide best effort services but later there have been develop managed IP networks with operator/provider controlled infrastructure. Additionally there is the MultiProtocol Label Switching (MPLS), which introduces Quality of Service and circuit switched service notion to the IP world. Above IP, there is usually datagram services User Datagram Protocol (UDP), or connection oriented Transmission Protocol (TCP). On this layer, several protocols have been designed, but none has gained wide popularity. The real problem starts above transport layer, where service dependent protocols exists and are used. Typical representants here are the HTTP for web-oriented services, FTP for file transfer, SMTP for e-mail, XMPP for chat, SIP for signalling and tens of others.

Described TCP/IP model, used in today works has several drawbacks. These drawbacks have incited new research in the area of the improvement of the current Internet [3,4,5]. Two main approaches exist: evolution of the current Internet and the “clean slate” approach [6]. The clean slate approach rethinks Internet from the beginning: how would we implement Internet with today knowledge. For a moment the FGI focuses on the formalization of new models and interfaces.

Next Section provides the challenges for the Future Internet and confronts these challenges with the NGN existing approach.

## 3. NGI challenges

The main challenges for the NGI coming from the experiences of the users and experts with the Internet are:

- Addressing
- Identity
- Security and privacy
- End-to-end QoS/QoE
- Mobility

In the following text, we focus on these challenges.

### 3.1 Identity and adressing

Long years, there were separate networks for different purposes. Consequently, every network used its own identifying mechanisms. In current world, the customers prefer to have the same identity for all applications (e.g. the same phone number across any network).

To respond to this challenge, the NGN implements the system of private and public identity. The private identity is used by the network e.g. for authorization. The public identity (user is allowed to posses multiple public identities) is used by other users and by applications. In this way, anybody can reach the user using his public identity independently on the currently used access network. In such way, one can use the same number via mobile and fixed network (this notion does not have the same sense in the NGN as today).

### 3.2 Security and privacy

Security and privacy are typical problems of the current Internet. Existing networks have been designed just to work and no security mechanisms have been designed from the beginning. With increasing threats, several security mechanisms and architectures have been proposed (e.g. IPSec, SSL etc). These proposals have drawback, because they try to resolve one problem, but usually they do not have ambition to propose security concept.

The IMS based NGN proposes that every user would have the identity card (e.g. ISIM – IMS Subscriber Identity Module) including security credentials. The credentials are used for authorization of the user identity and for the secure connection to the network through access network. Afterwards, the user is trusted and the identity is only the application question. In such way, we can build a secure island of one (multiple interconnected) providers.

Another important point is privacy. The privacy can be maintained by using multiple public identities (as described previously). Even if required, the public identity in the access network for the signalling purposes can be changed in regular intervals. The major advantage is that all this can happen seamlessly from user perspective.

### 3.3 End-to-end QoS/QoE

Quality of Service (QoS) and Quality of Experience (QoE) are two terms, which are related to the perception of the services by end-users. The two terms are closely related. The substantial difference can be for the application, where the human senses are inaccurate and even objectively bad performance they accept in very good way.

As the NGN has born in the telco world, it has stressed the QoS from the beginning. There have been standardized and implemented mechanisms allowing the packet marking in the access to the transport networks and QoS management in the core network. However, the QoS management is done per class of application to minimize the quantity of the Class of Services (CoS) in the core network.

### 3.4 Mobility

Mobility is one of the services available to the end-users. The applications should be available anywhere and anytime.

Real problem is the connectivity. Moving user usually gets out of the network coverage. For WiFi, the network coverage is in the scale of hundreds meters, one mobile network covers at most the area of one country. Besides these limits, the user should make handover to other network with the same technology, or with different one. This process usually means the change of the IP address. However, this problem is not really important to the end-user.

End-user is more focused on the applications. Therefore, there should be a framework allowing the users seamless handover via various technological environments. NGN is independent from the access networks and it is able to provide the services through any access technology. This is ensured by the separation of the applications from access layer. The only problem is the resource allocation and QoS management, which can be different for various networks types.

## 4. Conclusions

In this article, we have focused on the future networks: the Next Generation Networks and Future Generation Internet. NGN has been born in operators

environment as revolution in telco world providing service continuity and QoS management over IP. FGI has been born in the Internet world as a response to the challenges from existing IP networks. FGI has also used revolutionary approach known as “clean slate” approach, when existing protocols should be replaced by the new ones.

We have described how NGN has managed with selected challenges of the FGI like mobility, identity, security and privacy, and QoS management. The NGN of course cannot respond easily to all the FGI challenges, but it is important to get inspiration from more strict world than open Internet has always been.

## 5. Acknowledgement

This article is the result of work in progress started within Celtic/Eureka project Netlab. It has been mainly sponsored by Slovak National Research agency.

## References

[1] ITU-T, SG13, NGN 2004 Project description v.3, 2004

[2] Clark, D. D.: *The design philosophy of the DARPA internet protocols.* In: Proc. ACM SIGCOMM'88, Computer Communication Review 18 (4) pp. 106--114 (August, 1988).

[3] Blumenthal M. S., Clark, D. D.: *Rethinking the design of the Internet: The end to end arguments vs. the brave new world*, In: ACM Transactions on Internet Technology, Vol 1, No 1, August 2001, pp 70-109

[4] User Centric Media Future and Challenges in European Research European Communities, 2007, ISBN 978-92-79-06865-2.

[5] Clark D., Chapin L., Cerf V., Braden, R. Hobby, R.: *RFC 1287 Towards the Future Internet Architect*

[6] Feldman A.: *Internet Clean-Slate Design: What and Why?*. in ACM SIGCOMM Computer Communication Review, Volume 37, Number 3, July 2007, pp. 59 – 64.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 65–74

# PlanetLab Automatic Control System

YONG YAO DAVID ERMAN

School of Computing
Blekinge Institute of Technology
*{yong.yao|david.erman}@bth.se*

**Abstract:** PlanetLab provides a global scale environment for network research. It consists of numerous controllable computer nodes. However, due to that these nodes are deployed in various network domains, there exist experimental issues, regarding network latency, resource utilization, and unpredictable behaviors of nodes. Experimenters need to pay much attention to manually adjusting their selected nodes during the whole experimental process. PLACS is a novel tool system based on PlanetLab, composed of *node selection* and *node management*. It can intelligently and automatically coordinate the involved elements, including PlanetLab nodes, experimental programs and experimenter requirements. In this article, the corresponding mechanism functionalities and implementation designs for development are presented.

**Keywords:** PlanetLab, Large-scale experiments, Automated experiments, Experiment control

## 1. Introduction

Much attention has been put on the research and development of new future network services. To obtain reliable and relevant experimental results requires a worldwide platform of the network system to be as close to real-world as possible. For this, many research groups adopt PlanetLab [1].

PlanetLab is a group of computers available as a testbed for computer networking and distributed systems research. It was established in 2002, and as of March 2009, was composed of 913 nodes at 460 sites worldwide [2]. Due to the high number of participating nodes, it is cumbersome to manually manage these computers to be involved in distributed experiments, *i. e.*, distributed storage, network mapping, Peer-to-Peer (P2P) systems, distributed hash tables, and query processing. In order to improve the experimental efficiency of PlanetLab, we have developed an automatic control system, called PlanetLab Automatic Control System (PLACS). It

provides novel mechanisms to intelligently select suitable nodes to satisfy distinct experimental requirements. Besides, PLACS can automatically manage involved experimental elements and coordinate their interactions, *e. g.*, PlanetLab nodes, executable programs, experimental requirements.

The purpose of this paper is to provide a overview of PLACS. Related work is addressed in Section 2. Section 3 represent the major mechanisms of PLACS, *node selection* and *node management*. Their corresponding implementation designs for future development are also represented in Section 5. Finally, we conclude the paper in Section 6.

## 2. Related work

Currently, PlanetLab consists of more than 1000 nodes from different continents. PlanetLab Central (PLC) provides detailed information about these nodes for potential management, *i. e.*, host name, hardware resource information, TCP connectivity, graphical illustration of locations, convenient interfaces for adding/removing nodes. To obtain suitable nodes for dedicated experiments, experimenters are forced to evaluate the characteristics of every PlanetLab node. Due to this, tools have been developed to simply this evaluation process, *e. g.*, pShell [3], CoDeeN [4] [5] [11]. The pShell is a Linux shell and works as a command center at the local machine. It provides a few basic commands to interact with Planetlab nodes. However, for pShell users, they still need to manually select suitable nodes. In CoDeeN, several PlanetLab nodes are adopted as proxy servers. They behave both as *request redirectors* and *server surrogates*. Each node operates independently, from node selection to forwarding logic [8]. They can exchange health information about nodes based on periodic heartbeat communications with each other. However, CoDeeN lacks centralized coordinations for experimenters. It can not be used to cope with interactions between PlanetLab nodes and experiments. For instance, if a node becomes disconnected from the network, the behavior of programs running on it will become uncontrollable.

Regarding carrying out actual experiments, tools are available for managing selected nodes and experimental files, *e. g.*, PlMan [6], AppManager [7]. They provide command interfaces for experimenters, *i. e.*, deploying files, monitoring status of nodes, starting or stopping programs on selected nodes. For distributing large files, CoBlitz [9] and CoDeploy [10] are available. Both use the same infrastructure for transferring data, which is layered on top of the CoDeeN content distribution network [8] [11]. However, these tools focus on individual tasks on PlanetLab nodes, *e. g.*, copying/deleting files, starting/stopping programs. They do not provided a effective solution to automatically perform multistep serial tasks,

composed of these individual ones. For example, when experimenters have used these tools for distribution, they still need to manually start or stop executable programs on selected nodes.

Furthermore, in previous experiments, we found fluctuations related to PlanetLab node characteristics. This experiment is regarding the evaluation of BitTorrent Extensions for Streaming (BiTES) [12] and started from August 22, 2009 until Septemper 22. Five different scenarios were carried out. Each of these scenarios were run 15 times and each run utilized around 100 PlanetLab nodes. More than 900 PlanetLab nodes were manually tested for availability. From this daunting but important work, several negative characteristics of the node in PlanetLab were obtained. These are:

- Connectivity fluctuation: Every PlanetLab node may become disconnected from the network. Un-connectable nodes can not be operated by experimenters, until they recover being connectable.
- Bandwidth fluctuation: Some nodes are provided by the same research group and deployed in the same local network domain. If all these nodes are involved in the same experiment, they may become competitors with each other for the local bandwidth allocation. This is important as there exists an automatic bandwidth limitation for all nodes in PlanetLab, *e. g.*, 10 Gigabytes (GB) per node per day for uploading.
- Latency fluctuation: If the network latency of nodes is evaluated from different network domains, result values may be different. Such multiple latency references may lead experimenters to a conflicting view for choosing suitable nodes.
- Utilization fluctuation: Nodes in PlanetLab are not exclusive for a single experiment. They can be simultaneously used by different experiments. This coexistence leads to competition of hardware resources on the same node. Thus, the performance of the different experiments may be interfered by each other.

Since PlanetLab is running on the top of the public internet, the above mentioned fluctuations are expected. Although PLC and some tools provide detailed information of node characteristics, experimenters still need to manually analyze them. To address these issues, we propose PLACS to provide a series of novel mechanisms to solve the above mentioned problems. According to the experimental requirements, PLACS intelligently and automatically selects suitable nodes and manages them to carry out experiments. This means that experimenters can focus less on manual operation of PlanetLab nodes and concentrate on developing their experiments. The following two sections describe the major mechanisms of PLACS and corresponding implementation designs.

## 3. Mechanisms

We classify the major mechanism of PLACS as *node selection* and *node management*. The role of *node selection* is to intelligently select out of suitable PlanetLab nodes, according to experimental requirements. Then, *node management* is used to automatically manage selected nodes to carry out experiments.

### 3.1. Node selection

Several basic node characteristics can be collected by using interfaces provided by PLC or other relevant tools, *i. e.*, host name, geographical location, research group. This information will be recorded by the Central Controller (CC) of PLACS for system initialization. We propose to use the CC to directly evaluate node connectivity and resource utilization, *e. g.*, CPU utilization, Memory utilization, Hard Disk utilization. Besides, constraints for desired nodes should be given by experimenters, *e. g.*, allowable maximum network latency and CPU utilization. Undesirable nodes will be eliminated by PLACS.

Because deployment locations of PlanetLab nodes can not be changed by experimenters, the latency fluctuation is difficult to be avoided. We propose to deploy additional computers, called Assistant Agents (AAs), to obtain accurate latency information. These AAs will be deployed in different geographical network domains, *e. g.*, Europe, Asian, America. As shown in Fig. 1, AAs can respectively perform latency evaluations for the same destination node, *e. g.*, *ping command*. As a result, depending on different geographic locations of AAs, every PlanetLab node will provide multiple references of network latency. If the experiment is involved in the same geographic location with a certain Assistant Agent (AA), the latency evaluated by this AA becomes the main reference. Since it is expensive to deploy and maintain many independent AA computers, we propose to adopt representative PlanetLab nodes as AAs. Thus, the deployment of AAs can have global coverage.

Regarding the practical selection process, we propose two specific scenarios, *applicability* and *equilibrium*. The *applicability* refers to requirements of resource utilization and network bandwidth. Some experiments require a large amount of processing power. Thus, nodes with lower utilization of memory and CPU are preferred, even if they have higher network latency. In contrast, if experiments focus on the research of network communication, nodes with lower network latency have higher priority for selection. On the other hand, some distributed systems, *e. g.*, P2P systems, need multiple nodes to participate in experiments. The experimenters may then adopt a certain location distribution of selected nodes, *e. g.*, 25% in European, 25% in Asian, 25% in America, and other 25% in Australia. If different location distributions are adopted for repeated experiments, the corresponding results may

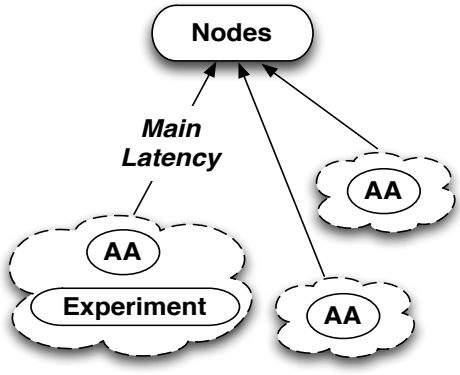



Fig. 1. Multiple latency evaluations from AAs.

differ, due to the distinct latency of different geographical network domains. Therefore, the adopted location distribution needs to be preserved as a *equilibrium* status, during the whole experimental process.

### 3.2. Node management

After suitable PlanetLab nodes have been selected by *node selection*, PLACS will automatically manage them to carry out actual experiments. To simplify the following representation, we call the selected PlanetLab nodes as Selected Nodes (SNs). They constitute a Selected Node (SN) swarm during the experimental process. In PLACS, there exist three kinds of *node management*. They are *basic* management, *composed* management and *advanced* management.

Akin to PlMan [6] and AppManager [7], *basic* management provides experimenters individual operations to manually manage SNs and their experiments. These operations include copying files to SNs, monitoring the copying progress, deleting files from SNs, starting programs on SNs, stopping programs on SNs, and collecting result files from SNs. In some cases, many SNs may be in the same destination domain, and experimental files need to be distributed from another source domain to them. If files are large, as well as large network latency between destination and source domains, the whole distribution may take a long time. For this issue, as shown in Fig. 2, a specific AA will be selected by PLACS to transmit files. It should be in the same network domain as the destination SNs. Firstly, files are copied to such AA. Then, this AA distributes them to all SNs within the same network domain. Because these SNs have a smaller network latency to this AA than from source network domain, the time for the whole distribution can be decreased.

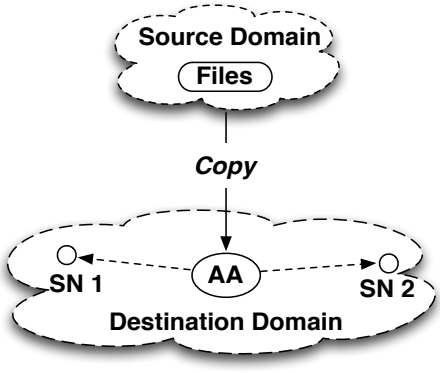



Fig. 2. Transmission role of AA.

The *basic* management operations can be composed together to construct *composed* operations. The main goal of such composition is to automatically carry out experiments. For example, PLACS can automatically start programs on all SNs, after completing the copying progress. Furthermore, we propose to maintain a *heartbeat* communication between every SN and PLACS. Both can periodically evaluate the connectivity between them and evaluated results can guide their future behaviors. As shown in Fig. 3, if a certain SN can not connect to PLACS for some while, the running program on this SN will stop itself. Meanwhile, this SN will be announced as a un-connectable node by PLACS, and will be eliminated from current SN swarm. This mechanism helps experimenters to indirectly avoid the uncontrollable behaviors of running programs on disconnected SNs. Furthermore, evey SN will periodically report the uploaded information to PLACS, with using the heartbeat communication. When PLACS has found that the total uploaded size on one node is close to 10 GB per day, it will remove this node from the SN swarm. But, this node will have a higher priority to return to the SN swarm after one day.

We propose *advanced* management to intelligently coordinate the interactions between PlanetLab nodes and practical experiments. As we know, SNs may become unsuitable, due to unpredictable reasons, *e. g.*, becoming un-connectable, network latency increasing. PLACS can adaptively select other suitable nodes to replace these unsuitable SNs, without any participations of experimenters. These new SNs will be automatically initialized by PLACS, *e. g.*, deploying experimental files to them.

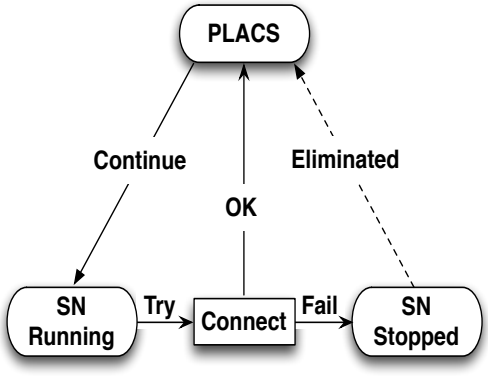



Fig. 3. Heartbeat communication between PLACS and SN.

## 4. Implementation design

As previous mentioned in Section 3, the CC and AAs are two major components in PLACS. We propose that the CC is made up of by one or more computers. They can be deployed independently with PlanetLab. However, processing node selections and managing SNs may increase the computer processing loads, as well as the network utilization, and can not be carried out on the CC. In addition, it is extremely complex to coordinate multiple different experiments on the CC. Therefore, we propose the Terminal Client (TC) to perform these tasks. As its name implies, TC can be used as client software to make different experiments involved in PLACS. Before carrying out actual experiments, preparations for experiments should be done independently from PLACS, *i. e.*, programming, debugging and designing experimental scenarios. Then, experimenters can input their specific parameters into TC as register information, *e. g.*, constraints of desired nodes, required SN number, and the location distribution of SNs.

As shown in Fig. 4, CC and AAs automatically evaluate characteristics of recorded PlanetLab nodes, *e. g.*, connectivity, resource utilization, network latency. CC also maintains mapping relationships between SNs and specific AAs, according to their network domains. In order to avoid node information from going stale, these evaluations are periodically performed by CC, *e. g.*, once per day. The feedback information will be updated into the data base of CC. When experimenters start TC, the register information will be sent to CC. Meanwhile, CC passively sends characteristic information of available nodes to TC. With the respect to the geographic location of involved TC, the CC will send out the main network latency of these nodes. According to the received information, TC performs *node selection*

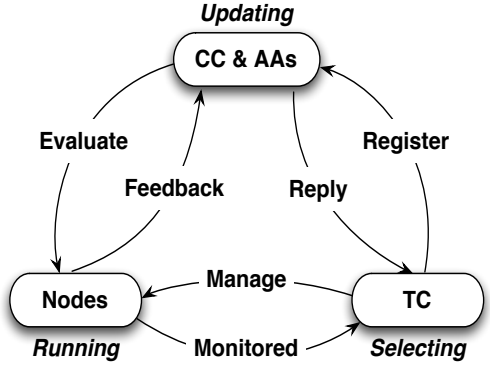



Fig. 4. Triangle communications among CC, TC, and nodes.

to pick out suitable SNs.

Then, the TC automatically distributes experimental files to all SNs. The corresponding distribution progress can be monitored by the TC, including AA transmissions. It will be exhibited to experimenters. After completing the whole distribution, programs on all SNs can be started by the TC. Moreover, the TC provides experimenters optional software interfaces. They can be integrated into experimental programs developed by experimenters. Using these interfaces, TCP *heartbeat* communication between SNs and TC will be automatically established, when programs start running on SNs. Typically, the running program will automatically stop itself when assigned tasks have been completed. Sometimes, the whole experimental system needs to be terminated. For instance, after all programs have been started, if a serious fault has been found that programs will run forever and never terminate. In this case, experimenters do not need to manually terminate every program on all SNs, in this case. They can just disconnect the TC from network. Then, programs running on all SNs will automatically stop, due to that they can not connect TC. This is also the reason why we adopt TC as the other object for *heartbeat* communications.

## 5. Conclusion

In this paper, we describe several experimental issues that when using PlanetLab nodes. We also represent two major mechanisms of PLACS, *node selection* and *node management*, as well as implementation designs. They help experimenters to intelligently select suitable PlanetLab nodes and automatically manage them to carry out experiments. Thus, experimenter can concentrate on their research and

experimental setups. As future work, other details of PLACS will be considered, *i. e.*, selecting and managing AAs, reducing the network traffic of simultaneous heartbeat communications on TC.

## References


[1] PlanetLab Org: http://www.planet-lab.org/

[2] PlanetLab EU: http://www.planet-lab.eu/

[3] B. Maniymaran, pShell: An Interactive Shell for Managing Planetlab Slices, http://cgi.cs.mcgill.ca/ anrl/projects/pShell/index.php/

[4] L. Wang, V. Pai and L. Peterson, The Effectiveness of Request Redirection on CDN Robustness, *Proceedings of the 5th OSDI Symposium*, December 2002.

[5] CoDeeN, *Princeton University*: http://codeen.cs.princeton.edu/

[6] PlanetLab Experiment Manager, http://www.cs.washington.edu/research/networking/cplane/

[7] PlanetLab Application Manager, http://appmanager.berkeley.intel-research.net/

[8] B. Biskeborn, M. Golightly, K. Park, and V. S. Pai, (Re)Design Considerations for Scalable Large-File Content Distribution, *Proceedings of Second Workshop on Real, Large Distributed Systems(WORLDS)*, San Francisco, CA, December 2005.

[9] K. Park and V. S. Pai Scale and Performance in the CoBlitz Large-File Distribution Service, *Proc. of NSDI*, 2006, pp. 29-44.

[10] K. Park and V. Pai. Deploying Large File Transfer on an HTTP Content Distribution Network, *Proceedings of the First Workshop on Real, Large Distributed Systems(WORLDS '04)*, 2004.

[11] L. Wang, K. Park, R. Pang, V. Pai, and L. Peterson, Reliability and security in the CoDeeN content distribution network, *Proceedings of the USENIX Annual Technical Conference*, 2004.

[12] D. Erman, On BitTorrent Media Distribution, *PhD thesis*, Blekinge Institute of Technology, March 2008.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 75 – 88

# Examination of robust stability of computer networks

JERZY KLAMKA[a] JOLANTA TAŃCULA[b]

[a] Institute Theoretical and Applied Informatics PAN Gliwice
*jerzy.klamka@iitis.pl*

[b] Silesian University of Technology
*tancula@vp.pl*

**Abstract**: The thesis presents a nonlinear mathematical model of a computer network with the RED algorithm. The model equations were linearized around the set operating point and next block diagrams of the system presented. Using the methods for examining stability known from the literature, the robust stability of the system has been analysed with slight changes in its parameters. An example of stability examination prepared on the basis of the data for a real computer network has been shown. The results generalise the results known in the literature.

**Keywords**: Computer networks, stability, robust stability.

## 1. Introduction

Active Queue Management (AQM) consists in early, preventive signalling the congestion (when a buffer is not full yet) to TCPs by IP routers, in order to avoid serious congestion in TCP. AQM routers serve a key role in producing better results in Internet applications.

## 2. Dynamic Simplified model of TCP

A dynamic simplified model of the TCP may be described with the use of stochastic non-linear differential equations [1], [2].

$$W'(t) = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t-R(t))}\, p(t-R(t))$$

$$q'(t) = \begin{cases} -C + \dfrac{N(t)}{R(t)} W(t) & dla \quad q > 0 \\ 0 & dla \quad q = 0 \end{cases}$$

where x'(t) denotes the time-derivative,

W- average TCP window size (packets)
q - average queue length (packets)
R(t) - round trip time (sec)
C – link capacity (packets/sec)
$T_P$ - propagation delay (sec)
N – load factor (number of TCP sessions)
p – probability of packet mark

The equations mentioned above may be illustrated in the block diagram (see [1],[2] for details).

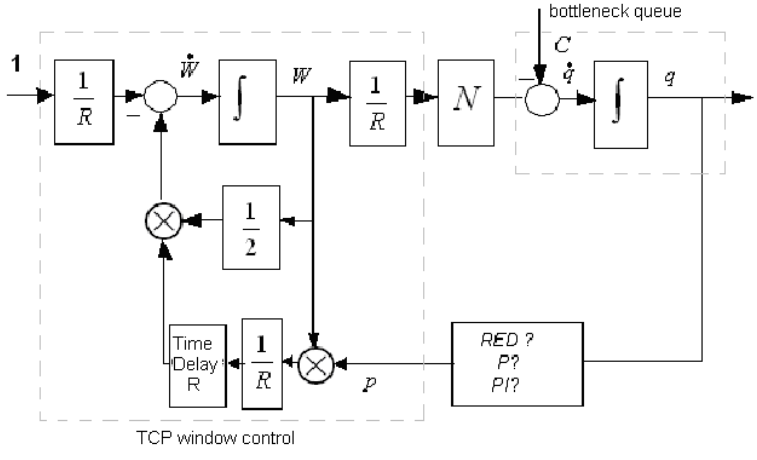



Fig. 1 Block diagram of differential equations.

## 3. Linearization of equations of mathematical model

We apply approximation of the system dynamics by linearization of a nonlinear mathematical model around the set operating point. Let's assume that a number of TCP sessions and the connection throughout are constant. Taking W and q as constants and p as input data, the set operating point ($W_0$, $q_0$, $p_0$) is defined by W = 0 and q=0. Hence

$$W'(t) = 0 \Rightarrow W_0^2 p_0 = 2$$

$$q'(t) = 0 \Rightarrow W_0 = \frac{R_0 C}{N} \;\; ; \quad R_0 = \frac{q_0}{C} + T_p$$

In the linearization process we assume (t-R(t)) = (t-Ro). Thus after the linearization around the set operating point we achieve

$$\delta W'(t) = -\frac{N}{R_0^2 C}\left(\delta W(t) + \delta W(t - R_0)\right) - \frac{1}{R_0^2 C}\left(\delta q(t) - \delta q(t - R_0)\right) - \frac{R_0 C^2}{2N^2}\delta p(t - R_0)$$

$$\delta q'(t) = \frac{N}{R_0}\delta W(t) - \frac{1}{R_0}\delta q(t)$$

where

$$\delta W = W - W_0$$
$$\delta q = q - q_0$$
$$\delta p = p - p_0$$

show changes of parameters around the set operating point.

Applying the Laplace transformation to differential equations we obtain a new block diagram

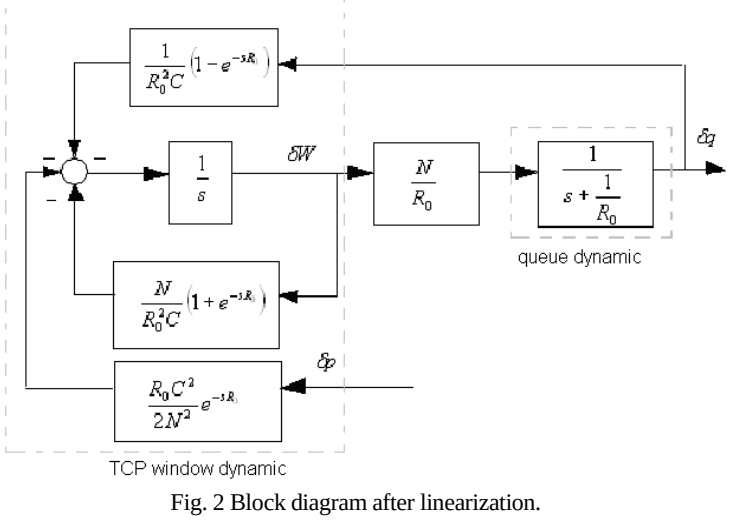



Fig. 2 Block diagram after linearization.

Transforming the block diagram in Fig. 3 and introducing Δs:

$$\Delta(s) = \frac{2N^2 s}{R_0^2 C^3}\left(1 - e^{-sR_0}\right)$$

the block diagram is simplified as follows:

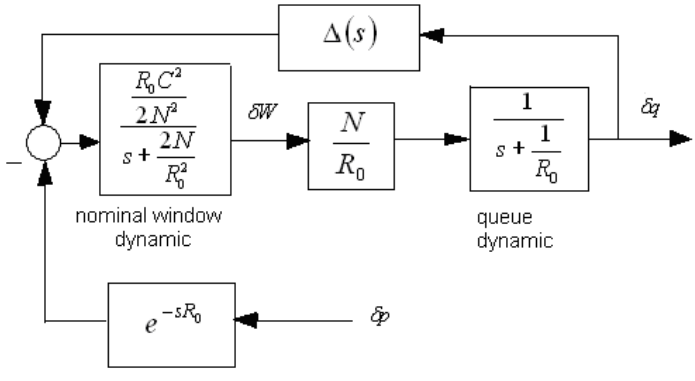



Fig. 3. Simplified block diagram.

The interpretation of the hard time window is shown by a linearized equation

$$\delta W'(t) = -\lambda \delta W(t) - \frac{R_0 C^2}{2N^2}\delta p(t - R_0)$$

where λ is a packet marking rate. The pace of change in the window size depends linearly on the window size itself and the packet marking probability.

## 4. Control problem for AQM

AQM determines packet marking probability *p* as a function of the queue length measured. The model dynamics is determined by operational transfer function P(s) (see Fig.4).

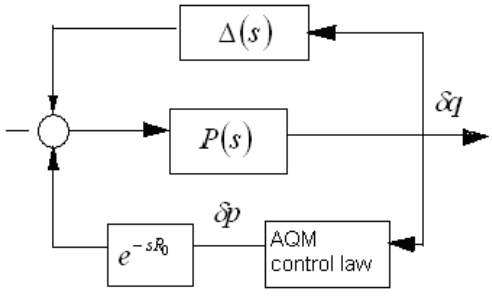



Fig. 4 Block diagram with AQM control law.

Transfer function P(s) defines how the dynamic packet marking probability affects the queue length, while Δ(s) represents a high frequency of the window dynamics. We calculate transfer function P(s)

$$P(s) = \frac{\frac{C^2}{2N}}{\left(s + \frac{2N}{R_0^2 C}\right)\left(s + \frac{1}{R_0}\right)}$$

Transfer function C(s) represents AQM linear control law.
Further simplification of the block diagram given in Fig. 5:

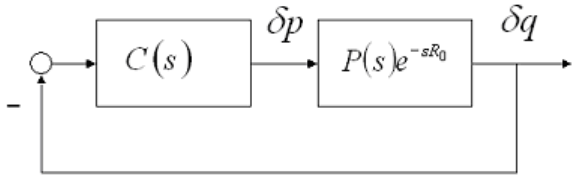



Fig. 5 Simplified block diagram.

A basic algorithm for Internet routers is RED algorithm. The algorithm calculates packet marking/dropping probability as a function of the queue length *q* described with the application of the AQM control law. The RED algorithm is composed of low pass filter K (to determine the average queue length) and a packet marking/dropping function with probability *p* shown in Fig. 6.

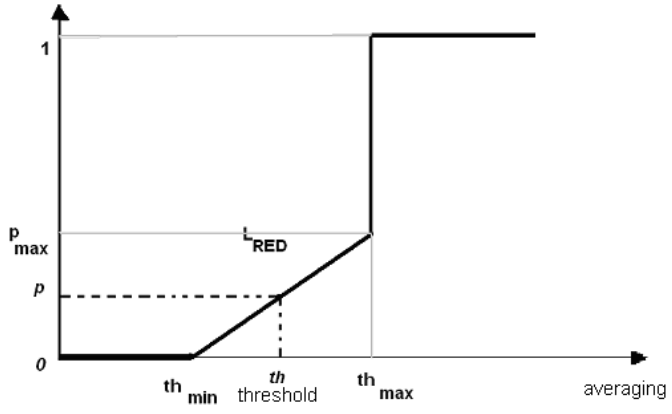



Fig. 6 Packet marking/dropping function with probability *p*.

Transfer function of the RED algorithm has the following form:

$$C(s) = \frac{K}{s+K} L_{RED} ,$$

where

$$L_{RED} = \frac{p_{\max}}{\max_{th} - \min_{th}} ; \; K = \frac{\log_e (1-\alpha)}{\delta}$$

$a>0$ is a parameter of the average queue length, and $\delta$ is time sample. $Max_{th}$ and $min_{th}$ are threshold values responsible for packet dropping and average queue length in time $t$ is calculated following the formula for moving average

$$avg(t) = (1-w)avg(t-1) + wq(t)$$

## 5.Examination of stability

In order to determine stability conditions under which the system in Fig. 7 is stable, the Hurwitz criterion or the Nyquist criterion should be applied. Below we are going to employ a method of uncertain parameters based on characteristic quasi-polynomials [3].
A family of quasi-polynomials will be designated with *W(s,h,Q)*, i.e.

$$W(s,h,Q) = \{w(s,h,q) : q \in Q\} ,$$

where Q is a set of parameter values

$$Q = \{q : q_k \in [q_k^-, q_k^+], q_k^- < q_k^+, k = 1,2,\ldots,l\},$$

and $q=[q_1, q_2, \ldots, q_1]$ is a vector of uncertain parameters of the dynamic system, while $w_o(s,h)=w(s,h,q_0)$ designates a nominal (reference) quasi-polynomial.
The examination of robust D-stability of the quasi-polynomial family using a method of uncertain parameters [3] consists in establishing limits of D-stability (i.e. of $Q_g$ set) in the space of uncertain parameters and checking if they cross or not Q set (the set of uncertain parameter values). As it is not easy to check a multidimensional space, for polynomials D-stability and Q set limits may be projected on the surface of two selected uncertain parameters. D region is a shifted open left half-plane whose edge has a parameter description.

$$f(\omega) = -\gamma + j\omega,\ \omega=[0,\infty),\ \text{a}\ \ Q_g=\{q \in R^l : w(f(\omega),h,q)=0,\ \omega\geq 0\}.$$

If nominal quasi-polynomial $w_0(s,h)$ is D-stable, then a necessary and sufficient condition for robust D-stability of the quasi-polynomial family is not to cross Q set by $Q_g$ set.
For $s=f(\omega)$ quasi-polynomial $w(s,h,q)$ may be presented as follows:

$$w(f(\omega),\ h\ ,q)=U(\omega,q) + jV(\omega,q),$$

where

$$U(\omega,q)=Rew(f(\omega),h,q),\ \ V=(\ \omega,q)=Imw(f(\omega),h,q).$$

If for some $q \varepsilon R^1$ quasi- polynomial $w(s,h,q)$ has zeros at the edge of D region, it may be real zero $s = -y$ (then vector $q$ is at the limit of real zeros) or combined conjugated para zeros $s = -y \pm j\omega$, for $\omega>0$ (then vector $q$ is at the limit of conjugated zeros). Then the relation is as follows:

$$Q_g = Q_{gr} \cup Q_{gz},$$

where:
$Q_{gr} = \{q:\ w(-\gamma,h,q) = 0\}$ is a limit of real zeros, and
$Q_{qz}=\{q:U(\omega,q)=0$ *and* $V(\omega,q)=0$ for some $\omega>0\}$ is a limit of combined zeros.
It follows that quasi-polynomial family $W(s,h,Q)=\{w(s,h,q): q \in Q\}$ is robustly D-stable if and only if the planes described by the equation and the system of equations

$$w(-\gamma,h,q) = 0$$

and differential equations

$$U(\omega,q)=0 \text{ and } V(\omega,q)=0,$$

where $\omega \varepsilon (0, \infty)$ is a parameter, do not cross Q set.

A process of determining D-stability limits may be simplified if quasi-polynomial rates depend linearly on uncertain parameters.

If quasi-polynomial rates $w(s,h,q)$ depend linearly on uncertain parameters, then a quasi-polynomial may be presented as follows:

$$w(s,h,q)=w_0(s,q)+\sum_{i=1}^{m} w_i(s,q)\exp(-sh_i),$$

and equation $w(-\gamma,h,q) = 0$ described by the equation

$$B_0+\sum_{k=1}^{l} q_k B_k = 0,$$

where $B_k = w_0(-\gamma)+\sum_{i=1}^{m} w_{ik}(-\gamma)\exp(\gamma h_i)$, k=0,1,...,l.

## 6. Example

Following a method of uncertain parameters we are going to examine robust D-stability of the automatic control system with delay with uncertain parameters and the block diagram shown in Fig. 6. Operational transmittance P(s) of the system is

$$P(s)\cdot e^{-sR_0} = \frac{\frac{C^2}{2N}}{s^2+\left(\frac{2N+R_0C}{R_0^2C}\right)s+\frac{2N}{R_0^3C}}\cdot e^{-sR_0}$$

we accept

$$q_1=\frac{C^2}{2N}, \qquad q_2=\frac{R_0C+2N}{R_0^2C}, \qquad q_3=\frac{2N}{R_0^3C}$$

we get

$$P(s)e^{-sR_0} = \frac{q_1e^{-sR_0}}{s^2+q_2s+q_3}.$$

Transfer function of the RED algorithm is

$$C(s)=\frac{L_{RED}}{\frac{s}{K}+1}=\frac{KL_{RED}}{s+K},$$

for transparency of calculations we assume $L_{RED} = L$.
Transfer function of whole system K(s) is calculated as follows:

$$K(s)=\frac{\frac{KL}{s+K}\cdot\frac{q_1\cdot e^{-sR_0}}{s^2+q_2s+q_3}}{1+\frac{KL}{s+K}\cdot\frac{q_1\cdot e^{-sR_0}}{s^2+q_2s+q_3}}=\frac{KL\cdot q_1e^{-sR_0}}{(s+K)(s^2+q_2s+q_3)+KLq_1e^{-sR_0}}.$$

A characteristic quasi-polynomial of the system under consideration has the following form

$$w(s,R_0,q)=w_0(s,q)+w_1(s,q)e^{-sR_0},$$

where

$$w_0(s,q)=(s^3+Ks^2)+(s^2+Ks)q_2+(s+K)q_3,$$
$$w_1(s,q)=KLq_1.$$

It is possible to prove that a nominal quasi- polynomial

$$w_0(s,h)=(s^3+Ks^2)+KL$$

is D-stable corresponding to $q$ values.
A problem of robust D-stability we are going to examine on plane ($q_1$, $q_2$). A projection of Q set onto plane ($q_1$, $q_2$) is a rectangle.

$$Q_p=\{q_p=[q_1,q_2]: q_1\in[-49035,\quad 7243], q_2\in[-2.9;\quad 1.8]\}.$$

The set is a set of deviation values of uncertain parameters from their nominal values calculated as follows:

$$q_1^0=\frac{C_0^2}{2N_0}=117187.5,$$

$$q_1^+=\frac{(C_0+0.1C_0)^2}{2(N_0-0.1N_0)}=124431.13,$$

$$q_1^-=\frac{(C_0-0.1C_0)^2}{2(N_0+0.1N_0)}=68151.99,$$

$$q_2^0 = \frac{R_0 C_0 + 2N}{R_0^2 C_0} = 4.59,$$

$$q_2^+ = \frac{(R_0 + 0.1R_0)(C_0 + 0.1C_0) + 2(N_0 + 0.1N_0)}{(R_0 - 0.1R_0)^2 (C_0 - 0.1C_0)} = 7.55,$$

$$q_2^- = \frac{(R_0 - 0.1R_0)(C_0 - 0.1C_0) + 2(N_0 - 0.1N_0)}{(R_0 + 0.1R_0)^2 (C_0 + 0.1C_0)} = 2.83,$$

$$q_3^0 = \frac{2N_0}{R_0^3 C_0} = 2.15,$$

$$q_3^+ = \frac{2(N_0 + 0.1N_0)}{(R_0 - 0.1R_0)^3 (C_0 - 0.1C_0)} = 3.55,$$

$$q_3^- = \frac{2(N_0 - 0.1N_0)}{(R_0 + 0.1R_0)^3 (C_0 + 0.1C_0)} = 0.13$$

D region is a shifted open left half-plane whose edge has a parameter description.

$$f(\omega) = -\gamma + j\omega, \ \omega = [0, \infty)$$ , gdzie $\gamma = 0.05$.

For $s = f(\omega)$ quasi-polynomial $w(s, R_0, q)$ may be presented as follows

$$w(f(\omega), R_0, q) = U(\omega, q) + jV(\omega, q),$$

where

$$U(\omega, q_1, q_2) = R_0(\omega) + q_1 R_1(\omega) + q_2 R_2(\omega) + q_3 R_3(\omega)$$

$$V(\omega, q_1, q_2) = I_0(\omega) + q_1 I_1(\omega) + q_2 I_2(\omega) + q_3 I_3(\omega).$$

We calculate a limit of complex zeros, i.e. we are solving the following equation

$$A(\omega) q_p(\omega) = b(\omega, q_3),$$

$$A(\omega) = \begin{bmatrix} R_1(\omega) & R_2(\omega) \\ I_1(\omega) & I_2(\omega) \end{bmatrix} \quad q_p(\omega) = \begin{bmatrix} q_1(\omega) \\ q_2(\omega) \end{bmatrix} \quad b(\omega, q_3) = \begin{bmatrix} -R_0(\omega) - q_3 R_3(\omega) \\ -I_0(\omega) - q_3 R_3(\omega) \end{bmatrix}$$

$$\begin{cases} R_0(\omega) + R_1(\omega) q_1 + R_2(\omega) q_2 + R_3(\omega) q_3 = 0 \\ I_0(\omega) + I_1(\omega) q_1 + I_2(\omega) q_2 + I_3(\omega) q_3 = 0 \end{cases}$$

$$\begin{bmatrix} R_1(\omega) & R_2(\omega) \\ I_1(\omega) & I_2(\omega) \end{bmatrix} \cdot \begin{bmatrix} q_1(\omega) \\ q_2(\omega) \end{bmatrix} = \begin{bmatrix} -R_0(\omega) - q_3 R_3(\omega) \\ -I_0(\omega) - q_3 I_3(\omega) \end{bmatrix} \text{ or}$$

$$\begin{bmatrix} KL & \gamma^2 - \omega^2 - K\gamma \\ 0 & K\omega - 2\gamma\omega \end{bmatrix} \cdot \begin{bmatrix} q_1(\omega) \\ q_2(\omega) \end{bmatrix} = \begin{bmatrix} \gamma^3 - K\gamma^2 + K\omega^2 - 3\omega^2\gamma + \gamma q_3 - Kq_3 \\ -3\gamma^2\omega + \omega^3 + 2K\gamma\omega - \omega q_3 \end{bmatrix}$$

The above equation, we have

$$\begin{bmatrix} q_1(\omega) \\ q_2(\omega) \end{bmatrix} = \mathrm{A}(\omega)^{-1} \cdot \mathrm{b}\,(\omega)\,,$$

$$\begin{bmatrix} q_1(\omega) \\ q_2(\omega) \end{bmatrix} = \begin{bmatrix} \dfrac{1}{KL} & 0 \\ \dfrac{\gamma^2 - \omega^2 - K\gamma}{KL\omega(K - 2\gamma)} & \dfrac{1}{\omega(K - 2\gamma)} \end{bmatrix} \cdot \begin{bmatrix} \gamma^3 - K\gamma^2 + K\omega^2 - 3\omega^2\gamma + \gamma q_3 - Kq_3 \\ -3\gamma^2\omega + \omega^3 + 2K\gamma\omega - \omega q_3 \end{bmatrix}$$

Hence we have

$$q_1(\omega) = \frac{\left(\gamma^3 - K\gamma^2 + K\omega^2 - 3\omega^2\gamma + \gamma q_3 - Kq_3\right)}{KL},$$

$$q_2(\omega) = \frac{\left(\gamma^2 - \omega^2 - K\gamma\right)\left(\gamma^3 - K\gamma^2 + K\omega^2 - 3\omega^2\gamma + \gamma q_3 - Kq_3\right) + KL\omega(-3\gamma^2 + \omega^2 + 2K\gamma - q_3)}{KL\omega(K - 2\gamma)}$$

For K=0.005, L=1.86·10$^{-4}$, γ=0.05 and $q_3^0 = 2.15$, $q_3^+ = 3.55$ and $q_3^- = 0.13$ and for ω from ranges ω ε [0; 10], ωε[0.1; 10.1] ω ε [0.2;10.2] we get a chart of three curves. When magnified we get diagram (Fig.7) for $q_3^+ = 3.55$ (curve1), $q_3^0 = 2.15$ (curve 2) and $q_3^- = 0.13$ (curve 3). Curve 1 does not cross the rectangle

$$Q_p = \left\{q_p = [q_1, q_2] : q_1 \in [-49035,\quad 7243], q_2 \in [-2.9;\quad 1.8]\right\}$$

which means that the system is stable for value $q_3$ = 3,55.

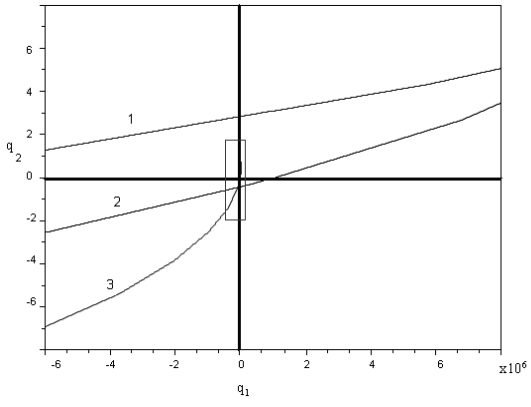



Fig. 7 Curve 1 does not cross $Q_p$ rectangle for value $q_3 = 3{,}55$, where $q_1$, $q_2$ is a limit value for a set of uncertain parameters of a dynamic system.

Now it is only to determine limits of real zeros. We solve the following equation:

$$B_0 + \sum_{k=1}^{l} q_k B_k = 0 ,$$

where $B_k = w_0(-\gamma) + \sum_{i=1}^{m} w_{ik}(-\gamma)\exp(\gamma h_i)$, k=0,1,...,l.

We assume $s = -y$ and for $q_3 = 2{,}15$ we receive a linear equation

$$q_2 = 42.9444 + 0.0004185 q_1 .$$

For every $q_3 \in [-2.02; 1.4]$ a straight line is on plane $(q_1, q_2)$ far from $Q_p$ rectangle.

## 7. Summary

Following the method above to calculate a limit of complex zeros and a limit of real zeros it is shown that rectangle

$$Q_p = \{q_p = [q_1, q_2] : q_1 \in [-49035,\ \ 7243], q_2 \in [-2.9;\ \ 1.8]\},$$

will not cross curve 1 which appeared in stability limits indicated by $q_1(\omega)$ and $q_2(\omega)$. It means that the quasi-polynomial family is robustly D-stable for value $q_3$ from range $[0;\ \ 1.4]$

A method of uncertain parameters may be applied to examine robust D-stability of quasi-polynomial family whose rates depend linearly, multilinearly and polynomially on uncertain parameters. The method may be used only for a small number of parameters.

# References

[1] Hollot C.V., Misra V., Towsley D., Wei Bo Gong: *A Control Theoretical Analysis of Red.*

[2] Hollot C.V., Misra V., Towsley D., Wei Bo Gong: *Analysis and Design of Controllers for AQM Routers Supporting TCP Flows.*

[3] Busłowicz M.: *Odporna stabilność układów dynamicznych liniowych stacjonarnych z opóźnieniami,* Wydawnictwa Politechniki Białostockiej, Białystok, 2002.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 89–98

# Simulation environment for delivering quality of service in systems based on service-oriented architecture paradigm

ADAM GRZECH [a] PIOTR RYGIELSKI [a] PAWEŁ ŚWIĄTEK [a]

[a]Institute of Computer Science
Wroclaw University of Technology, Poland
{adam.grzech, piotr.rygielski}@pwr.wroc.pl

**Abstract:** In this paper a model of complex service in service-oriented architecture (*SOA*) system is presented. A complex service is composed with a set of atomic services. Each atomic service is characterized with its own non-functional parameters what allows to formulate quality of servie optimization tasks. A simulation environment has been developed to allow experiments execution to determine quality of service (*QoS*) of composed service.

**Keywords:** : quality of service, service-oriented architecture, simulation

## 1. Introduction

Recent popularity of system based on service-oriented architecture (*SOA*) paradigm has lead to growth of interest concerning quality of service level provisioning in such systems. Efficient management of system resources can lead to delay decrease, cost optimization and security level increase [6].

In this paper a model of serially connected atomic services is considered. Atomic services are organized into layers where each atomic service in one layer has equal functionality and differs in non-functional parameters values such as processing speed, security level, cost of execution etc. Incoming requests should be distributed in such way that user requirements concerning quality of service are satisfied and are profitable for service provider. For testing purposes for such service distribution algorithms a simulation environment has been developed.

The paper is organised as follows. In section 2 a serial-parallel complex service model is presented. Problem of resource distribution for incoming service request has been formulated in section 3. In section 4 a simulation environment proposed

as a testbed for quality of service provisioning algorithms is described. Usage of developed simulation software is presented by example in section 5. Section 6 is dedicated for final remarks and future work outline.

## 2. Complex service model

It is assumed that new incoming $i$-th complex service request is characterized by proper Service Level Agreement description denoted by $SLA(i)$. The $SLA(i)$ is composed of two parts describing functional and nonfunctional requirements, respectively $SLA_f(i)$ and $SLA_{nf}(i)$. The first part characterizes functionalities that have to be performed, while the second contains values of parameters representing various quality of service aspects. The $SLA_f(i)$ is a set of functionalities subsets:

$$SLA_f(i) = \{\Gamma_{i1}, \Gamma_{i2}, \ldots, \Gamma_{ij}, \ldots, \Gamma_{in_i}\} \tag{1}$$

where:

- $\Gamma_{i1} \prec \Gamma_{i2} \prec \ldots \prec \Gamma_{ij} \prec \ldots \prec \Gamma_{in_i}$ ordered subset of distinguished functionalities subsets required by $i$-th complex service request; $\Gamma_{ij} \prec \Gamma_{ij+1}$ (for $j = 1, 2, \ldots, n_i - 1$) denotes that delivery of functionalities from the subset $\Gamma_{ij+1}$ cannot start before completing functionalities from the $\Gamma_{ij}$ subset.
- $\Gamma_{ij} = \{\varphi_{ij1}, \varphi_{ij2}, \ldots, \varphi_{ijm_j}\}$ (for $i = 1, 2, \ldots, n_i$) is a subset of functionalities $\varphi_{ijk}$ ($k = 1, 2, \ldots, m_j$) that may be delivered in parallel manner (within $\Gamma_{ij}$ subset); the formerly mentioned feature of particular functionalities are denoted by $\varphi_{ijk} \parallel \varphi_{ijl}$ ($\varphi_{ijk}, \varphi_{ijl} \in \Gamma_{ij}$ for $k, l = 1, 2, \ldots, m+j$ and $k \neq l$).

The proposed scheme covers all possible cases; $n_i = 1$ means that all required functionalities may be delivered in parallel manner, while $m_j = 1$ (for $j = 1, 2, \ldots, n_i$) means that all required functionalities have to be delivered in sequence.

It is also assumed that the $\varphi_{ijk}$ ($j = 1, 2, \ldots, n_i$ and $k = 1, 2, \ldots, m_j$) functionalities are delivered by atomic services available at the computer system in several versions.

The nonfunctional requirements may be decomposed in a similar manner, i.e.:

$$SLA_nf(i) = \{H_{i1}, H_{i2}, \ldots, H_{ij}, \ldots, H_{in_i}\} \tag{2}$$

where $H_{ij} = \{\gamma_{ij1}, \gamma_{ij2}, \ldots, \gamma_{ijm_j}\}$ is a subset of nonfunctional requirements related respectively to the $\Gamma_{ij} = \{\varphi_{ij1}, \varphi_{ij2}, \ldots, \varphi_{ijm_j}\}$ subset of functionalities.

According to the above assumption the $SLA_f(i)$ of the $i$-th complex service request may be translated into ordered subsets of atomic services:

$$SLA_f(i) = \{\Gamma_{i1}, \Gamma_{i2}, \ldots, \Gamma_{ij}, \ldots, \Gamma_{in_i}\} \Rightarrow \{AS_{i1}, AS_{i2}, \ldots, AS_{ij}, \ldots, AS_{in_i}\}, \tag{3}$$

where $\{AS_{i1}, AS_{i2}, \ldots, AS_{ij}, \ldots AS_{in_i}\}$ is a sequence of atomic services subsets satisfying an order $(AS_{i1} \prec AS_{i2} \prec \ldots \prec AS_{ij} \prec \ldots \prec AS_{in_i})$ predefined by order in the functionalities subsets. The order in sequence of atomic services is interpreted as the order in functionalities subsets: $AS_{ij} \prec AS_{i,j+1}$ (for $j = 1, 2, \ldots, n_i - 1$) states that atomic services from subsets $AS_{i,j+1}$ cannot be started before all services from the $AS_{ij}$ subset are completed.

Each subset of atomic services $AS_{ij}$ (for $j = 1, 2, \ldots, n_i$) contains $a_{ijk}$ atomic services (for $k = 1, 2, \ldots, m_j$) available at the computer system in several versions $a_{ijkl}$ ($l = 1, 2, \ldots, m_k$). Moreover, it is assumed that any version $a_{ijkl}$ ($l = 1, 2, \ldots, m_k$) of the particular $a_{ijk}$ atomic services (for $k = 1, 2, \ldots, m_j$) assures the same required functionality $\varphi_{ijk}$ and satisfies nonfunctional requirements at various levels.

The above assumption means that – if $fun(a_{ijkl})$ and $nfun(a_{ijkl})$ denote, respectively, functionality and level of nonfunctional requirements satisfaction delivered by $l$-th version of $k$-th atomic service ($a_{ijkl} \in AS_{ij}$) – the following conditions are satisfied:

- $fun(a_{ijkl}) = \varphi_{ijk}$ for $l = 1, 2, \ldots, m_k$,
- $nfun(a_{ijkl}) \neq nfun(a_{ijkr})$ for $l, r = 1, 2, \ldots, m_k$ and $l \neq r$.

The ordered functionalities subsets $SLA_f(i)$ determines possible level of parallelism at the $i$-th requested complex service performance (in the particular environment). The parallelism level $l_p(i)$ for $i$-th requested complex service is uniquely defined by the maximal number of atomic services that may be performed in parallel manner at distinguished subsets of functionalities $(SLA_f(i))$, i.e.,

$$l_p(i) = \max\{m_1, m_2, \ldots, m_j, \ldots, m_{n_i}\}. \tag{4}$$

The possible level of parallelism may be utilized or not in processing of the $i$-th requested complex service. Based on the above notations and definitions two extreme compositions exist. The first one utilizes possible parallelism (available due to computation resources parallelism), while the second extreme composition means that the requested functionalities are delivered one-by-one (no computation and communication resources parallelism).

The above presented discussion may be summarized as follows. The known functional requirements $SLA_f(i)$ may be presented as a sequence of subsets of

functionalities, where the size of the mentioned latter subsets depends on possible level of parallelism. The available level of parallelism defines a set of possible performance scenarios according to which the requested complex service may be delivered. The space of possible solutions is limited – from one side – by the highest possible level parallelism and – from the another side – by natural scenario, where all requested atomic services are performed in sequence.

The mentioned above extreme compositions determines some set of possible $i$-th requested complex service delivery scenarios. The possible scenario can be represented by a set of graphs $G(i)$ – nodes of graph represent particular atomic services assuring $i$-th requested complex service functionalities, while graph edges represent an order according to which atomic services functionalities have to be delivered.

The set of all possible graphs $G(i)$ $\Big(G(i) = \{G_{i1}, G_{i2}, \ldots, G_{is}\}\Big)$ assures the requested functionality, but offers various level of nonfunctional requirement satisfaction. The latter may be obtained (and optimized) assuming that at least one node of the particular graph contains at least two versions of the requested atomic service.

## 3. Problem formulation

In general, the optimization task of maximizing delivered quality may be formulated as follows:

**For given:**

- Subsets of atomic services $a_{ijkl} \in a_{ijk} \in AS_{ij}$
- Set of all possible graphs $G(i)$ for $i$-th requested complex service
- Subsets of nonfunctional requirements $H_{ij}$
- Processing scheme given by order $AS_{i1} \prec AS_{i2} \prec \ldots \prec AS_{ij} \prec \ldots \prec AS_{in_i}$

**Find:** such subset of atomic services versions $a_{ijkl}$ that executed with respect to processing scheme maximizes quality of service $SLA^*_{nf}(i)$.

$$SLA^*_{nf}(i) \leftarrow \max_{G_{i1}, G_{i2}, \ldots, G_{is}} \left\{ \max_{a_{ijkl} \in a_{ijk} \in AS_{ij}} \{H_{i1}, H_{i2}, \ldots, H_{in_i}\} \right\}. \qquad (5)$$

The latter task may be reduced where the particular $i$-th requested complex service composition (i.e., an graph equivalent to particular $i$-th requested complex

service processing scheme) is assumed. In such a case the optimization task can be formulated as:

$$SLA^{*}_{nf}(i, G_i) \leftarrow \max_{a_{ijkl} \in a_{ijk} \in AS_{ij}} \{H_{i1}, H_{i2}, \ldots, H_{in_i}\}. \quad (6)$$

The above formulated task means that the optimal versions of atomic services, determined by the selected $i$-th requested complex service performance scenario, should be selected.

## 4. Simulation environment

Testing new methods and approaches for improving system quality on real system is time-consuming and can be difficult, expensive and cause damage to system [1]. Therefore there is a need to develop a testbed for testing such mechanisms. There are several simulation environments dedicated for communication networks and in this work OMNeT++ was used. OMNeT++ is an open source, component-based, modular and open-architecture simulation framework written in C++ with good documentation and on-line support. As an *IDE* (Integrated Development Environment) Eclipse CDT is used, so OMNeT++ can be launched at the most popular operation systems – Linux, Mac OS X and Windows.

Developed environment contains four distinguishable modules: generator module, request distribution unit (*RDU*), set of layers with atomic services and a sink module where global statistics are collected and requests are removed from system. Generator module is a complex module and consists of sub-generators each generating requests from different classes. Each class of requests is characterized with parameters describing e.g. service method (based on Integrated services (IntServ) model, Differentiated services (DiffServ) model or best effort [5]), request data size, requests interarrival time, non-functional requirements and others. The last developed component was an adapter which allows to connect simulation environment to a real network [4]. Generated requests are sent to request distribution unit which is responsible for choosing an execution path in system. There is also implemented a request flow shaping mechanism like token bucket and admission control algorithm. Structure of considered system is presented on figure 1.

Assignment of resources is performed to a system which structure is composed by an outer mechanism of composition [2]. One can consider a situation when some atomic services subsets $AS_{ij}$ are executed in parallel or in loop, so different layers can be interpreted as presented on figure 2.

Each atomic service version $a_{ijkl}$ is modelled as a single-queue single processor node but in general it can be multi-queue single processor. Distinction of

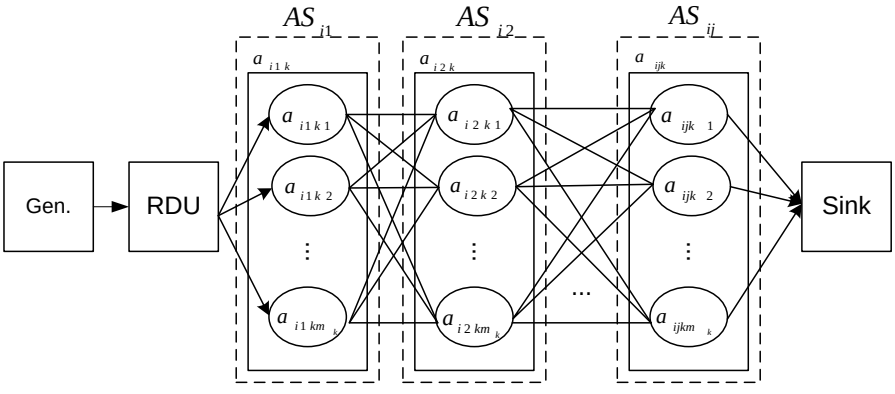



Fig. 1: Structure of considered system with distinguished modules: generator module (*Gen.*), request distribution unit (*RDU*), atomic services structure and sink module.

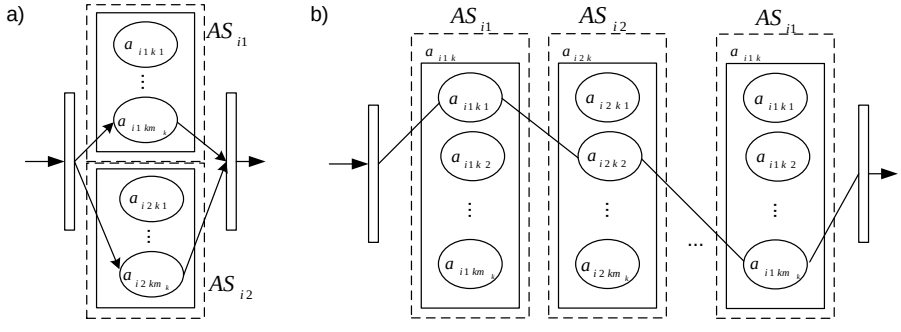



Fig. 2. Exemplary interpretation of atomic services structure: a) parallel, b) single loop

more than one queue can differentiate requests e.g. coming from different classes or being serviced in different way. Noteworthy is also a fact that communication channels can be modelled in flexible way, user can tune such parameters as transfer delay, channel capacity or packet loss ratio.

Last module of presented environment is responsible for removing serviced request from the system and collecting data about the progress of service. Collected data is useful for determination of quality of service level. Configuration of data being collected is easy and one is able to collect additional data if needed.

Simulation environment was designed as a testbed for system resource allocation algorithms (choosing best subset of atomic service versions) so possibility of algorithm replacement without changing simulator structure is a strong advantage.

## 5. Simulation environment usage example

In order to evaluate the quality of service delivered by the considered service-oriented system there were following algorithms of service requests distribution implemented: *IS Greedy*, *BE Greedy*, *BE DAG-SP*, *BE random* and *BE round-robin*. Prefix *BE* in algorithms names means that algorithm deliver quality of service based on *best effort* approach (without any warranty) and *IS* algorithm was based on integrated services model. *IS* algorithm uses reservation mechanism of computational and communication resources so the quality of service can be guaranteed. Repository of reference algorithms based on *Best-effort* approach includes: *BE Greedy* which checks every possibility to choose atomic services versions subset to optimize quality; *BE DAG-SP* (directed acyclic graph – shortest path) which finds shortest path in weighted graph taking into consideration communication delays between atomic services; *BE random* and *BE round-robin* which chooses respectively random and sequential atomic service versions.

To present potential of described tool a comparison of reference algorithms was performed. One can formulate problem as follows: for given current atomic services queue lengths, processing speeds, communication channel delay and request size, find such subset of atomic services that minimizes complex service execution delay.

To examine presented algorithm efficiency the simulation environment was configured as follows. There were three subsets of atomic services each delivering same functionality $n_i = 3$ and each containing three versions of atomic services $\forall a_{ijkl}, j = 1, 2, 3 : k = 1, l = 3$. Communication delay of traffic channels was proportional to the request size which was random with exponential distribution with mean 200 bytes. Atomic services processing speed was also random value with uniform distribution from range 3 to 10 $kbytes/s$. Average complex service execution delays for each considered algorithm are presented on figure 3.

The *IS* algorithm delivered lowest delay because of resource reservation and higher service priority of requests coming from that class than from *best-effort* one. During resource reservation period no *best-effort* service request can be serviced except of situation, when request with resource availability guarantee will leave atomic service earlier than end of reservation period. The best from *best-effort* algorithms group was greedy version of that algorithm. In situation when requests serviced by *IS* algorithms do not share resources with those being serviced by *BE* algorithm both delivers the same delay [3]. Delay delivered by *BE DAG-SP* algorithm was a little higher and quality for *BE round-robin* and *BE random* was much worse than other reference algorithms.

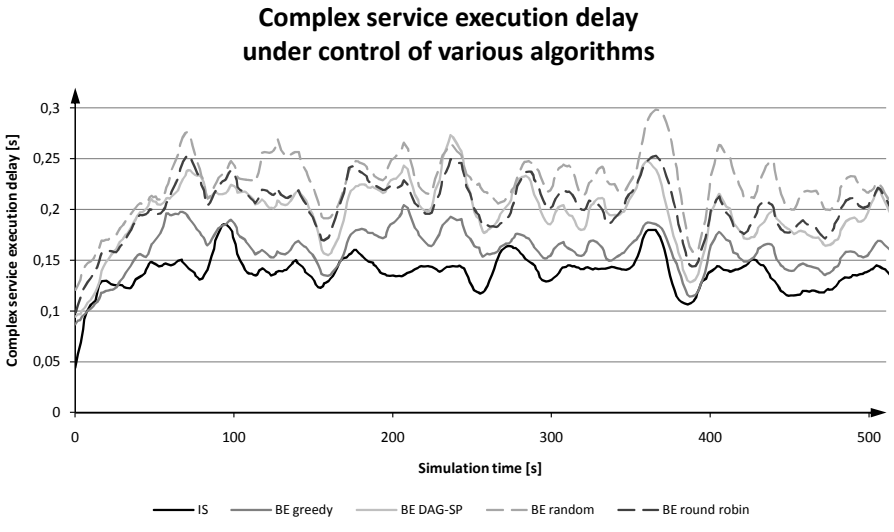



Fig. 3: Results of exemplary simulation run. Comparison of complex service execution delay under control of various algorithms. Requests served with *IS* algorithm were present in system simultaneously with requests serviced by reference *BE* algorithms.

## 6. Final remarks

In this paper model of complex service processing scheme was presented. Distinction of functional and nonfunctional requirements of $SLA$ request was introduced and problem of delivering quality of service was formulated. For the evaluation of performance of the presented system a simulation environment in OMNeT++ was implemented. The simulation environment enables to implement various algorithms of request distribution in the presented system. Moreover the environment allows to use any source of request stream; presented generator can be substituted by component translating real web service server logs or even use stream of requests from real network.

Future plans for development of simulation environment include implementation of new requests distribution algorithms to enlarge algorithms repository and reorganize structure of modelled system to general graph structure with more possible inputs and outputs of a simulated system.

## Acknowledgements

The research presented in this paper has been partially supported by the European Union within the European Regional Development Fund program no. POIG.01.03.01-00-008/08.

## References

[1] L. Borzemski , A. Zatwarnicka, K. Zatwarnicki, The framework for distributed web systems simulation, *Proc. of ISAT 2007, in: Information Technology and Web Engineering: Models, Concepts, and Challenges*, ed. L. Borzemski, Wrocław, 2007, pp. 17–24

[2] K. J. Brzostowski, J. P. Drapała, P. R. Świątek, J. M. Tomczak: Tools for automatic processing of users requirements in soa architecture, *Information Systems Architecture and Technology (ISAT'2009)* "Service oriented distributed systems: concepts and infrastructure", Szklarska Poręba, Poland, September 2009, pp. 137-146

[3] A. Grzech, P. Świątek: Modeling and optimization of complex services in service-based systems, *Cybernetics and Systems* , 40(08), pp. 706–723, 2009.

[4] M. Tüxen, I. Rüngeler, E. P. Rathgeb: Interface connecting the INET simulation framework with the real world, *Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops* ICST, Brussels, Belgium, 2008, pp. 1–6.

[5] Z. Wang: Internet QoS: architecture and mechanisms for Quality of Service, *Academic Press*, 2001.

[6] Wang, G.; Chen, A.; Wang, C.; Fung, C.; Uczekaj, S.: Integrated quality of service (QoS) management in service-oriented enterprise architectures *Enterprise Distributed Object Computing Conference*, 2004. EDOC 2004. Proceedings. Eighth IEEE International Volume , Issue , 20-24 Sept. 2004 pp. 21 - 32

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 99–112

# Decreasing delay bounds for a DiffServ network using leaky bucket shaping for EF PHB aggregates

JANUSZ GOZDECKI [a]

[a]Department of Telecommunications
AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków
gozdecki@agh.edu.pl

**Abstract:** Handling of delay-sensitive traffic in IP networks requires strict control of QoS in DiffServ networks. The models allowing to determine fundamental limitations of traffic aggregation and aggregate-base scheduling have been studied. Application of these models to DiffServ networks, where aggregate-based servicing is a fundamental paradigm, was the base of defining EF PHB. In this paper we prove that hop-by-hop shaping of EF PHB traffic aggregates with a leaky bucket algorithm in a general DiffServ architecture with Guaranteed Rate FIFO scheduling decreases the deterministic edge-to-edge delay bounds if for each link in the network a service rate of an EF traffic aggregate is lower than a throughput of a link. Applying leaky bucket shaping for DiffServ aggregates helps to avoid excessive packet accumulation in nodes and leverages decreasing of delay bounds in case of using such scheduling algorithms as e.g. WFQ, SCFQ, end VC.

**Keywords:** aggregate-based scheduling, DiffServ, EF PHB, leaky bucket, shaping, delay bounds.

## 1. Introduction

Most of the investigations in the area of Next Generation (NG) networks consider Internet Protocol (IP) as the ultimate mean for integrating access networks of different technologies with core networks and as a convergence protocol for versatile applications. Packet-based transport is the preferred solution due to the expected benefits in terms of deployment, management and maintenance costs. However, the departure from a circuit-switched operation towards a packet-based one brings advantages, but also poses substantial difficulties. The Quality of Service (QoS) architecture in IP networks is required to provide the resource reservation guarantees that allow the differentiation and prioritisation of flows, and deployment

of advanced audio and video multimedia services. So far two QoS architectures have been standardised within an Internet Engineering Task Force (IETF) organisation [1]: Integrated Services (IntServ) [2] with a per micro-flow Quality of Service (QoS) guarantees, and Differentiated Services (DiffServ) [3] with aggregate-based QoS mechanisms.

The most crucial problem of QoS provisioning is the efficient support for delay-sensitive applications, with voice communication prevailing, which is a dominant factor for the overall success of the packet-based services. Managing a pure IP system based on DiffServ involves a problem of how to support per-domain services for transport of flow aggregates with a given QoS, especially for a Guaranteed Rate service [4]. Per-domain services support data exchange by mixing traffic of different users and applications in traffic aggregates. Therefore, different aggregates are required to support delay-sensitive traffic and delay-tolerant traffic. Here, it can be necessary to support strict edge-to-edge QoS guarantees using aggregate scheduling, like in a Guaranteed Rate (GR) service [4].

The research on supporting multimedia services in DiffServ networks led to define an Expedited Forwarding Per Hop Behaviour (EF PHB) [5] mechanism for servicing real-time applications. Delay bounds for network services based on EF PHB have been derived. In the paper we propose to use hop-by-hop aggregate-based shaping to decrease delay bounds for network services based on EF PHB. Our findings are fundamental to understand how aggregate-based shaping can influence performance of DiffServ networks. In the paper we present a theorem which is valid for such scheduling algorithms like Weighted Fair Queuing (WFQ), Self-Clocked Fair Queuing (SCFQ), and Virtual Clock (VC) used in networks with FIFO aggregate-base aggregation.

In the next section the state-of-the art and main objectives of the paper are presented. In section 3 a theorem which defines a delay bound for DiffServ network with hop-by-hop EF PHB aggregate leaky bucket shaping is presented . In section 4 the author's delay bound derived in section 3 is compared with the delay bound for a classical DiffServ network. Section 5 concludes the paper. In Appendix A the theorem is derived and proved.

## 2. Delay bounds in networks with Guaranteed Rate aggregate-based scheduling

Strict deterministic control of the end-to-end delay for delay-sensitive applications can be based on mathematical formulation of delay bounds for aggregate-based scheduling proposed by Charny and Le Boudec [5, 6, 8] and Jiang [7]. The referred delay bounds are the base of EF PHB definition in a DiffServ architecture.

Several research proposals aim to obtain better delay bounds for DiffServ networks than the referred above. Is is however done at the expense of more elaborate schedulers while preserving aggregate scheduling. They are mainly based on information carried in a packet and used by a scheduling algorithm in core routers. Examples include a concept of "dampers" [9, 10] and a Static Earliest Time First (SETF) scheduling algorithm [11, 12].

The main objective of this paper is to prove that the application of aggregate-based shaping in a general network architecture with FIFO Guaranteed Rate (GR) [13] aggregate scheduling decreases the deterministic end-to-end delay bound for EF PHB based services in DiffServ networks. The statement is true when a service rate of EF PHB aggregate is lower than throughput of a link what holds for such scheduling algorithms as e.g. WFQ, SCFQ, and VC. Applying a leaky bucket shaping for flows aggregates in each node of the DiffServ domain avoids excessive jitter accumulation in the network. For the derivation of delay bounds in a multi-hop scenario the theory that was originally developed for per-flow scheduling has been adopted for aggregate scheduling.

GR scheduling was defined for per-flow scheduling algorithms but it can be simply extended for class-based aggregate scheduling algorithms when an aggregate of the same class flows in a link is treated as a single flow. In the proposed model GR type scheduler provides the same delay guarantee for the aggregate of flows as provided by the per-flow GR scheduler. An aggregate GR scheduling has been defined in [7].

## 3. Delay bounds for the network with hop-by-hop EF PHB aggregate leaky bucket shaping

In this section delay bounds within the boundaries of a DiffServ network with hop-by-hop EF PHB aggregate leaky bucket shaping are considered. First a model of network architecture is presented. Based on this model we formulate a theorem which describes the end-to-end delay bound for EF PHB based services in a DiffServ network where hop-by-hop EF PHB aggregate leaky bucket shaping is applied.

For the purpose of investigation of the proposed solution it has been assumed a general network topology with all nodes being output-buffered devices implementing First-In-First-Out (FIFO) class-based aggregate scheduling of GR type [13]. The considered network model follows closely the model used in [6] and [7] for comparison purposes. Traffic enters the network at ingress edge nodes and exits it at egress edge nodes. Is is assumed that in the network there is at least one class of edge-to-edge flows served by a GR type scheduler. A flow is a collection of pack-

ets with the common ingress and egress node pair. Packets belonging to one class are buffered in a single queue, which is separate from the queues supporting other classes. Our attention and definitions are focused on one of the classes served by a GR type scheduler, further referred to as a priority class, while other classes are treated as classes of background traffic. All flows belonging to the priority class are referred to as priority flows. All priority flows at a singe link are referred as a priority aggregate. It is assumed that every edge-to-edge priority flow $i$ is shaped at the ingress node to conform to a leaky bucket shaping algorithm with rate $r_i$ and bucket depth $b_i$. Flow $i$ can consist of a number of other flows, however no assumption is made on their characteristics. Let $N$ denote a set of all network nodes $n$, $L$ denote the set of all links $l$ in the network, and $O_l$ denote a set of all priority flows belonging to priority aggregate at link $l$. Finally, we assume that a maximum packet size in our network is bounded by $M$.

The network model defined above corresponds to EF PHB/PDB of DiffServ architecture. In [8, 7] it is proved that the definition of packet scale rate guarantee for EF PHB is stronger than the definition of GR, therefore the delay bounds derived for GR servers apply also to the EF PHB defined in RFC 3246 [5].

Let us assume that for all links $l \in L$ total priority traffic meets the following inequality $\sum_{i \in O_l} r_i < \alpha R_l$, where $R_l$ is a service rate of priority class at link $l$, and $\alpha$ is a coefficient such that $0 \leq \alpha \leq 1$. In addition, it is required that the sum of the leaky bucket depths $\sigma_i$ for the flows traversing any link $l \in O_l$ is bounded by $\sum_{i \in O_i} b_i < \beta R_l$. If in the considered network a leaky bucket shaping algorithm for the priority aggregate is deployed in each link $l \in L$ with parameters $(\rho_l = \eta C_l, \sigma_l = \epsilon C_l)$, then the following theorem is true:

**Theorem 1** *If at each link $l \in L$*

$$\rho_l \geq R_l \tag{1}$$

*or*

$$\alpha R_l \leq \rho_l < R_l \tag{2}$$

*and*

$$\frac{\sigma_l R_l}{R_l - \rho_l} > \frac{R_l \left((H-1)T_l\alpha + \beta\right)\rho_n + \left(\alpha(H-1)\left(u_l + \rho_n - u_l\rho_n\right) - 1\right)\sigma_n}{\left((H-1)u_l\alpha - 1\right)\left(\rho_n - R_l\alpha\right)} \tag{3}$$

*and common conditions are true:*

$$\sigma_n < \frac{R_l\left((H-1)T_l\alpha + \beta\right)\left(P_n - \rho_n\right) + M_n\left(\rho_n + \alpha\left((H-2)R_l - \rho_n(H-1)\right)\right)}{(H-2)R_l\alpha + P_n(1 - \alpha(H-1))} \tag{4}$$

*and*

$$\alpha < \min_{l \in L} \left\{ \frac{P_n}{(P_n - R_l)(H-1) + R_l}, \frac{\rho_n}{(\rho_n - R_l)(H-1) + R_l} \right\} \tag{5}$$

*then the bound on the edge-to-edge queuing delay $D$ for priority flows is:*

$$D \leq \frac{H}{1 - v\alpha(H-1)}(E' + v\beta) \tag{6}$$

where:

$$E' = \max_l \left\{ \frac{\sigma_n (1 - v_l)}{R_l} + T_l \right\}, \tag{7}$$

$$v = \max_{l \in L} \left\{ v_l = \frac{(\rho_n - R_l)^+}{\rho_n - \alpha R_l} \right\}, \tag{8}$$

$$T_l = \frac{M_l}{R_l} + E_l. \tag{9}$$

Variable $H$ defines maximum number of hops over a network. $P_n$ is a peak rate from all incoming interfaces in the node $n$. For a router with large internal speed it can be calculated as $P_n = \sum_{l=1}^{k_n} C_l$ where $C_l$ is throughput of an incoming link $l$ to the node $n$, $k_n$ is a number of incoming links to the node $n$. When one cannot estimate the peak rate, it can be assumed that $P_n = \infty$. $M_n = \sum_{l=1}^{k_n} M_l$ and $M_l$ is a maximum packet size on link $l$. As in the considered model a packet size is bounded by $M$, then $M_n = k_n M$. Operator $(\cdot)^+$ is defined such that: $(a)^+ = 0$ for all $a < 0$ and $a$ for $a \geq 0$. $\sigma_n = \sum_{l=1}^{k_n} \sigma_l$ and $\rho_n = \sum_{l=1}^{k_n} \rho_l$. $E_l$ is a constant, which depends on scheduling algorithm used in the link $l$, e.g. for strict priority scheduling and for the highest priority queue $E_l = \frac{M_l}{C_l}$, where $C_l$ is a speed of a link $l$, for a WFQ scheduler $E_l = \frac{M_l}{R_{WFQ}}$, where $R_{WFQ}$ is a service rate guaranteed for the priority class.

The proof of Theorem 1 is presented in Appendix A. To derive delay bounds Network Calculus [14, 9] theory is used.

In the next section we prove that the delay bound defined in Theorem 1 is lower than the delay bound in classical DiffServ network when $R_l < \rho_l < C_l$. The delay bound for the classical DiffServ network is bounded by [7]:

$$D \leq \frac{H}{1 - (H-1)\alpha u}(E + u\beta) \tag{10}$$

where $u = \max_{l \in L} \left\{ u_l = \frac{(P_n - R_l)^+}{P_n - \alpha R_l} \right\}$, and $E = \max_{l \in L} \left\{ \frac{(1-u_l)M_n}{R_l} + \frac{M}{R_l} + E_l \right\}$. The delay bound defined by inequality (10) is valid provided that resource utilisation factor $\alpha$ for all links in a DiffServ domain is controlled and does not exceed the pre-calculated value [5, 6, 7]:

$$\alpha < \min_{l \in L} \left\{ \frac{P_n}{(P_n - R_l)(H-1) + R_l} \right\} \tag{11}$$

Remaining variables used in inequalities (10) and (11) are the same as in the our model.

In the next section the comparison between delay bounds defined in the author's Theorem 1 and in inequality (10) by Jiang is presented.

## 4. Comparing delay bounds

In this section a comparison of the delay bound defined in the author's Theorem 1 with the delay bound defined in inequality (10) by Jiang is provided. When comparing inequalities (10) with (6) it is easy to notice that they differ only in two parameters: in the inequality (10) there are $E$ and $u$ parameters which are equivalent to $E'$ and $v$ parameters in inequality (6). Likewise variables $E$ and $E'$ are similar to each other (the difference is in variables $u$ and $M_n$ which correspond to variables $v$ and $\sigma_n$). Both equations defining variables $E$ and $E'$ are increasing functions against variables $u$ and $v$. The similar property stays with variables $u$ and $v$ which differ only in one parameter – $P_n$ versus $\rho_n$. It means that if we assume that $M_l = \sigma_l$ then the values of variables $P_n$ and $\rho_n$ decide which inequality would have the greater value, inequality (10) or inequality (6). So, if we want to have the lower delay bound within boundaries of a DiffServ network with the leaky bucket aggregate shaping than within boundaries of the corresponding classical DiffServ network, the inequality $\rho_n < P_n$ has to be true. Based on definition of variables $\rho_n$ and $P_n$ inequality $\rho_n < P_n$ is equivalent to $\rho_l < C_l$. Taking into account inequality (1), to lower the delay bound in DiffServ network with aggregate leaky bucket shaping, the following inequality has to be true at each link $l \in L$:

$$R_l < \rho_l < C_l \tag{12}$$

The inequality (12) means that for each $l \in L$ service rate $R_l$ of priority aggregate has to be lower than link throughput $C_l$. This is true e.g. for such GR scheduling algorithms like WFQ, SCFQ and VC. The gain from using aggregate leaky bucket shaping increases when value of $\rho_l$ decreases.

In Tab. 1 example network parameters are presented. This parameters are used to visualise the comparison between inequality (10) and inequality (6) in Fig. 1,

| Parameter | $R_l$ [b/s] | $C_l$ [b/s] | $M$ [B] | $P_n$ [b/s] | $M_n$ [B] | $H$ |
|---|---|---|---|---|---|---|
| Value | $5 \cdot 10^5$ | $10^9$ | 1500 | $6 \cdot 10^9$ | 9000 | 8 |
| Parameter | $E_l$ [s] | $T_l$ | $\epsilon$ [s] | $\eta$ | $\alpha$ | $\beta$ [s] |
| Value | $12 \cdot 10^{-6}$ | $36 \cdot 10^{-6}$ | $12 \cdot 10^{-6}$ | 0.505 | 0.149 | $24 \cdot 10^{-4}$ |

Table 1. Example network parameters

Fig. 2, and Fig. 3. Figure 1 presents comparison of edge-to-edge delay bounds between inequality (10) and inequality (6) in function of $\alpha$. In this figure the delay bound for the network with aggregate leaky bucket shaping is lower than in the corresponding classical DiffServ network, for all $\alpha$ values, and the difference is from $6.5\%$ for $\alpha = 0.01$ to $90\%$ for $\alpha = 0.149$.

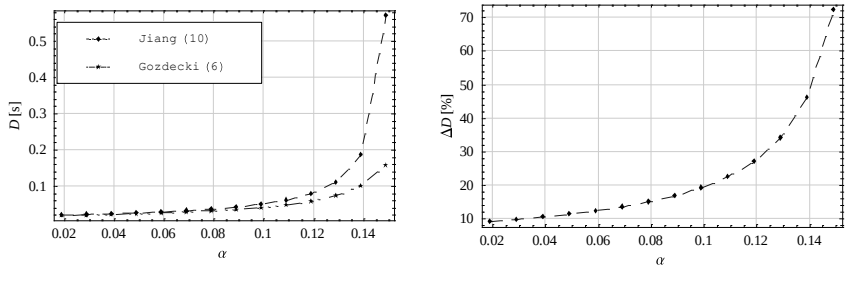



Fig. 1: Comparison of the edge-to-edge delay bounds for the EF aggregate flows as a function of $\alpha$; left figure – absolute values, right figure – variance in $\%$

Fig. 2 presents comparison of edge-to-edge delay bounds between inequality (10) and inequality (6) in function of $H$. In this figure the delay bound for the network with aggregate leaky bucket shaping is also lower than in the corresponding classical DiffServ network, for all $H$ values, and the difference is about from $7\%$ for $H = 2$ to $70\%$ for $H = 8$.

Fig. 3 presents comparison of edge-to-edge delay bounds between inequality (10) and inequality (6) in function of $H$, but in case of this figure for each $H$ value also $\alpha$ value is changing, according to function $\alpha(H) = 0.99\frac{P_n}{(P_n - R_l)(H-1)+R_l}$. In this case the difference between the delay bound for the network with aggregate leaky bucket shaping is also lower than in the corresponding classical DiffServ network, but the difference is hight for all $H$ values and is more than $80\%$ with one exception, for $H = 2$ the difference is about $20\%$.

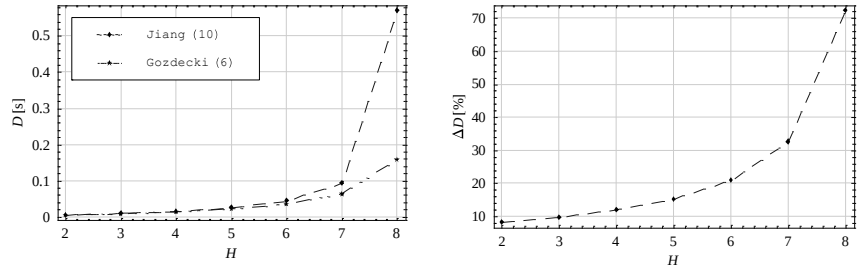



Fig. 2: Comparison of the edge-to-edge delay bounds for the EF aggregate flows as a function of maximum number of hops $H$; left figure – absolute values, right figure – variance in $\%$

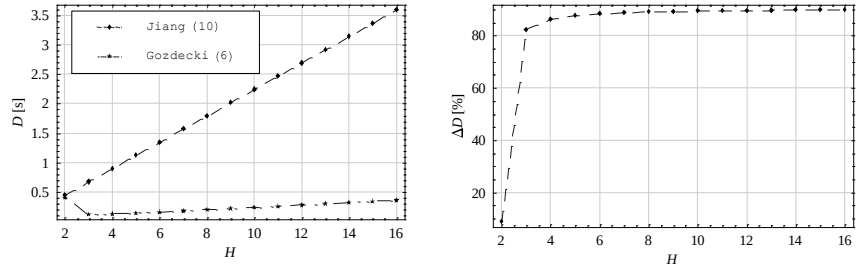



Fig. 3: Comparison of the edge-to-edge delay bounds for the EF aggregate flows as a function of maximum number of hops $H$, where $\alpha(H) = 0.99\frac{P_n}{(P_n - R_l)(H-1)+R_l}$; left figure – absolute values, right figure – variance in $\%$

## 5. Conclusions

In the paper application of aggregate leaky bucket shaping to decrease delay bounds for DiffServ network services based on EF PHB is proposed. It is shown that the application of hop-by-hop aggregate-based shaping to a general network architecture with FIFO GR aggregate scheduling enables decreasing the deterministic end-to-end delay bound in DiffServ networks. This is true in the case when a service rate of EF PHB aggregates in links is lower than throughput of links in the whole network. The above constraint is true for such scheduling algorithms like WFQ, SCFQ, and VC. Numerical comparison of the delay bounds for the example network shows that aggregate leaky bucket shaping can decrease the delay bound more than $80\%$. The decreasing of delay bound is in the expense of leaky bucket shaping implementation for an EF PHB aggregate in each node, what has no influence on scalability of network, but only little increases computation load in network nodes. Further research of presented approach will be concerned with more sound investigation of special cases.

*The work was supported by Poland Ministry of Science and Higher Education through the grant N N 517 228135.*

## References

[1] The Internet Engineering Task Force (IETF), *www.ietf.org*.

[2] R. Braden, D. Clark, S. Shenker, RFC 1633 – Integrated Services in the Internet Architecture: an Overview, June 1994.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, RFC 2475 – An Architecture for Differentiated Services, December 1998.

[4] S. Schenker, C. Partridge, R. Guerin, Specification of Guaranteed Quality of Service, *RFC 2212*, September 1997.

[5] B. Davie, A. Charny, F. Baker, J.C.R. Bennett, K. Benson, J.Y. Le Boudec, A. Chiu, W. Courtney, S. Davari, V. Firoiu, C. Kalmanek, K. Ramakrishnan, and D. Stiliadis, "An Expedited Forwarding PHB (Per-Hop Behavior)", RFC 3246, March 2002.

[6] A. Charny, J.-Y. Le Boudec, Delay Bounds in a Network with Aggregate Scheduling, *Proceedings of QOFIS*, Berlin, October 2000.

[7] Y. Jiang, Delay Bounds for a Network of Guaranteed Rate Servers with FIFO Aggregation, *Proceedings of ICCC 2002*, New York, May 2002.

[8] J. C. R. Bennett, K. Benson, A. Charny, W. F. Courtney, J.-Y. Le Boudec: Delay Jitter Bounds and Packet Scale Rate Guarantee for Expedited Forwarding, *ACM/IEEE Transactions on Networking*, August 2002, Vol. 10, No. 4, pp. 529–540.

[9] J.-Y. Le Boudec, P. Thiran: Network Calculus, *Springer Verlag LNCS 2050*, June 2001.

[10] R. L. Cruz, SCED+: efficient management of quality of service guarantees, *IEEE Infocom'98*, San Francisco, March 1998.

[11] Z. L. Zhang, Z. Duan, T. Y. Hou: Fundamental trade-offs in aggregate packet scheduling, *SPIE Vol. 4526*, August 2001.

[12] D. Verma, H. Zhang, D. Ferrari: Guaranteeing delay jitter bounds in packet switching networks, *Proceedings of Tricomm '91*, Chapel Hill, April 1991, pp. 35–46.

[13] P. Goyal, H. Vin, Generalized guaranteed rate scheduling algorithms: a framework, *IEEE/ACM Trans. Networking*, vol 5-4, August 1997, pp. 561–572.

[14] J.-Y. Le Boudec: Application of Network Calculus To Guaranteed Service Networks, *IEEE Transactions on Information Theory*, May 1998, Vol. 44 No. 3, pp. 1087–1096.

[15] V. Firoiu, J.-Y. Le Boudec, D. Towsley, Z.-L. Zhang, Theories and Models for Internet Quality of Service, *Proceedings of the IEEE, special issue in Internet Technology*, September 2002, Vol. 90 No. 9, pp. 1565–1591.

## A Appendix - Proof of Theorem 1

The proof is based on reasoning presented in [6, 13], and consists of showing in *Part 1*, that if a finite bound exists, then the main formula in Theorem 1 is true, and then in *Part 2*, that the finite bound does exist.

***Part 1***: In the first part of the proof we use Network Calculus (NC) [14, 9] theory to derive the worst case of delay bound within the boundaries of a network. The NC theory is a mathematical tool for networks bounds calculation. First we derive the formula describing the arrival curve $A(t)$ of incoming flows to a node and the service curve $S(t)$ of a node. Then the delay bound is determined based on horizontal deviation $h(A, S)$ [14] between the arrival and service curves.

**Determining arrival and service curves**

We assume that for any link the delay bound $d_l$ exists. Let $a_i(t)$ be an arrival curve [14] for a single priority flow $i$ at the network ingress. Because each flow $i$ is shaped by a leaky bucket algorithm with parameters $(r_i, b_i)$, then the arrival curve is defined as $a_i(t) = r_i t + b_i$. An arrival curve $A'(t)$ for a priority aggregate at an output of edge link $l$ in our network model can be expressed as

$$A'(t) = min \left\{ \sum_{i \in O_l} r_i t + b_i, C_l t + M_l, \rho_l t + \sigma_l \right\}. \tag{13}$$

The first argument of operator $min(\cdot)$ in formula (13) represents the aggregation of priority flows. The second argument is because the total incoming rate is limited by link throughput $C_l$ with accuracy $M_l$. The last argument of operator $min(\cdot)$ is because aggregate leaky bucket shaping with parameters $(\rho_l, \sigma_l)$ at link $l$. Taking into account assumptions to Theorem 1 $\alpha R_l < \sum_{i \in O_l} r_i$ and $\sum_{i \in O_i} b_i < \beta R_l$, the equation (13) can be rewritten as:

$$A'(t) = min \left\{ \alpha R_l t + \beta R_l, C_l t + M_l, \rho_l t + \sigma_l \right\}. \tag{14}$$

Lets assume that the worst case delay $d_l$ happens in link $l$ in the network. Then arrival curve $A(t)$ at link $l$ is as follows:

$$A(t) = min \left\{ \alpha R_l (t + (H-1) d_l) + \beta R_l, P_n t + M_n, \rho_n t + \sigma_n \right\}. \tag{15}$$

Where the component $(H-1)d_l$ in the first argument of $min(\cdot)$ is because any flow reaching link $l$ experienced delay $(H-1)d_l$. $P_n = \sum_{l=1}^{k_n} C_l$, $M_n = \sum_{l=1}^{k_n} M_l$, $\sigma_n = \sum_{l=1}^{k_n} \sigma_l$ and $\rho_n = \sum_{l=1}^{k_n} \rho_l$. In our network aggregates at each link are serviced by two servers: a GR type scheduler and a leaky bucket shaper. The service curve of a GR server at link $l$, based on Theorem 2 in [14], is

$$S_{GR}(t) = R_l \left(t - T_l\right)^+, \tag{16}$$

where $T_l$ is defined by equation (9). A service curve of a leaky bucket algorithm, which is applied to shape the priority aggregate, at each link $l \in L$ of our network has the following service curve [14]:

$$S_{LB}(t) = R_l \begin{cases} 0 & \text{for} \quad t \leq 0 \\ \rho_l t + \sigma_l & \text{for} \quad t > 0 \end{cases}, \tag{17}$$

Because both servers work in sequence then the resulting service curve $S(t)$ at link $l$ can be obtained as min-plus convolution of $S_{GR}(t)$ and $S_{LB}(t)$ service curves [14, 9]:

$$S(t) = (S_{GR} \otimes S_{LB}). \tag{18}$$

After some algebraic calculations $S(t)$ becomes:

$$S(t) = U(t_{T_l}) min\left\{\rho_l(t - T_l) + \sigma_l, R_l(t - T_l)\right\}, \tag{19}$$

where $U(t_a) = \begin{cases} 0 & \text{for} \quad t \leq a \\ 1 & \text{for} \quad t > a \end{cases}$.

**Calculating delay bound**

The delay bound $d_l$ in node $l$ is derived by calculating of horizontal deviation $h(\cdot)$, defined by the equation (7) in [14], between the arrival curve $A(t)$ - equation (15) – and the service curve $S(t)$ defined by the equation (19):

$$d_l \leq h\left(A(t), S(t)\right) = \left(sup_{t \geq 0}[inf\{\text{such that } A(t) \leq S(t+d)\}\right). \tag{20}$$

The solution for the inequality (20) expresses the worst case delay $d_l$ at link $l$ on our network. Theorem 1 is a special case solution of (20), where horizontal deviation is measured between curves $A(t) = min\left\{\alpha R_l(t + (H-1)d_l) + \beta R_l, \rho_n t + \sigma_n\right\}$ and $S(t) = R_l(t - T_l)^+$ - Figure 4:

$$d_l \leq h\left(min\left\{\alpha R_l(t + (H-1)d_l) + \beta R_l, \rho_n t + \sigma_n\right\}, R_l(t - T_l)^+\right). \tag{21}$$

To solve inequality between (20) and (21), the following conditions have to be met:

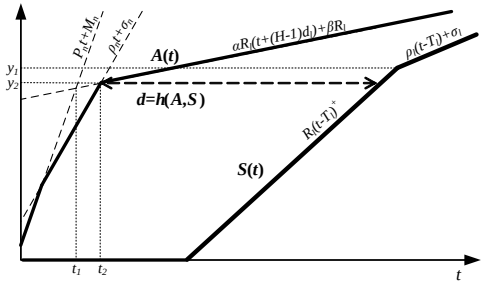



Fig. 4. Arrival curve $A(t)$ and service curve $S(t)$ for Theorem 1

1. The inclination of curve $\alpha R_l(t + (H-1)d_l) + \beta R_l$ has to be lower than the inclination of curves $\rho_n t + \sigma_n$ and $P_n t + M_n$. This is true because based on equations (1) and (2) of Theorem 1 we know that $\rho_n > \rho_l \geq \alpha R_l$ and because according to our network model $P_n > \alpha R_l$.
2. The value of abscissa $t_1$ at the intersection point between curves $\alpha R_l(t + (H-1)d_l) + \beta R_l$ and $P_n t + M_n$ has to be lower than the value of abscissa $t_2$ at the intersection point between curves $\alpha R_l(t + (H-1)d_l) + \beta R_l$ and $\rho_n t + \sigma_n$. This corresponds to condition (4) of Theorem (1).
3. The value of ordinate $y_1$ at the intersection point between curves $\alpha R_l(t + (H-1)d_l) + \beta R_l$ and $\rho_n t + \sigma_n$ has to be lower than the value of ordinate $y_2$ at the intersection point between curves $\alpha R_l(t + (H-1)d_l) + \beta R_l$ and $P_n t + M_n$. This it corresponds to condition (3) of Theorem 1.

Based on condition 2, the author derives equation (4) of Theorem 1. The values of abscissas $t_1$ and $t_2$ are: $t_1 = \frac{\alpha R_l(H-1)d_l + \beta R_l - M_l}{P_n - \alpha R_l}$, $t_2 = \frac{\alpha R_l(H-1)d_l + \beta R_l - \sigma_n}{\rho_l - \alpha R_l}$. According to condition 2 the following inequality has to be met:

$$t_2 - t_1 = \frac{\alpha R_l(H-1)d_l + \beta R_l - \sigma_n}{\rho_l - \alpha R_l} - \frac{\alpha R_l(H-1)d_l + \beta R_l - M_l}{P_n - \alpha R_l} \geq 0. \quad (22)$$

Because at each link $l$, $\rho_l \geq \alpha R_l$ the solution of inequality (22) is as follows:

$$\sigma_n \leq \frac{\alpha R_l((P_n - \rho_n)(H-1)d_l + \beta) + M_n(\rho_n - \alpha R_l}{P_n - \alpha R_l}, \quad (23)$$

where $d_l$ is the solution of equation (21). The final form of inequality (23) will be presented later.

Based on condition 3, the author derives equation (3) of Theorem 1. The values of ordinates $y_1$ and $y_2$ are: $y_1 = S(\frac{\sigma_l}{R_l - \rho_l} + T_l) = \frac{\sigma_l R_l}{R_l - \rho_l}$, $y_2 =$

$A(\frac{\alpha R_l(H-1)d_l+\beta R_l-\sigma_n}{\rho_n-\alpha R_l}) = \frac{\rho_n(\alpha R_l(H-1)d_l+\beta R_l)-\sigma_n\alpha R_l}{\rho_n-\alpha R_l}$. According to condition 3, the following inequality has to be met:

$$\frac{\sigma_l R_l}{R_l-\rho_l} \geq \frac{\rho_n(\alpha R_l(H-1)d_l+\beta R_l)-\sigma_n\alpha R_l}{\rho_n-\alpha R_l}. \tag{24}$$

Lets rewrite the equation (21) to the form that equation (8) from [15] can be used directly to derive the delay bound:

$$d_l \leq d' = h\left(min\left\{\alpha R_l t + (\alpha R_l(H-1)d_l) + \beta R_l), \rho_n t + \sigma_n\right\}, R_l(t-T_l)^+\right). \tag{25}$$

Based on equation (8) from [15]:

$$d_l \leq \frac{\sigma_n + \frac{(\alpha R_l(H-1)d_l)+\beta R_l-\sigma_n)(\rho_n-R_l)}{\rho_n-\alpha R_l}}{R_l} + T_l. \tag{26}$$

Moving $d_l$ on the left side of inequality (26) and defining $v_l = \frac{(\rho_n-R_l)^+}{\rho_n-\alpha R_l}$ we have:

$$d_l(1-\alpha(H-1)v_l) \leq \frac{\sigma_n - v_l\sigma_n + \beta R_l v_l}{R_l} + T_l. \tag{27}$$

The expression on the left side of the inequality must be greater than 0 for inequality (27) to make sense. Solving inequality $(1-\alpha(H-1)v_l) > 0$ the author derives condition on $\alpha$:

$$\alpha \leq \frac{1}{(H-1)v_l}. \tag{28}$$

After some algebraic manipulations of equation (27) we have:

$$d_l \leq \frac{\sigma_n - v_l\sigma_n + \beta R_l v_l + R_l T_l}{(1-\alpha(H-1)v_l)R_l}. \tag{29}$$

The worst case edge-to-edge delay in our network is:

$$D \leq H max_l\left\{d_l\right\}. \tag{30}$$

Designating $v = max_l\left\{v_l\right\}$ and $E' = max_l\left\{\frac{(1-v_l)\sigma_n}{R_l} + \frac{M_l}{R_l} + E_l\right\}$, and after some algebraic calculations inequality (30) becomes:

$$D \leq \frac{H}{1-(H-1)\alpha v}(E' + v\beta). \tag{31}$$

Inequality 31 is compliant with equation (6), of Theorem 1. Applying equation (29) to equations (23) and (24) the author obtains the following conditions on network parameters:

$$\frac{\sigma_l R_l}{R_l-\rho_l} > \frac{R_l\left((H-1)T_l\alpha+\beta\right)\rho_n + \left(\alpha(H-1)\left(u_l+\rho_n-u_l\rho_n\right)-1\right)\sigma_n}{\left((H-1)u_l\alpha-1\right)\left(\rho_n-R_l\alpha\right)} \tag{32}$$

and

$$\sigma_n < \frac{R_l\left((H-1)T_l\alpha+\beta\right)\left(P_n-\rho_n\right)+M_n\left(\rho_n+\alpha\left((H-2)R_l-\rho_n(H-1)\right)\right)}{(H-2)R_l\alpha+P_n(1-\alpha(H-1))}. \tag{33}$$

The inequalities (32) and (33) are compliant with conditions (3) and (4) of Theorem 1. Because inequality (32) is a Möbius transformation function of $\alpha$ (for values of network parameters that can be assigned in real networks), it has an asymptote crossing an abscissa axis at point $\frac{P_n}{(P_n-R_l)(H-1)+R_l}$ and the value of parameter $\alpha$ has to be lower than the value of this point. Taking into account equation (28) and the above properties of $\alpha$ the final form of constrain on $\alpha$ is as follows:

$$\alpha < \min_{l\in L}\left\{\frac{P_n}{\left(P_n-R_l\right)\left(H-1\right)+R_l}, \frac{\rho_n}{\left(\rho_n-R_l\right)\left(H-1\right)+R_l}\right\} \tag{34}$$

This inequality is compliant with condition (5) in Theorem 1. This finishes the proof of Theorem 1, where the author has proved that if the constraints to Theorem 1 are true, then the maximum delay between the edges of our network is bounded by (31).

***Part 2:*** Here we give a proof that the finite bound does exist. This proof is similar to one in [6, 7] and uses time-stopping method [13]. For any time $t$ consider a virtual system made of our network. All sources in the virtual system are stopped at time $t$. The virtual system satisfies the assumptions of Part 1, since the amount of traffic at the output is finite. Suppose $d_l$ is the worse case delay across the all nodes for the virtual system indexed by $t$. From Part 1 we have $d_l < d'$ for all $t$. Letting $t \to \infty$ shows that worse case delay remains bounded by $d'$.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 113 –120

# A Markov model of multi-server system with blocking and buffer management

WALENTY ONISZCZUK

Computer Science Faculty
Bialystok Technical University
*w.oniszczuk@pb.edu.pl*

**Abstract:** This paper is aimed at designing a mathematical model of active buffer management in multi-server computer systems. The use of adequate buffer management scheme with thresholds and blocking well-known techniques for computer systems traffic congestion control. This motivates the study of multi-server computer systems with two distinct priority classes (high and low priority traffics), partial buffer sharing scheme with thresholds and with blocking. Adaptive buffer allocation algorithm (scheme) is designed to allow the input traffic to be portioned into different priority classes and based on the input traffic behaviour it controls the thresholds dynamically. Using an open Markov queuing schema with blocking, and thresholds, a closed form cost-effective analytical solution for this model of computer system is obtained.

**Keywords:** Markov chains, blocking, threshold-base systems

## 1. Introduction

Finite capacity queuing network models (QNMs) are of great value towards effective congestion control and quality of service (QoS) protection of modern discrete flow computer systems. Blocking in such systems arises because the traffic of jobs may be momentarily halted if the destination buffer has reached its capacity. As a consequence, cost-effective numerical techniques and analytic approximations are needed for study of complex queuing networks. The traditional analyses of the ONMs with blocking are based on the Markov Chain approach [2]. In addition, many interesting theories and models appeared in a variety of journals and at worldwide conferences in the field of computer science, traffic engineering and communication engineering [1, 7]. The introduction of flexible buffer management policy with thresholds can give rise to inter-dependency between the thresholds setting, which are also dependent on the blocking phenomena. In this context, congestion control through adequate buffering and blocking is becoming particularly significant to minimize the job

delay and providing some acceptable level of system utilization. A proper congestion control [3, 5, 6] is needed to protect low priority jobs from long delay time by reducing an arrival rate to the buffer for high priority jobs or not sending the job to the buffer (blocking). The use of thresholds and blocking for controlling congestion in computer system buffers is well known and used. Congestion control based on thresholds and blocking [4] is aimed to control the traffic-causing overload and so to satisfy the Quality of Service (QoS) requirements of the different classes of traffic.

The main aim of this paper is to formulate such a model with a dynamical buffer allocation policy and examine the queuing behaviour under a priority service discipline and blocking mechanisms.

## 2. Model description

The general model description is:

- There are two source stations, which generate high and low priority jobs.
- The arrival processes from source stations are Poisson, with rates $\lambda_1$ and $\lambda_2$.
- Service station consists of a multi-server and a common buffer.
- Servers provide exponentially distributed service with rate $\mu$.
- Buffer has finite capacity *m,* with dynamically changed thresholds *m1* and *m2*.
- Controller – a decision maker agent.

Fig. 1 presents a simplified multi-server network description of the proposed model. The jobs via the controller arrive from the source stations at servers station buffer. The controller with Partial Buffer Sharing (PBS) scheme controls incoming traffic from different priority classes based on thresholds in buffer. When the buffer level is below a forward threshold *m2,* controller accepts both high priority and low priority jobs and when the number of jobs exceeds this level, low priority jobs cannot access the buffer and the Source2 station is blocked. Whenever the number of jobs falls below a reverse threshold *m1*, the transmission process from Source2 is resumed.

For high priority traffic the complete buffer is accessible irrespective of the buffer occupancy level and thresholds value. When the buffer is full, the accumulation of new jobs from the Source1 is temporarily suspended and another blocking occurs, until the queue empties and allows new inserts. The goal is to accommodate more incoming jobs from various sources. The dynamical threshold scheme adapts to changes in traffic conditions.

A controlled computer system (see Figure 1) consists of three components: servers, buffer, and controller. Controller may improve system utilization by reducing expected throughput time or queue length. A decision maker, agent, or

controller is forced with the problem of influencing the behaviour of a probabilistic system as it evolves through time. He does this by making decisions (blocking) or choosing actions (increasing or decreasing thresholds value). Generally, a controller regulates the computer systems load by accepting or blocking arriving jobs and by dynamically changing the threshold levels if blocking action is indicated. It means that both threshold *m1* and *m2*, starting from some initial value, are dynamically modified. For each class of jobs, the controller counts the number of accepted and blocked jobs. Each controller counter is assigned two parameters to control the thresholds: blocking ratio (a relation of the number blocked to accepted jobs) $p_h$ for high priority jobs and $p_l$ for low priority and modification step threshold levels. The blocking ratio for high priority jobs when reaches its limit $p_h$ the threshold *m1* is made to decrease and the threshold *m2* to increase by modification step level. Similarly when the blocking ratio for low priority jobs reaches the limit $p_l$ the threshold *m1* is made to increase and the threshold *m2* to decrease by modification step level.

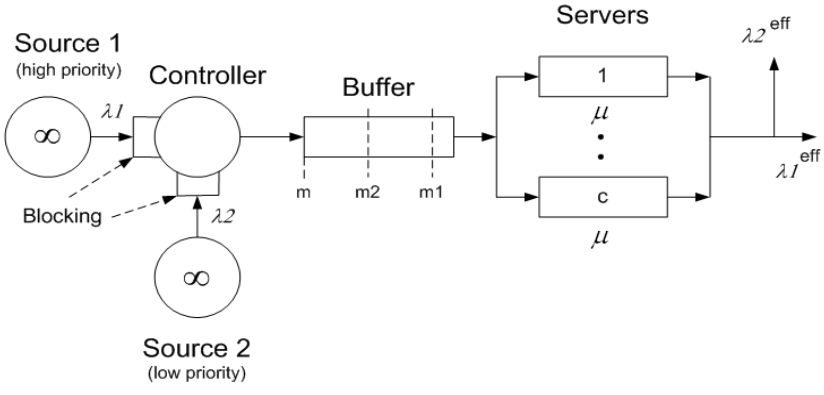



Fig.1. Illustration of the computer system model with blocking, and thresholds.

## 3. Performance analysis

Each queuing system can, in principle, be mapped onto an instance of a Markov process and then mathematically evaluated in terms of this process. For analyzing the performance of the proposed model, the queuing system can be represented by a continuous-time Markov chain, in which the underlying Markov process can analyze the stationary behaviour of the network. If a queue has finite capacity, the underlying process yields finite state space. The solution of the Markov chain representation may then be computed and the desired performance characteristics, such as blocking probabilities, utilization, and throughput, obtained directly from the stationary probability vector.

In theory, any Markov model can be solved numerically. In particular, solution algorithm for Markov queuing networks with blocking, and thresholds is a three-step procedure:

1. Definition a state space representation and enumerating all the transitions that can possible occur among the states.
2. Solution of linear system of the global balance equations to derive the stationary state distribution vector.
3. Computation from the probability vector of the average performance indices.

The state of the queuing network with blocking, and thresholds (see Fig. 2) can be described by random variables (*i, k*), where *i* indicate the number of jobs at the service station, and *k* represents its state. Here, the index *k* may have the following values: *0* - idle system, *1* - regular job service, *2* - high priority service and blocking low priority jobs, *3* - blocking high priority jobs.

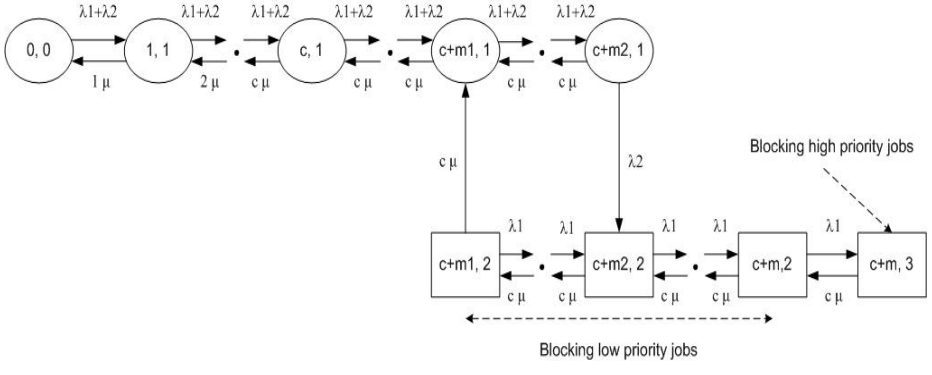



Fig. 3. State transition diagram of a two dimensional Markov chain for modelling the flexible buffer sharing scheme.

Let $p_{i,k}$ denote the joint probability of state (*i, k*) in the two-dimensional Markov chain. Based on an analysis the state transition diagram, the process of constructing the steady-state equations in the Markov chain can be divided into several independent step. These steady-state equations are:

$$
\begin{aligned}
&(\lambda_1+\lambda_2)\cdot p_{0,1} = \mu \cdot p_{1,1} \\
&(\lambda_1+\lambda_2+i\cdot\mu)\cdot p_{i,1} = (\lambda_1+\lambda_2)\cdot p_{i-1,1} + (i+1)\cdot\mu\cdot p_{i+1,1} \quad \text{for } i = 1, \ldots, c\text{-}1 \\
&(\lambda_1+\lambda_2+c\cdot\mu)\cdot p_{i,1} = (\lambda_1+\lambda_2)\cdot p_{i-1,1} + c\cdot\mu\cdot p_{i+1,1} \quad \text{for } i = c, \ldots, c+m1\text{-}1 \\
&(\lambda_1+\lambda_2+c\cdot\mu)\cdot p_{c+m1,1} = (\lambda_1+\lambda_2)\cdot p_{c+m1-1,1} + c\cdot\mu\cdot p_{c+m1+1,1} + c\cdot\mu\cdot p_{c+m1,2} \quad (1) \\
&(\lambda_1+\lambda_2+c\cdot\mu)\cdot p_{i,1} = (\lambda_1+\lambda_2)\cdot p_{i-1,1} + c\cdot\mu\cdot p_{i+1,1} \quad \text{for } i = c+m1+1, \ldots, c+m2\text{-}1 \\
&(\lambda_2+c\cdot\mu)\cdot p_{c+m2,1} = (\lambda_1+\lambda_2)\cdot p_{c+m2-1,1}
\end{aligned}
$$

For states with blocking the equations are:

$$
\begin{aligned}
&(\lambda_1 + c \cdot \mu) \cdot p_{c+m1,2} = c \cdot \mu \cdot p_{c+m1+1,2} \\
&(\lambda_1 + c \cdot \mu) \cdot p_{i,2} = \lambda_1 \cdot p_{i-1,2} + c \cdot \mu \cdot p_{i+1,2} \quad \text{for } i = c+m1+1, \ldots, c+m2-1 \\
&(\lambda_1 + c \cdot \mu) \cdot p_{c+m2,2} = \lambda_1 \cdot p_{c+m2-1,2} + \lambda_2 \cdot p_{c+m2,1} + c \cdot \mu \cdot p_{c+m2+1,2} \\
&(\lambda_1 + c \cdot \mu) \cdot p_{i,2} = \lambda_1 \cdot p_{i-1,2} + c \cdot \mu \cdot p_{i+1,2} \quad \text{for } i = c+m2+1, \ldots, c+m-1 \\
&(\lambda_1 + c \cdot \mu) \cdot p_{c+m,2} = \lambda_1 \cdot p_{c+m-1,2} + c \cdot \mu \cdot p_{c+m,3} \\
&c \cdot \mu \cdot p_{c+m,3} = \lambda_1 \cdot p_{c+m,2}
\end{aligned}
\tag{2}
$$

Here, a queuing network with blocking and buffer thresholds, is formulated as a Markov process and the stationary probability vector can be obtained using numerical methods for linear systems of equations. The desired performance characteristics, such as blocking probabilities, utilization, and throughputs can be obtained directly from the stationary probability distribution vector. The procedures for calculating of performance measures use the steady-state probabilities in the following manner:

1. Idle probability of a service station $p_{idle}$:

$$p_{idle} = p_{0,0} \tag{3}$$

2. Source1 blocking probability $p_{blS1}$:

$$p_{blS1} = p_{c+m,3} \tag{4}$$

3. Source2 blocking probability $p_{blS2}$:

$$p_{blS2} = \sum_{i=c+m1}^{c+m} p_{i,2} \tag{5}$$

4. The mean number of blocked jobs in the Source1 $n_{blS1}$:

$$n_{blS1} = 1 \cdot p_{c+m,3} \tag{6}$$

5. The mean number of blocked jobs in the Source2 $n_{blS2}$:

$$n_{blS2} = \sum_{i=c+m1}^{c+m} (1 \cdot p_{i,2}) \tag{7}$$

6. The mean blocking time in the Source1 $t_{blS1}$:

$$t_{blS1} = n_{blS1} \cdot \frac{1}{c \cdot \mu} \tag{8}$$

7. The mean blocking time in the Source2 $t_{blS2}$:

$$t_{blS2} = n_{blS2} \cdot \frac{1}{c \cdot \mu} \tag{9}$$

8. The effective arrival rate (intensity) from the Source1 :

$$\lambda_1^{eff} = \frac{1}{\frac{1}{\lambda_1} + t_{blS1}} \tag{10}$$

9. The effective arrival rate (intensity) from the Source2 :

$$\lambda_2^{eff} = \frac{1}{\frac{1}{\lambda_2} + t_{blS2}} \tag{11}$$

10. Servers utilization parameter $\rho$ :

$$\rho = \frac{l}{c} \tag{12}$$

Generally, presented above set of performance measures indicates that most of them depend on input traffic arrival rate and thresholds control parameters. These measures allow us to select the control parameter value to get the expected relative blocking ratios, if the arriving traffic pattern has been clear. This is a very important characteristic of adaptive buffer management scheme, for it has solved the thresholds setting problem for partial buffer sharing scheme.

## 4. Numerical results

To demonstrate our analysis procedures of a multi-server computer system with blocking, and flexible buffer management proposed in Section 2, we have performed numerous calculations. Using the above analysis, we can control the blocking of consecutive high priority jobs and low priority jobs through the combination of parameters like - $\lambda_1$, $\lambda_2$, *m1, m2,* $\mu$. The part of calculations were realized for many parameters combinations by varying both threshold values within a range from *1* to *17* for *m1*, plus within a range from *3* to *19* for *m2* (keeping relation *m2-m1* as constant). The inter-arrival rates from the source stations to the service station are chosen as $\lambda_1 = 1.6$ and $\lambda_2 = 1.4$. The service rate in service station is equal to $\mu = 0.3$. The buffers size is taken as $m = 20$ and $c = 4$. Based on such parameters, the following results were obtained and presented in Fig. 3. Figure 3 gives consecutive high and low priority jobs blocking probabilities and the effective arrival rates from Source1 and Source2 as a function of the thresholds policy.

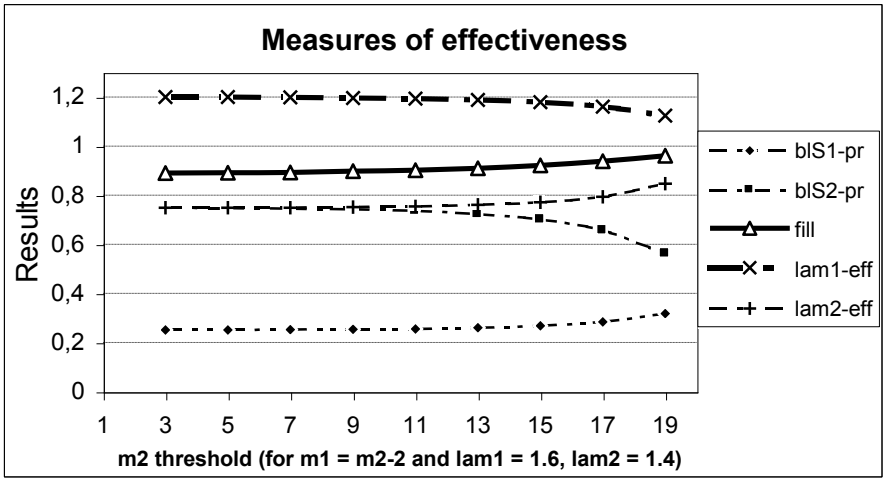



Fig. 3**.** Graphs of QoS parameters, where, *blS1-pr* is the Source1 blocking probability, *blS2-pr* is the Source2 blocking probability, *fill* is the buffer filling parameter, *lam1-eff* and *lam2-eff* are the effective arrival rates of high and low priority jobs, respectively.

The results of the experiment clearly show that the effect of the blocking, and a flexible threshold policy must be taken into account when analyzing performance of a computer system. As noted above, blocking factor and threshold policy considerably change the performance measures in such computer systems.

**Acknowledgements**

The Bialystok Technical University S/WI/5/08 and W/WI/5/09 grants support this work.

# References

[1] Awan I. *Analysis of multiple-threshold queues for congestion control of heterogeneous traffic streams*, Simulation Modelling Practice and Theory, Vol. 14, pp. 712-724, 2006.

[2] Balsamo S., de Nito Persone V., Onvural R. *Analysis of Queueing Networks with Blocking*, Kluwer Academic Publishers, Boston, 2001.

[3] Choi B.D., Choi S.H., Kim B., Sung D.K. *Analysis of priority queueing systems based on thresholds and its application to signalling system no. 7 with congestion control*, Computer Networks, Vol. 32, pp. 149-170, 2000.

[4] Oniszczuk W.: *An Intelligent Service Strategy in Linked Networks with Blocking and Feedback*, Studies in Computational Intelligence N. 134 "New Challenges in Applied Intelligence Technologies", N.T. Nguyen, R. Katarzyniak (Eds.), Springer-Verlag, Berlin, Heidelberg, pp. 351-361, 2008.

[5] Paganini F., Wang Z., Doyle J.C., and Low S.H. *Congestion Control for High Performance, Stability, and Fairness in General Networks*, IEEE/ACM Transactions on Networking, Vol. 13, No. 1, pp. 43-56, 2005.

[6] Sabrina F., Kanhere S.S., Jha S.K. *Design, Analysis, and Implementation of a Novel Multiple Resources Scheduler*, IEEE Transactions on Computers, Vol. 56, No. 8, pp. 1071-1086, 2007.

[7] Zhang H., Jiang Z-P., Fan Y., Panwar S. *Optimization based flow control with improved performance*, Communications in Information and Systems, Vol.4, No. 3, pp. 235-252, 2004.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 121–132

# Optimization problems in the theory of queues with dropping functions

ANDRZEJ CHYDZINSKI[a]

[a]Institute of Computer Sciences
Silesian University of Technology
andrzej.chydzinski@polsl.pl

**Abstract:** In this paper we first study the abilities to control the performance of a queueing system by exploiting the idea of the dropping function. In particular, we demonstrate how powerful the dropping function can be for controlling purposes. Then we formulate a number of optimization problems that are crucial for the design of queueing systems with dropping functions. We believe that these problems are of great importance and have to be solved on the queueing theory ground.

**Keywords:** single server queue, dropping function, active queue management

## 1. Introduction

In the classic single-server FIFO queue we cannot control the performance of the system. Given the input and service process parameterizations all we can do is compute the queue size, the waiting time, their average values, distributions etc. However, we cannot control these values at all[1].

Therefore, in many fields of application of queueing systems, a natural need arises – a need to control the performance of the system, namely to set the average queue size or the average waiting time etc. to a desired value.

There are three basic possibilities to control the performance of a single-server queue.

Firstly, we may try to manipulate the rate of the input stream. If we can somehow force the source of jobs to reduce or increase the intensity of the arrival stream then we can control the queue.

[1]In fact, even computing of these values may be very difficult, like in the $G/G/1$ queue.

Secondly, we may try to manipulate the service rate. For instance, we can design a system in which the average service time is reversely proportional to the current queue size. Naturally, it is possible to invent virtually infinite number of policies of this type.

Thirdly, we may try to block arriving jobs depending on the current system state or its history. This is somewhat similar, but not equivalent, to the manipulation of the input rate. The main difference is that blocking causes losses, i. e. jobs that are not served and will never return to the system – a phenomenon that is not present in systems with variable arrival rate. The characteristics of the loss process, like the overall loss ratio, the statistical structure of losses (tendency of losses to group together) etc. are very important for practical reasons.

In the literature we can find several examples of the queueing systems of the first and the second type, i. e. with variable arrival rate or variable service time distribution. For instance, the threshold-based systems are thoroughly studied in [1]-[5]. In such systems, the arrival or service process change when the queue size crosses (one or more) threshold level.

In this paper we deal with the third type of controlling the system performance. This type is probably the simplest one as it is often a simple matter to block a job (however, it comes at a cost of losses). In particular, we deal with systems that exploit the idea of the dropping function. Namely, an arriving job is blocked (dropped) with a probability that is a function of the queue size observed upon the job arrival. This function, mapping the queue sizes into the dropping probabilities, is called the dropping function.

In networking, the idea of the dropping function is used in the active queue management (AQM, see e. g. [6]–[12]) in Internet routers. For instance, it is used in three well-known algorithms: RED ([6], linear dropping function), GRED ([7], doubly-linear dropping function), REM ([9], exponential dropping function). It must be stressed, that usage of a particular shape of the dropping function has not been supported by rigorous analytical arguments so far.

In this paper we are aiming at two goals. Firstly, we want to demonstrate the powerful control capabilities that are connected with the usage of the dropping function. This is done in Section 2, where examples of dropping functions able to force certain values of the system throughput, the queue size, and combinations of these parameters are given. Then, in Section 3, we formulate several optimization problems that are crucial to the optimal design of the dropping-function based queue. Generally speaking, these optimization problems have the following form: we want to optimize the shape of the dropping function in such a way that one of the performance parameters achieves a target, given in advance, value, while other performance parameter achieves the best possible (minimal or maximal) value. Al-

ternatively, we want to optimize one performance parameter but for two distinct system loads. Finally, in Section 4 remarks concluding the paper are gathered.

## 2. Control capabilities of dropping functions

In order to demonstrate the powerful control capabilities connected with the usage of the dropping function we will use the single-server FIFO queue with Poisson arrivals (with rate $\lambda$), general type of service time distribution and final buffer (waiting room) of size $b-1$. Therefore, the total number of jobs in the system, including service position, cannot exceed $b$. A job that arrives when the buffer is full is dropped and lost.

This well-known model is further extended by the job dropping mechanism based on the dropping function. Namely, an arriving job can be dropped, even if the buffer is not full, with probability $d(n)$, where $n$ is the queue length (including service position) observed on this job arrival.

The function $d(n)$ is called the dropping function. It can assume any value in interval $[0,1]$ for $n=0,\ldots,b-1$. For $n \geq b$ we have $d(n)=1$, which is equivalent to the finite-buffer assumption.

Two very important characteristics of queueing systems with job losses are the loss ratio and the system throughput. The loss ratio, $L$, is defined as the long-run fraction of jobs that were dropped. The throughput, $T$, is defined as the long-run fraction of jobs that were allowed to the queue and it is equal to $1-L$.

The queueing system described above has been recently solved analytically in [13]. In particular, the formula for the queue size distribution in the system as well as the formula for the loss ratio have been shown there. Using these formulas we can easily manipulate the shape of the dropping function in order to control the queue size, the variance of the queue size and the system throughput.

For demonstrating purposes we will use herein a queue with arrival rate $\lambda=1$ and constant service time equal to 1. Therefore, the load offered to the queue, defined as

$$\rho = \lambda m,$$

with $m$ denoting the average service time, is also equal to 1. This value of $\rho$ was chosen for the sake of simplicity and the results are not specific to this value of $\rho$. All the examples presented below can be easily rearranged for other values of $\rho$ and for other distributions of the service time.

Now it is time to demonstrate the examples. It would be very easy to show a dropping function that gives a particular value of the average queue size, or a particular value of the variance of the queue size or a particular value of the throughput. Therefore, we start with more interesting examples - dropping functions that can

set the system throughput and the average queue size at the same time. For instance, the following dropping functions, $d_1$–$d_4$, were parameterized so that they all give the system throughput of 0.9, but the average queue size of 3, 3.5, 4 and 4.5, respectively. We have:

$$d_1(n) = \begin{cases} 0 & \text{if} \quad n \leq 1, \\ -0.0548821 + 0.0490536n & \text{if} \quad 1 < n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

$$d_2(n) = \begin{cases} 0 & \text{if} \quad n \leq 2, \\ 0.256407 & \text{if} \quad 2 \leq n < 3, \\ 0 & \text{if} \quad 3 \leq n < 5, \\ 0.2 & \text{if} \quad 5 \leq n < 6, \\ 0 & \text{if} \quad 6 \leq n < 7, \\ 0.194625 & \text{if} \quad 7 \leq n < 8, \\ 0.2 & \text{if} \quad 8 \leq n < 9, \\ 0.1 & \text{if} \quad 9 \leq n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

$$d_3(n) = \begin{cases} 0.13921803 - 0.037655n + 0.0037655n^2 & \text{if} \quad n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

$$d_4(n) = \begin{cases} 0.2275822 - 0.087680n + 0.0087680n^2 & \text{if} \quad n < 5, \\ 0 & \text{if} \quad 5 \leq n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

These dropping functions are depicted in Fig. 1. Detailed performance characteristics for $d_1$–$d_4$ are given in Tab. 1, while their steady-state queue size distributions are depicted in Fig. 2.

The functions $d_1$–$d_4$ exemplify well, how powerful tool the dropping function can be. Using them we were able to control two performance characteristics at the same time. Now, can we control more than two characteristics? To answer this question, let us consider the following dropping function:

$$d_5(n) = \begin{cases} 0 & \text{if} \quad n < 3, \\ -1.33946 + 0.66885n - 0.066885n^2 & \text{if} \quad 3 \leq n \leq 7, \\ 0 & \text{if} \quad 7 < n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

| | throughput | average queue size | variance of the queue size |
|---|---|---|---|
| $d_1$ | 0.90000 | 3.0000 | 4.3467 |
| $d_2$ | 0.90000 | 3.5000 | 6.6136 |
| $d_3$ | 0.90000 | 4.0000 | 8.0085 |
| $d_4$ | 0.90000 | 4.5000 | 9.3859 |

Table 1. Queueing performance characteristics for dropping functions $d_1$-$d_4$.

| | throughput | average queue size | variance of the queue size |
|---|---|---|---|
| $d_5$ | 0.90000 | 3.0000 | 4.8860 |

Table 2. Queueing performance characteristics for dropping function $d_5$.

depicted in Fig. 3. Its performance characteristics are given in Table 3.

As we can see, the dropping functions $d_5$ and $d_1$ have common throughput and common average queue size but different variance of the queue size. This leads to the supposition, that there are many shapes of the dropping function that provide the same throughput and the same queue size but different variance. Therefore we probably can control also the latter parameter.

To conclude this section, we present another control possibility obtained by application of the dropping function. Let us assume that the rate of the arrival process may vary. For instance, the arrival rate may assume two values: $\lambda_1$ and $\lambda_2$. In this case the system load varies as well and can reach two values, $\rho_1$ and $\rho_2$, respectively. Now, assume that we want the average queue size to be $Q_1$ when the arrival rate is $\lambda_1$ and $Q_2$ when the arrival rate is $\lambda_2$. Apparently, this is possible if a proper dropping function is used.

As a first example, consider the following dropping function:

$$d_6(n) = \begin{cases} 0 & \text{if} \quad n < 2, \\ 0.49724 & \text{if} \quad 2 \leq n < 3, \\ 0 & \text{if} \quad 3 \leq n < 5, \\ 0.55520 & \text{if} \quad 5 \leq n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

depicted in Fig. 4. This function was carefully parameterized in such a way that it gives the average queue size equal to 2 when $\lambda = 1$ and equal to 3 when $\lambda = 1.2$ (see also Tab. 3 for the variances of the queue size).

| | average queue size | variance of the queue size |
|---|---|---|
| $\lambda = 1$ | 2.0000 | 2.5692 |
| $\lambda = 1.2$ | 3.0000 | 3.4568 |

Table 3: Queueing performance characteristics for the dropping function $d_6$ and two distinct arrival rates, $\lambda_1 = 1$ and $\lambda_2 = 1.2$.

On the other hand, consider the dropping function:

$$d_7(n) = \begin{cases} 0 & \text{if} \quad n < 2, \\ 0.805035 & \text{if} \quad 2 \leq n < 3, \\ 0 & \text{if} \quad 3 \leq n < 5, \\ 0.077840 & \text{if} \quad 5 \leq n < 10, \\ 1 & \text{if} \quad n \geq 10, \end{cases}$$

also depicted in Fig. 4. The function $d_7$ was parameterized in such a way that it gives the average queue size equal to 2 when $\lambda = 1$ and equal to 5 when $\lambda = 1.2$ (see also Tab. 4).

| | average queue size | variance of the queue size |
|---|---|---|
| $\rho = 1$ | 2.0000 | 4.8866 |
| $\rho = 1.2$ | 5.0000 | 11.408 |

Table 4: Queueing performance characteristics for the dropping function $d_7$ and two distinct arrival rates, $\lambda_1 = 1$ and $\lambda_2 = 1.2$.

## 3. Optimization problems

Before we proceed to the optimization problems, we have to give a though to the important problem, that was silently present in the previous section: in which intervals we can obtain particular performance characteristics? In other words, what are the domains of control?

When we try to control only one parameter, the answer is usually simple. For instance, if we have $\rho \geq 1$ then the target queue size, $Q$, can assume any value, i. e.:

$$Q \in [0, \infty).$$

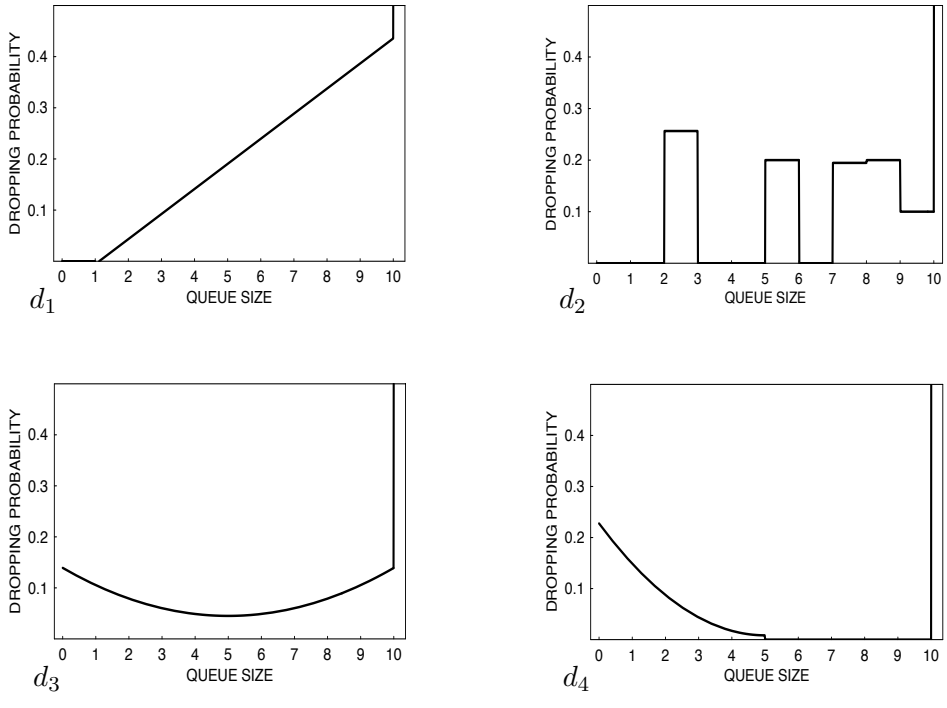



Fig. 1. Dropping functions $d_1$-$d_4$.

This fact is easy to explain. Similarly, if $\rho < 1$ then the target queue size may be:

$$Q \in [0, Q_{\max}),$$

where $Q_{\max}$ is the average queue size in the infinite-buffer system without the dropping function.

In a similar way we can obtain control intervals for the system throughput.

Now, suppose that we are interested in controlling of more than one characteristics. This leads, for instance, to the following questions about control domains. Given the system parametrization and the target queue size $Q$ in which interval the throughput $T$ can be obtained? Or, given the system parametrization, the target queue size and the target throughput, in which interval the variance of the queue size $V$ can be obtained? Or, given the target queue size for arrival rate $\lambda_1$, in which interval the queue size can be obtained for another arrival rate, $\lambda_2$?

The answers to these questions are not simple. In fact, they are closely connected to the optimization problems discussed below. For instance, given the tar-

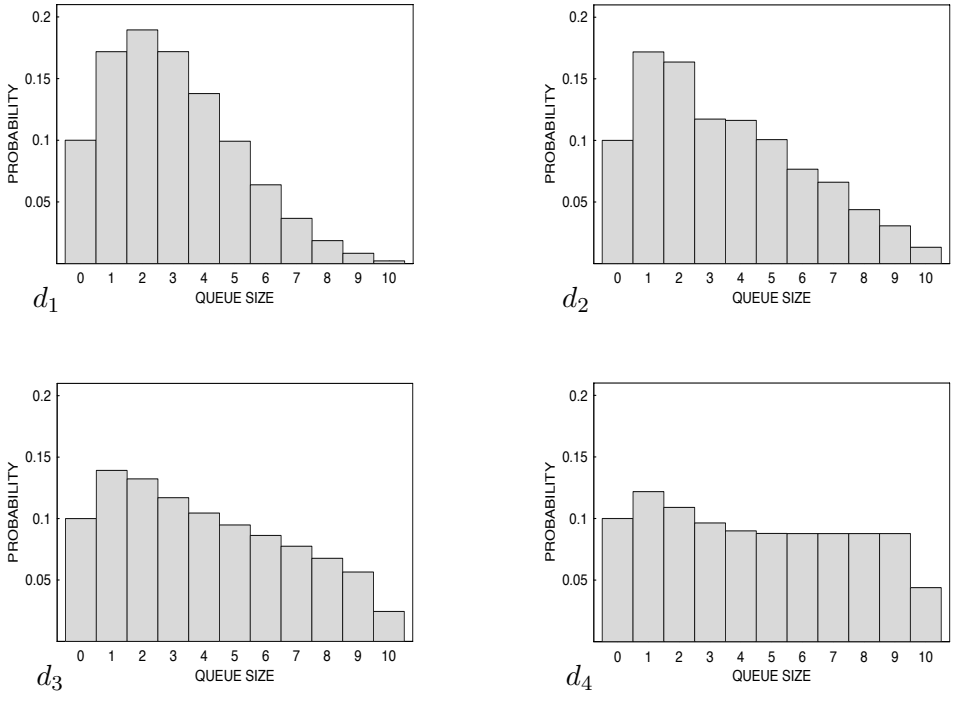



Fig. 2. Queue size distributions for dropping functions $d_1$-$d_4$.

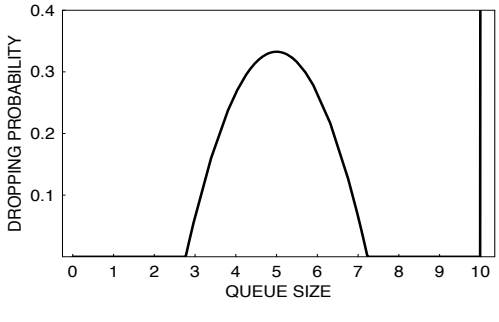



Fig. 3. Dropping function $d_5$.

get queue size, $Q$, the achievable throughput belongs to the interval $[T_{\min}, T_{\max}]$, where $T_{\max}$ is the solution of one of the optimization problems.

Finally, the last question regarding control possibilities is: how many performance characteristics can be controlled in some, non-reduced to point, intervals.

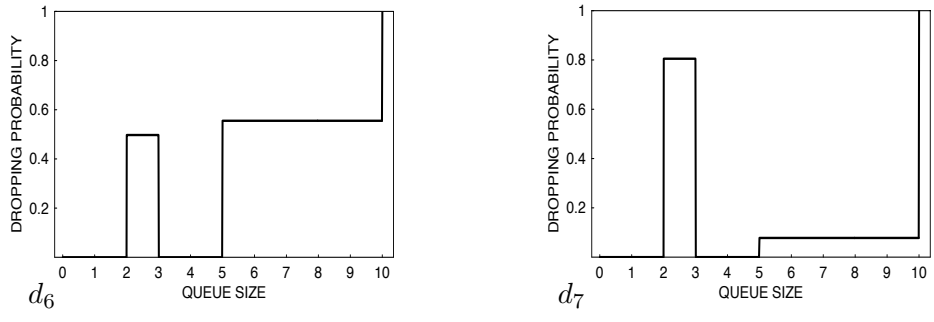



Fig. 4. Dropping functions $d_6$ and $d_7$.

Above we showed that it is possible to control three characteristics: the queue size, the throughput and the variance of the queue size. Can it be four? For instance, can the tail probability, $\mathbf{P}(Q > q_0)$, for some $q_0$, be the fourth controlled characteristic? This is yet another open question.

Now we may proceed to the optimization problems. They are motivated by a natural need to optimize less important performance characteristics, given that the most important characteristic assume the target value. For instance, we may need the queue size to be short and stable. To accomplish this, we can set the target average queue size to 5 and search for the dropping function that provides this average queue size and the minimal possible variance at the same time. Or, we may want to have large throughput and short queue size at the same time. In this case we can assume the target throughput of 0.95 and search for the dropping function that provides the minimal queue size. Or, vice versa, we may need the queue to be short and the throughput to be as high as possible. Or we may want to assure a good trade-off between the queue size and the throughput. Naturally, there are many problems of this type. Herein we formulate a few among most interesting of them.

1. Given the system parametrization and the target queue size, $Q$, which dropping function provides $Q$ and the maximal possible throughput?
2. Given the system parametrization and the target throughput, $T$, which dropping function provides $T$ and the minimal possible queue size?
3. Given the system parametrization and the target queue size, $Q$, which dropping function provides $Q$ and the minimal possible variance of the queue size?

4. (The trade-off problem). Given the system parametrization, which dropping function gives the maximal value of the product:

$$I(T)D(Q),$$

where $I(T)$ is an increasing function of the throughput, while $D(Q)$ is a decreasing function of the queue size. The simplest form of this problem is maximization of the function $T/Q$.

5. Given the system parametrization, the target queue size and the target throughput, which dropping function provides the minimal possible variance? Or, similarly, given the target throughput and the target variance, which dropping function provides the minimal queue size?

6. Given the target queue size $Q_1$ for arrival rate $\lambda_1$, which dropping function provides the minimal possible queue size for another arrival rate, $\lambda_2$?

7. Given the target throughput, $T_1$ for arrival rate $\lambda_1$, which dropping function provides the maximal possible throughput for another arrival rate, $\lambda_2$?

Solving analytically these problems is an important, but not easy task. It is not even clear in which classes of functions the solutions should be looked for. Intuitively, the optimal dropping functions should be non-decreasing, but this intuition should be rigorously proven first. Another candidate to solve some of the problems presented above might be a simple, drop-tail dropping function ($d(n) = 0$ for $n < b$, $d(n) = 1$ for $n \geq b$). However, such dropping function usually does not allow to achieve the target characteristic. For instance, it is unlikely that we can obtain the average queue size of 3 using the drop-tail function.

## 4. Conclusions

In this paper the powerful control abilities gained by usage of the dropping function were shown. Firstly, the dropping functions that give not only the target throughput but also the target queue size were presented. Secondly, it was shown that setting the two characteristics does not determine the shape of the dropping function and still leaves some room to control yet another characteristic, like the variance of the queue size. Thirdly, it was demonstrated that the dropping function can be used to control one characteristic in the system with variable load (i. e. variable arrival rate). Finally, several optimization problems to be solved analytically were described. These problems are crucial for the optimal design of queueing systems with dropping functions.

## 5. Acknowledgement

This work was supported by MNiSW under grant N N516 381134.

## References

[1] Chydzinski, A. The M/G-G/1 oscillating queueing system. Queueing Systems, vol. 42:3, pp. 255–268, (2002).

[2] Chydzinski, A. The M-M/G/1-type oscillating systems. Cybernetics and Systems Analysis, vol. 39(2), pp. 316–324, (2003).

[3] Chydzinski, A. The oscillating queue with finite buffer. Performance Evaluation, vol. 57(3), pp. 341–355, (2004).

[4] Pacheco, A. and Ribeiro, H. Consecutive customer losses in oscillating GIX/M//n systems with state dependent services rates. Annals of Operations Research, Volume 162, Number 1, pp. 143–158, (2008).

[5] Pacheco, A. and Ribeiro, H. Consecutive customer losses in regular and oscillating MX/G/1/n systems. Queueing Systems, Volume 58, Issue 2, pp. 121–136, (2008).

[6] Floyd, S.; Jacobson, V. Random early detection gateways for congestion avoidance. IEEE/ACM Transactions on Networking, Volume 1, Issue 4, Page(s): 397 – 413, (1993).

[7] Rosolen, V.; Bonaventure, O., and Leduc, G. A RED discard strategy for ATM networks and its performance evaluation with TCP/IP traffic. SIGCOMM Comput. Commun. Rev. 29, 3, Jul. (1999).

[8] Floyd, S.; Gummadi, R. and Shenker, S. Adaptive RED: An algorithm for increasing the robustness of RED. Tech. Rep.: ACIRI, (2001).

[9] Athuraliya, S.; Low, S. H.; Li, V. H.; Qinghe Yin. REM: active queue management, IEEE Network. On page(s): 48-53, Volume: 15, Issue: 3, May (2001).

[10] Feng, W.; Shin, K. G.; Kandlur, D. D. and Saha, D. The BLUE active queue management algorithms. IEEE/ACM Transactions on Networking, pp. 513 – 528, Volume: 10, Issue: 4, Aug. (2002).

[11] Wydrowski, B. and Zukerman, M. GREEN: an active queue management algorithm for a self managed Internet, in: Proceeding of IEEE International Conference on Communications ICC'2002, vol. 4, pp. 2368-2372, April, (2002).

[12] Kunniyur, S. S. and Srikant, R. An adaptive virtual queue (AVQ) algorithm for active queue management. IEEE/ACM Transactions on Networking. Page(s): 286–299, Volume: 12, Issue: 2, April (2004).

[13] Chydzinski, A. and Chrost, L. Analysis of AQM queues with queue-size based packet dropping. Submitted for publication, (2009).

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 133–142

# Active Queue Management with non-linear packets dropping function

DARIUSZ RAFAŁ AUGUSTYN [a] ADAM DOMAŃSKI [a] JOANNA DOMAŃSKA [b]

[a]Institute of Informatics
Silesian Technical University
Akademicka 16, 44–100 Gliwice, Poland
*{draugustyn, adamd}@polsl.pl*

[b]Polish Academy of Sciences
Baltycka 5, 44–100 Gliwice, Poland
*joanna@iitis.pl*

**Abstract:** Algorithms of queue management in IP routers determine which packet should be deleted when necessary. The article investigates the influence of packet rejection probability function on the performance, i.e. response time for in case of RED and nRED queues. In particular, the self-similar traffic is considered. The quantitative analysis based on simulations is shown.
**Keywords:** RED, active queue management, dropping packets.

## 1. Introduction

Algorithms of queue management at IP routers determine which packet should be deleted when necessary. The Active Queue Management, recommended now by IETF, enhances the efficiency of transfers and cooperates with TCP congestion window mechanism in adapting the flows intensity to the congestion at a network [16].

This paper describes another approach to packed dropping function used in Active Queue Management. Here, we reconsider the problem of non linear packet loss probability function in presence of self-similar traffic.

Sections 2. gives basic notions on active queue management, Section 3. presents briefly a self-similar model used in the article. Section 4. gives simulation models of the considered two active queue management schemes: RED and non linear RED. Section 5. discusses numerical results, some conclusions are given in Section 6..

## 2. Active Queue Management

In *passive* queue management, packets coming to a buffer are rejected only if there is no space in the buffer to store them, hence the senders have no earlier warning on the danger of growing congestion. In this case all packets coming during saturation of the buffer are lost. The existing schemes may differ on the choice of packet to be deleted (end of the tail, head of the tail, random). During a saturation period all connections are affected and all react in the same way, hence they become synchronized. To enhance the throughput and fairness of the link sharing, also to eliminate the synchronization, the Internet Engineering Task Force (IETF) recommends *active* algorithms of buffer management. They incorporate mechanisms of preventive packet dropping when there is still place to store some packets, to advertise that the queue is growing and the danger of congestion is ahead. The probability of packet rejection is growing together with the level of congestion. The packets are dropped randomly, hence only chosen users are notified and the global synchronization of connections is avoided. A detailed discussion of the active queue management goals may be found in [16].

The RED (Random Early Detection) algorithm was proposed by IETF to enhance the transmission via IP routers. It was primarily described by Sally Floyd and Van Jacobson in [23]. Its idea is based on a drop function giving probability that a packet is rejected. The argument *avg* of this function is a weighted moving average queue length, acting as a low-pass filter and calculated at the arrival of each packet as

$$avg = (1 - w)avg' + wq$$

where $avg'$ is the previous value of $avg$, $q$ is the current queue length and $w$ is a weight determining the importance of the instantaneous queue length, typically $w \ll 1$. If $w$ is too small, the reaction on arising congestion is too slow, if $w$ is too large, the algorithm is too sensitive on ephemeral changes of the queue (noise). Articles [23, 8] recommend $w = 0.001$ or $w = 0.002$, and [9] shows the efficiency of $w = 0.05$ and $w = 0.07$. Article [10] analyses the influence of $w$ on queuing time fluctuations, obviously the larger $w$, the higher fluctuations. In RED drop function there are two thresholds $Min_{th}$ and $Max_{th}$. If $avg < Min_{th}$ all packets are admitted, if $Min_{th} < avg < Max_{th}$ then dropping probability $p$ is growing linearly from 0 to $p_{max}$ :

$$p = p_{max} \frac{avg - Min_{th}}{Max_{th} - Min_{th}}$$

and if $avg > Max_{th}$ then all packets are dropped. The value of $p_{max}$ has also a strong influence on the RED performance: if it is too large, the overall throughput

is unnecessarily choked and if it's too small the danger of synchronization arises; [26] recommends $p_{max} = 0.1$. The problem of the choice of parameters is still discussed, see e.g. [11, 24]. The mean $avg$ may be also determined in other way, see [25] for discussion. Despite of evident highlights, RED has also such drawbacks as low throughput, unfair bandwidth sharing, introduction of variable latency, deterioration of network stability. Therefore numerous propositions of basic algorithms improvements appear, their comparison may be found e.g. in [12].
This paper describes also another approach to packed dropping function used in Active Queue Management. The linear function of probability of packed dropping is known since years. But there is no hard premises to this assumption of linearity. There are also many well-known nonlinear approaches like ARED [7], NLRED[27]. This paper used the methods known in calculus of variations. Here an unknown function $f(x)$ with domain $[0, l]$ is approximated by as a finite linear combination of basing functions:

$$f(x) = \sum_{i=1}^{N} a_j \Phi_j(x),$$

where $a_j$ are undetermined parameters and $\Phi_j$ can be a series of orthogonal polynomials

$$\Phi_j = x^{j-1}(l - x)$$

Optimal values of $a_j$ can be obtained by finding minimum of some functional $J$ implicitly defined on $f$. Only a few $\Phi_j$ $(e.g. N = 2)$ are required to achieve the acceptable accuracy of approximation of optimal $f$.
Basing on these equations we propose to define $p$ - the function of probability of packed dropping with domain $[Min_{th}, Max_{th}]$ as follows:

$$p(x, a_1, a_2) = \begin{cases} 0 & \text{for } x < Min_{th} \\ \varphi_0(x) + a_1\varphi_1(x) + a_2\varphi_2(x) & \text{for } Min_{th} <= x <= Max_{th} \\ 1 & \text{for } x > Max_{th} \end{cases}$$

where basis functions are defined:
$\varphi_0(x) = p_{max} \frac{x - Min_{th}}{Max_{th} - Min_{th}}$
$\varphi_1(x) = (x - Min_{th})(Max_{th} - x)$
$\varphi_2(x) = (x - Min_{th})^2(Max_{th} - x)$
Sample of a $p$ function was shown on figure 1.

The functional J can based on one of two different parameters: average length queue or the average waiting time. Obtaining the optimal function $p$ is equivalent to finding of minimum of $J$ which is implicitly defined on parameters $a1$ and $a2$.

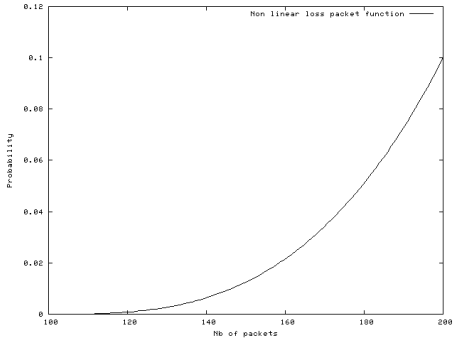



Fig. 1: Sample function of probability of packed dropping for some $a_1$, $a_2$ and given values $Min_{th} = 100$, $Max_{th} = 200$, $p_{max=0.1}$

## 3. Self-similarity of network traffic

Measurements and statistical analysis of network traffic, e.g. [5, 19] show that it displays a self-similar character. It is observed on various protocol layers and in different network structures. Self-similarity of a process means that the change of time scales does not affect the statistical characteristics of the process. It results in long-range dependence and makes possible the occurrence of very long periods of high (or low) traffic intensity. These features have a great impact on a network performance. They enlarge the mean queue lengths at buffers and increase the probability of packet losses, reducing this way the quality of services provided by a network. Also TCP/IP traffic is characterized by burstiness and long-term correlation, [20], its features are additionally influenced by the performance of congestion avoidance and congestion management mechanisms, [21, 22].

To represent the self-similar traffic we use here a model introduced by S. Robert [18, 17]. The time of the model is discrete and divided into unit length slots. Only one packet can arrive during each time-slot. In the case of memoryless, geometrical source, the packet comes into system with fixed probability $\alpha_1$. In the case of self-similar traffic, packet arrivals are determined by a $n$-state discrete time Markov chain called modulator. It was assumed that modulator has $n = 5$ states ($i = 0, 1, \ldots 4$) and packets arrive only when the modulator is in state $i = 0$. The elements of the modulator transition probability matrix depend only on two parameters: $q$ and $a$ – therefore only two parameters should be fitted to match the mean value and Hurst parameter of the process. If $p_{ij}$ denotes the modulator transition probability from state $i$ to state $j$, then it was assumed that $p_{0j} = 1/a^j$, $p_{j0} = (q/a)^j$, $p_{jj} = 1 - (q/a)^j$ where $j = 1, \ldots, 4$, $p_{00} = 1 - 1/a - \ldots - 1/a^4$, and remaining probabilities are equal to zero. The passages from the state 0 to one

of other states determine the process behavior on one time scale, hence the number of these states corresponds to the number of time-scales where the process may by considered as self-similar.

## 4. Simulation models

The simulation evaluations were carried out with the use of OMNeT++ simulation framework of discrete events. The OMNeT++ is the modular, component-based simulator, with an Eclipse-based IDE and a graphical environment, mainly designed for simulation of communication networks, queuing networks and performance evaluation. The framework is very popular in research and for academic purposes [13], [14]. To emphasize the importance of using self-similar sources of traffic the comparative research has been carried out for the self-similar and poisson source. Input traffic intensity was chosen as $\alpha = 0.5$ or $\alpha = 0.081$, and due to the modulator characteristics, the Hurst parameter of self-similar traffic was fixed to $H = 0.8$. For both considered in comparisons cases, i.e. for geometric interarrival time distribution (which corresponds to Poisson traffic in case of continuous time models) and self-similar traffic, the considered traffic intensities are the same. A detailed discussion of the choice of model parameters is also presented in [15].

## 5. Numerical results

In this section we present more interesting results achieved in the simulation. Input traffic intensity (for geometric and self-similar traffic) was chosen as $\alpha = 0.5$, and due to the modulator characteristics, the Hurst parameter of self-similar traffic was fixed to $H = 0.78$.

The RED parameters had the following values: buffer size 250 packets, threshold values $Min_{th} = 100$ and $Max_{th} = 200$, $p_{max} = 0.1$, $w = 0.002$. Parameter $\mu$ of geometric distribution of service times (probability of the end of service within a current time-slot) was $\mu = 0.25$ or $\mu = 0.5$. Due to the changes of $\mu$, two different traffic loads (low and high) were considered. For the nRED $p_{max}$ changes from 0.1 to 0.9.

The first experiments concerned the case of Poisson traffic. For these sources the impact of non-linearly of probability dropping function depended on queue load. The figure 2 shows situation of the overloaded queues ($\alpha = 0.5$, $\mu = 0.25$). For nRED were chosen on an experimental basis, the following parameters: $p_{max}$=0.6, $a_1$=-0,00006, $a_2$: -0,0000006. For the nRED queue, we obtained the same probability of loss and shorter average waiting times (3.2%) and queue length (3.5%).

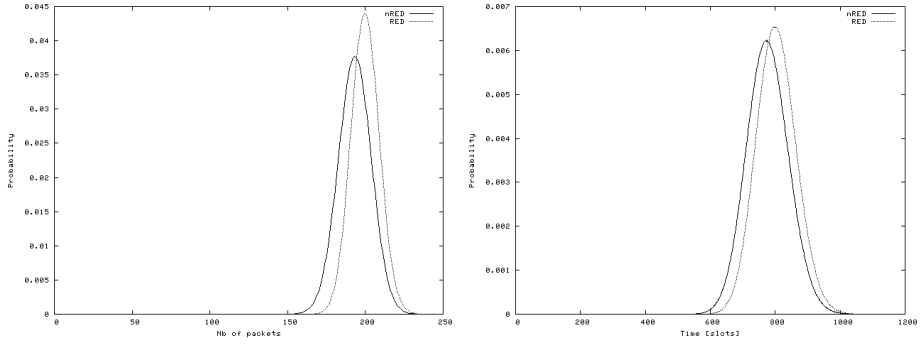



Fig. 2: Distribution of the avg queue length (left), waiting times (right) for geometrical source $\alpha = 0.5$, $\mu = 0.25$

The figure 3 shows unloaded queues ($\alpha = 0.5$, $\mu = 0.25$). For the same simulation parameters the probability of loss reduced by 1.3 percent. Unfortunately, increased average waiting times (7%) and queue length (7%).

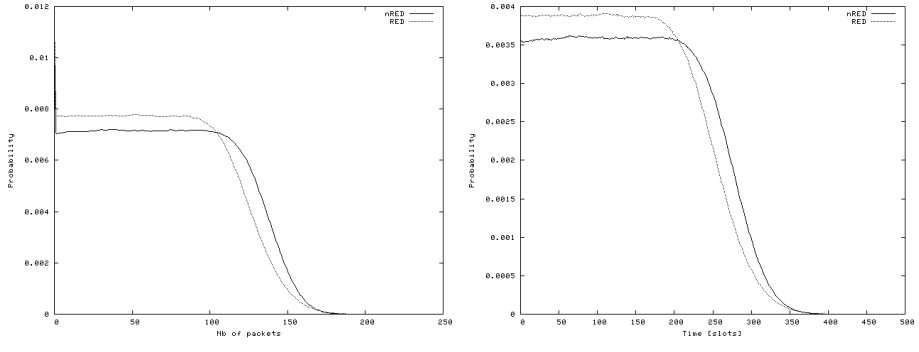



Fig. 3: Distribution of the avg queue length (left), waiting times (right) for geometrical source $\alpha = 0.5$, $\mu = 0.5$

These results are caused the function of packet loss. For short queue length it adopted too small values.

The figure 4 shows the results for different nRED parameters:
nRED1 - $p_{max}$=0.6, $a_1$=-0,00006, $a_2$: -0,0000006,
nRED2 - $p_{max}$=0.1, $a_1$=-0,00001, $a_2$: -0,0000001,
nRED3 - $p_{max}$=0.8, $a_1$=-0,00008, $a_2$: -0,0000008.

For the nRED2 queue, probability of loss reduced by 20.5% and increased average waiting times (19.5%) and queue length (19.9%) For the nRED3 queue,

probability of loss reduced by 7% and increased average waiting times (5%) and queue length (9%).

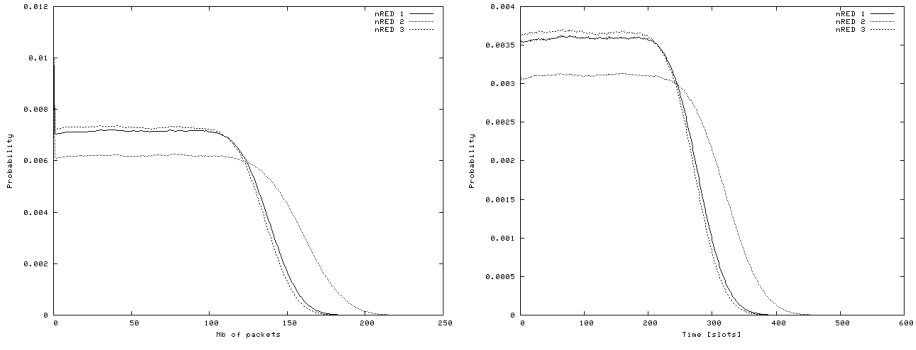



Fig. 4: Distribution of the avg queue length (left), waiting times (right) for geometrical source $\alpha = 0.5$, $\mu = 0.25$

Figure 4 shows an adequate situation (overloaded queue) but for self-similar traffic. For this case we received the same loss probability (but nRED reduced the losses associated with the queue overflow (2.7%)) and shorter average waiting times (2.4%) and queue length (2.9%).

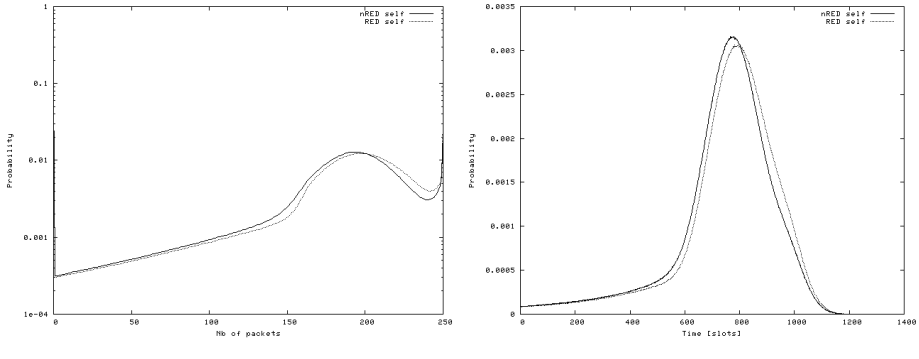



Fig. 5: Distribution of the avg queue length (left), waiting times (right) for self-similar source $\alpha = 0.5$, $\mu = 0.25$

For unloaded queue (figure 6 the probability of loss increases by 1.9%, but with less average waiting time of 7.8% and less the average queue occupancy of 8.8%.

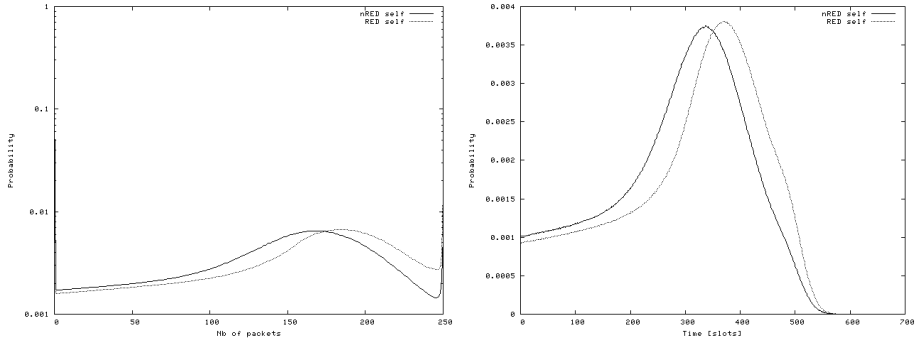



Fig. 6: Distribution of the avg queue length (left), waiting times (right) for self-similar source $\alpha = 0.5$, $\mu = 0.5$

## 6. Conclusions

In this article we present the problem of packet loss probability function and its influence the behavior of the AQM queue. During the tests we analyzed the following parameters of the transmission with AQM: the length of the queue, the number of rejected packets and waiting times in queues. In case of light load, the difference is more visible for self-similar traffic. In case of heavy load, the difference is also substantial for short-dependent traffic. Future works may concentrate on using different kinds of basis functions (not only orthogonal polynomials) used in definition of non-linear packets dropping function.

## References

[1] J. Domańska, A. Domański, Czachórski T., "Implementation of modified AQM mechanisms in IP routers", Journal of Communications Software and Systems, Volume: 4 Number: 1, March 2008

[2] J. Domańska, A. Domański, "Active Queue Management in Linux based routers" IWSI 2008

[3] J. Domańska, A. Domański, T. Czachórski, "The Drop-From-Front Strategy in AQM", Lecture Notes in Computer Science, Vol. 4712/2007, pp. 61-72, Springer Berlin/Heidelberg, 2007.

[4] S. Athuraliya, V. H. Li, S. H. Low and Q. Yin, REM: Active Queue Management http://netlab.caltech.edu/FAST/papers/cbef.pdf

[5] Srisankar S. Kunniyur, Member, IEEE, and R. Srikant, Senior Member, IEEE An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management http://comm.csl.uiuc.edu/srikant/Papers/avq.pdf

[6] Sally Floyd, Ramakrishna Gummadi, and Scott Shenker Adaptive RED: An Algorithm for Increasing the Robustness of REDs Active Queue Management http://citeseer.ist.psu.edu/448749.html

[7] Sally Floyd, Ramakrishna Gummadi, and Scott Shenker Adaptive RED: An Algorithm for Increasing the Robustness of REDs Active Queue Management http://citeseer.ist.psu.edu/448749.html

[8] S. Floyd, Discussions of setting parameters, http://www.icir.org/floyd/ REDparameters.txt, 1997.

[9] B. Zheng and M. Atiquzzaman, A framework to determine the optimal weight parameter of red in next generation internet routers, The University of Dayton, Department of Electrical and Computer Engineering, Tech. Rep., 2000.

[10] M. May, T. Bonald, and J. Bolot, Analytic evaluation of red performance, IEEE Infocom 2000, Tel-Aviv, Izrael, 2000.

[11] W. Chang Feng, D. Kandlur, and D. Saha, Adaptive packet marking for maintaining end to end throughput in a differentiated service internet, IEEE/ACM Transactions on Networking, vol. 7, no. 5, pp. 685-697, 1999.

[12] M. Hassan and R. Jain, High Performance TCP/IP Networking. Pearson Education Inc., 2004.

[13] OMNET++ homepage, http://www.omnetpp.org/.

[14] Domanska J., Grochla K., Nowak S., Symulator zdarzeń dyskretnych OMNeT++, Wyd. Wyzsza Szkola Biznesu w Dabrowie Górniczej, Dabrowa Górnicza 2009.

[15] J. Domańska "Procesy Markowa w modelowaniu natężenia ruchu w sieciach komputerowych.", PhD thesis, IITiS PAN, Gliwice, 2005.

[16] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., Zhang, L., "Recommendations on queue management and congestion avoidance in the internet", RFC 2309, IETF (1998)

[17] Robert, S., Boudec, J.Y.L., "New models for pseudo self-similar traffic", Performance Evaluation **30**(1-2) (1997) 57–68

[18] Robert, S.,"Modélisation Markovienne du Trafic dans les Réseaux de Communication.", PhD thesis, Ecole Polytechnique Fédérale de Lausanne (1996) Nr 1479.

[19] Willinger, W., Leland, W.E., Taqqu, M.S., “On the self-similar nature of ethernet traffic”, IEEE/ACM Transactions on Networking (February 1994)

[20] Abry, P., Flandrin, P., Taqqu, M., Veitch, D., “Wavelets for the analysis, estimation and synthesis of scaling data”, In: Self-similar Network Traffic Analysis and Performance Evaluation. K. Park i W. Willinger (eds) (1999)

[21] Paxson, V., Floyd, S.: “Wide area traffic: the failure of poisson modeling”, IEEE/ACM Transactions on Networking **3** (1995)

[22] Feldman, A., Gilbert, A., Huang, P., Willinger, W., “Dynamics of ip traffic: Study of the role of variability and the impact of control”, ACM SIGCOMM’99, Cambridge (1999)

[23] Floyd, S., Jacobson, V., “Random early detection gateways for congestion avoidance”, IEEE/ACM Transactions on Networking **1**(4) (1993) 397–413

[24] May, M., Diot, C., Lyles, B., Bolot, J., “Influence of active queue management parameters on aggregate traffic performance”, Technical report, Research Report, Institut de Recherche en Informatique et en Automatique (2000)

[25] Zheng, B., Atiquzzaman, M., “Low pass filter/over drop avoidance (lpf/oda): An algorithm to improve the response time of red gateways”, Int. Journal of Communication Systems **15**(10) (2002) 899–906

[26] Floyd S., “Discussions of setting parameters”, http:// www.icir.org/ floyd/ REDparameters.txt (1997)

[27] Zhou K., Yeung K. L., Li V., “Nonlinear RED: Asimple yet efficient active queue management scheme”, Computer Networks 50, 3784-3794, Elsevier 2006

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 143 – 152

# QoS management for multimedia traffic in synchronous slotted-ring OPS networks

MATEUSZ NOWAK PIOTR PECKA

Institute of Theoretical and Applied Informatics
Polish Academy of Science
ul. Bałtycka 5, Gliwice, Poland
*{m.nowak, piotr}@iitis.gliwice.pl*

**Abstract:** The paper presents synchronous slotted-ring network with optical packet switching. The network is equipped with quality of service management, particularly considering multimedia transfers. Thanks to proposed mechanism of guarantees, multimedia data are delivered with guaranteed delay time, not greater than assumed, and in the case the delivery on time is impossible – they are removed, freeing place for remaining packets.

**Keywords:** OPS networks, network simulation.

## 1. Introduction

Contemporary computer network of medium and big range usually use optical technologies for transferring digital data. These networks, based on technologies like SONET/SDH or optical Ethernet, use optic fibre for signal transmission, however in nodes all data are converted into electronic form. In commonly used network technologies, management of digital data traffic is possible only when they exist in the form of electric signals. There is possible then to read, interpret, process the data with help of network traffic routing algorithms, and next send them into next network segment, likely after another conversion into a form of modulated optical wave.

All over the world, work on fully optical networks in conducted. Switching (routing to proper network segment) in a node of such a network takes place without conversion of information on electric form, therefore such a networks are described as fully optical or – more precisely – as network with optical packet switching (OPS networks).

Optical switching is much faster than electronic one. OPS technology already turned from theoretical stage to construction of prototype devices. In Europe a number of companies and research organizations is engaged in both construction of OPS devices and in working out management methods, i.a. gathered in research projects like ECOFRAME or CARRIOCAS.

OPS networks differ from traditional electronic and electronic-optical ones. Main distinction is complete lack of data buffering in optical switches. With this feature necessity of working out new solutions in the field of traffic management in OPS network is bound, both in fully optical nodes and in border nodes, being an electronic-optical interface for clients of OPS network.

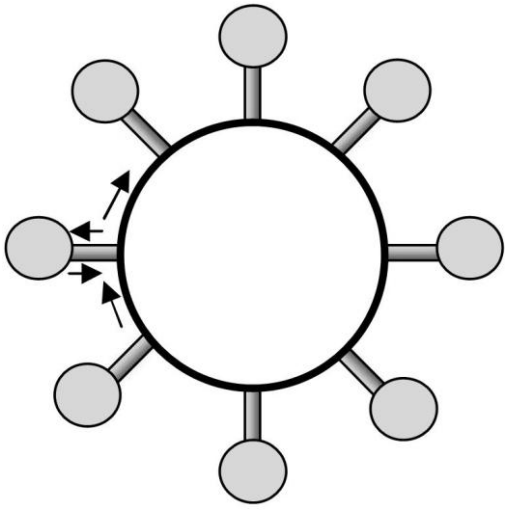


Fig. 1. Ring-type network.

In the paper research are presented, which concern synchronous slotted-ring networks. Ring-type architecture is easy to configure and manage, as opposed to complex mesh-type networks. It's architecture, which is often applied in metropolitan networks (MANs). It originates from Cambridge-Ring architecture[1] whose successors are the widely used networks like Token-Ring [2], FDDI [3], ATMR [4,5] or RPR [6]. Currently accommodating the ring type architecture to the OPS network is under research (eg. [7]). The proposals presented in the paper complement this research.

## 2. Network Architecture

The paper presents synchronous slotted ring-type network witch optical packet switching. Transportation unit in the network is a frame of constant capacity. Frames are passed in synchronous way, and in each network segment there is one frame, full or empty.

Network nodes are equipped with the Add/Drop mechanism[8], which makes it possible to remove a packet from a frame, if it is assigned to this node or if the quality management mechanism has requested this, add a packet to the empty frame if the node has the information to send, or pass a frame in unchanged form if the node is an intermediary in passing information between other nodes (is a transit node). All nodes in the network are equivalent to each other. The scheme of the network node is presented in the Fg. 2.

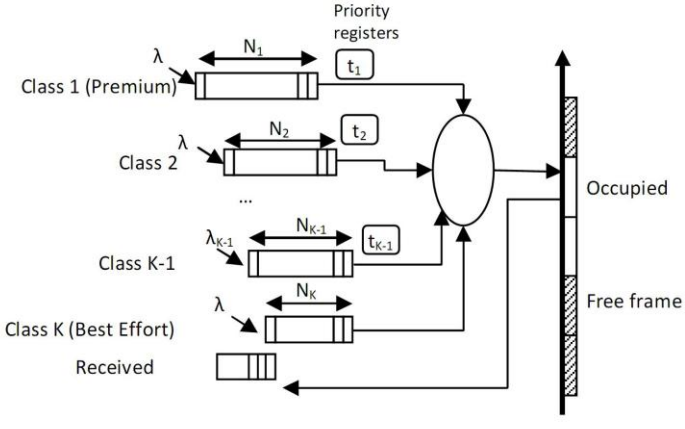



Fig. 2. OPS slotted ring network node.

The architecture discussed in this paper is based on the synchronous slotted-ring architecture proposed in the framework of the ECOFRAME project of the French National Research Agency (ANR) and on the architecture shown in [9] and [10]. The network node is an interface between the electronic and the optical parts of network structure. Packets sent over the ring ore switched optically. Movement of data takes place synchronously – thanks to constant distance between nodes frames in all nodes are received simultaneously.

The node presented on fig. 2 attends network traffic of different quality of service (QoS) classes. For every QoS class *x* there is a separate buffer able to store N*x* blocks of length *b*. Data from customers flow to the node, and are

divided into blocks of constant length *b*. Each block has QoS class assigned to it, and is placed in a buffer appropriate for its class, or lost when the buffer is overflowing.

Frames in optical part of the network can have different sizes (but constant in given network), depending on the distance, and thus the time of trip between nodes. Every frame is able to carry *p* blocks of data. If there are *p* waiting blocks of given class in the node, a packet is formed, which is sent with next possible frame. If there are blocks waiting longer than certain time *t* in a buffer of given class, a packet is formed as well, despite of not being filled in full (*sending timeout* mechanism, described in [11]). The packet contains only blocks of one class which facilitates quality of service management in transit nodes.

## 3. Removing of Overdue Blocks

Due to increasing meaning of multimedia (especially streamed) data transfers in contemporary computer networks, in examined model special QoS class for such a type of data is provided. Blocks of "Multimedia" class have guaranteed time of delivery, however they do not have guarantee of delivery. This corresponds to specific character of streamed multimedia transfers, where – as far as non-delivered packets do not constitute too big part of the whole stream – lack of part of data in the stream is received as temporary reduced quality of transmission, less strenuous than breaks in audio-video transmission, resulting from badly changing delays in data delivery. Multimedia data in proposed solution with removing of overdue blocks (ROB) are delivered at time or never.

Second QoS class provided in the model, "Standard", is foreseen to transfer other (non-multimedia) types of data, as files, WWW pages etc. Packets of "Standard" class, which were completed in the buffer of electronic-optical node, have guaranteed delivery to the receiver, however the time of delivery is not guaranteed. Both guarantee of delivery and guarantee of delivery time do not concern the situation, when data block is rejected on input to the node due to overfilling of the input buffer

Third class of service, "Best Effort", does not give guarantee of delivery or guarantee of delivery time either.

"Multimedia class" is treated as highest priority class, packets of "Standard" class have medium quality, whereas "Best Effort" – lowest. Priority management system is identical to proposed in [10] and examined also in [12]. It provides, that packets of lower classes are not sent until there are no packets of higher classes are ready to dispatch. Additionally, packets of highest class can be inserted into

the optical network in place of lowest “Best Effort” class packets – such a situation means loss of “Best Effort” packet, what is in accordance with assumed lack of delivery guarantee for this class of data.

For ensuring delivery time guarantee for “Multimedia” class blocks, mechanism of removing of overdue blocks (ROB) is proposed. In order to measure time of stay (TOS) of multimedia block, for every block of the class in the buffer a temporary header is created, containing counter storing block time of stay in the buffer. The counters are periodically incremented, in time of arriving of frames to the node. A block, which would stay in buffer too long, would have counter value bigger than assumed. According to delivery time guarantee rules, such a block is not delivered to the receiver, so ROB mechanism does not increments counter of the block, which achieved maximum TOS, but removes it from the queue instead. This block is overdue and has no chance to be delivered on time, so removing it will not decrease the quality of multimedia stream, while it will shorten waiting time for remaining blocks. Freeing place in the buffer will also lower the probability of block loss due to buffer overflow.

ROB mechanism described above generates a risk of removing block, which is waiting for completion of a packet. Paradoxically, this phenomenon will take place by low intensity of data traffic, when time of waiting of the oldest block in buffer for arrival of $p$ (in total) blocks of “Multimedia” class able to form a packet can be bigger than time of overdue of this packet. Loss of data, resultant form this phenomenon, can be reduced by setting a *sending timeout* mentioned above to value lower than time of overdue of multimedia block.

## 4. Simulations scenarios and results

All experiments presented in the paper were performed for the ring consisting of K = 10 nodes. Buffer sizes for particular classes of data, counted in blocks of size b=50 bytes are N1 = N2 = N3 = 250. Optical fibre throughput assumed is 10Gb/s. Time slot was assumed as equal to time of trip of signal between the nodes, amounting to 1us, what gave packet size p = 25 blocks. Sending timeout $t$, causing sending of the packet despite of its incompleteness was set to 40 μs, equally for all queues. One assumed, that data incoming to network node from the client side respond to typical Internet statistics, according to [13]. Traffic for each class of client packets is decomposed as follows: 40% of packets have size of 50 bytes, 30% has size of 500 bytes and the rest are packets of 1500 bytes of size. Packet of size greater than 50 bytes in the moment of arrival to the node is automatically decomposed into blocks of 50 bytes. If the number of free blocks

in buffer is lesser than number of blocks, which the client packet consists of, the packet is rejected.

| Block size | Timeslot | Optical packet size | Client packet sizes | Arrival probability of packet of given size |
|---|---|---|---|---|
| 50 bytes | 1μs | 25 blocks/ 1250 bytes | 50,500,1500 bytes | $\pi_{50}$=0.4, $\pi_{500}$=0.3, $\pi 1_{500}$=0.3 |

Table 1. Global parameters of OPS network model

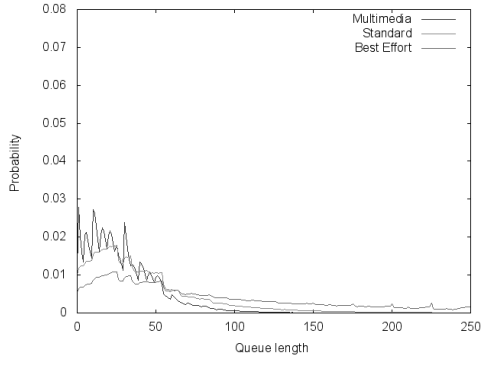
| | Multimedia | Standard | Best Effort |
|---|---|---|---|
| Buffer size N*x* (blocks) | N1 = 250 | N2 = 250 | N3 = 250 |
| Sending timeout | 40 μs | 40 μs | 40 μs |
| Maximum TOS | 70 μs | - | - |
| $\lambda_i$ | $\lambda_1$ = 0.33 λ | $\lambda_2$ = 0.33 λ | $\lambda_3$ = 0.33 λ |

Table 2. Simulation parameters for each class of client packets

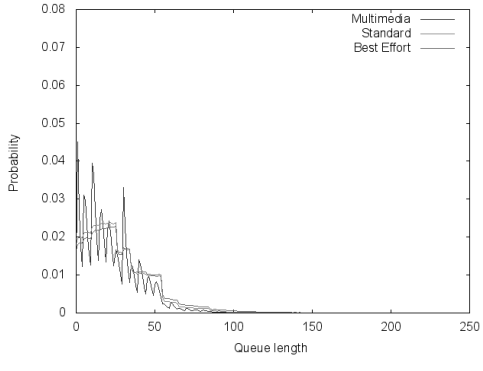
| | High Load | Medium Load | Low Load |
|---|---|---|---|
| λ | 0.10 | 0.08 | 0.05 |
| Network load | 1.062 | 0.494 | 0.184 |

Table 3. Simulation scenarios

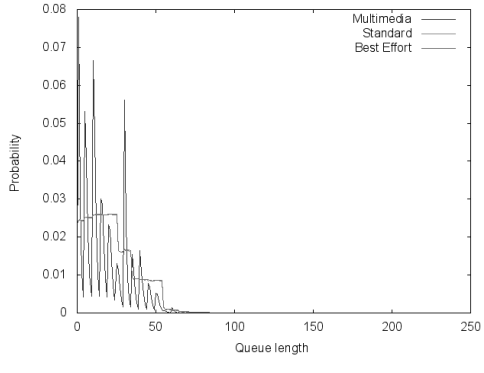
Three network load scenarios were foreseen to examine work of ROB mechanism, compared to simple QoS management with high, medium and low network load. In high load scenario load value slightly exceeding 1.0 were foreseen, medium load was assumed to be close to 0.5 and as a low load we assumed value close to 0.2. Exact simulation parameters are gathered in Tab 1, 2 and 3. Results, obtained with OMNeT++ simulator [14], comparing performance of network with and without PRM are summarized below. Probability of packet loss due to input buffer overflow for different packet classes is shown in Tab. 4. Tab. 5a-5c show average time of block stay in buffer as well as average buffer fulfilment (number of blocks) for buffers of particular classes. Fig.3 shows probability of finding given number of packets in the particular buffers (queue lengths) for different load scenarios (ROB mechanism on).




a)




b)




c)

Fig. 3. Probability of queue length for particular classes of traffic, with high load (a), medium load (b) and low load (c) of the network.

| Class/ QoS | Multimedia/ Simple | Multimedia/ ROB | Standard/ Simple | Standard/ ROB | Best Effort/ Simple | Best Effort/ ROB |
|---|---|---|---|---|---|---|
| High load (λ=0.10) | 1.98873e-05 | 0.00197744 | 0.00212205 | 0.00188898 | 0.0609499 | 0.0609988 |
| Medium load (λ=0.08) | <1e-8 | 3.33692e-04 | 1.35949e-06 | 2.34588e-06 | 3.08546e-05 | 1.90118e-05 |
| Low load (λ=0.05) | <1e-8 | 2.24942e-06 | <1e-8 | <1e-8 | <1e-8 | <1e-8 |

Tab. 4. Probability of packet loss for particular packet classes

| | Simple | | | With ROB | | |
|---|---|---|---|---|---|---|
| | Avg time | Max time | Avg blocks | Avg time | Max time | Avg blocks |
| High load (λ=0.10) | 16.2995 | 145.684 | 27.6826 | 16.1347 | 69.9985 | 27.6447 |
| Medium load (λ=0.08) | 13.4052 | 108.689 | 19.9327 | 13.3735 | 69.9739 | 19.9226 |
| Low load (λ=0.05) | 12.4292 | 71.9863 | 11.5037 | 12.4349 | 68.8162 | 11.4948 |

Tab 5a. Average and maximum time of "Multimedia" class block stay in the buffer with ROB switched off and on

| | Simple | | | With ROB | | |
|---|---|---|---|---|---|---|
| | Avg time | Max time | Avg blocks | Avg time | Max time | Avg blocks |
| High load (λ=0.10) | 28.1959 | 368.248 | 42.7261 | 28.143 | 315.66 | 42.6958 |
| Medium load (λ=0.08) | 18.0467 | 313.861 | 26.066 | 18.0069 | 338.33 | 26.06 |
| Low load (λ=0.05) | 21.5811 | 447.049 | 21.5101 | 21.5482 | 477.749 | 21.5105 |

Tab 5b. Average and maximum time of "Standard" class block stay in the buffer with ROB switched off and on

| | Simple | | | With ROB | | |
|---|---|---|---|---|---|---|
| | Avg time | Max time | Avg blocks | Avg time | Max time | Avg blocks |
| High load (λ=0.10) | 67.5091 | 1199.2 | 78.7522 | 67.4296 | 867.733 | 78.6911 |
| Medium load (λ=0.08) | 20.5938 | 318.854 | 28.5353 | 20.5604 | 306.083 | 28.5967 |
| Low load (λ=0.05) | 21.7729 | 502.409 | 21.6522 | 21.8097 | 454.081 | 21.6647 |

Tab 5c. Average and maximum time of "Best Effort" class block stay in the buffer with ROB switched off and on.

## 5. Markovian analysis

The simulation model in its simple version (without the ROB mechanism) was verified using the method of Markov processes with continuous time (CTMP). Markovian model was implemented using object-oriented OLYMP-2 library [15]

Because of a very big number of Markov chain states (resulting from a very big transition matrix) calculations were made for lesser model, however working according to the same rules described in section 2.

The ring in analytical model consisted of three nodes ($w$ = 3). Each of the nodes owns 2 queues ($K$ = 2): the Standard queue and the Best Effort queue, each being N1 = N2 blocks long. The node is described with state sub-vector ($n1$, $n2$), where $0 \le n1 \le N1$ and $0 \le n2 \le N2$. Each node lodges $(n1 + 1)(n2 + 1)$ states.

The ring itself is represented by 4-elements state subvector: ($f$, $r1$, $r2$, $r3$), where $f$ means phase of Erlang distribution and takes values 1..5. Components $r1$, $r2$, $r3$ take values 0..2 and describe state of particular segments of the ring. Value of 0 means free frame, 1 – "Standard" packet and 2 – "Best Effort" packet. The frame is able to carry a packet consisting of single block. Number of Markov states generated by the ring comes to $k3^w$, where $k$ – number of Erlang phases, $w$ – number of nodes in the ring.

The whole vector has the form:

$$((n1_1, n2_1), (n1_2, n2_2), (f, r1, r2, r3)).$$

Total number of possible states of the model (dimension of transition matrix) amounts to:

$$S = k3^w[(n1+1)(n2+1)]^w.$$

For $k$ = 5, $w$ = 3 and N1 = N2 = 3, number of states S amounts to 552.960. For N1 = N2 = 8 (still very low, comparing to full simulation model), value of S comes to 71.744.535.

Description of all transitions in Markov chain exceeds the scope of the paper. The results of Markovian analysis confirmed correctness of simulation model in simple version of QoS management.

## 6. Conclusions

The removing of overdue blocks (ROB) mechanism of QoS management for network traffic containing multimedia streams is proposed. It is intended for use in synchronous slotted ring networks with optical packet switching. The aim was to ensure delivery guarantees proper for each classes of traffic – delivery time guarantee for multimedia blocks and delivery guarantee for standard blocks. For remaining data of "Best Effort" class no guarantees are secured. As simulation results confirmed by Markovian analysis show, the aim is fulfilled. Especially for the Multimedia class delivery time is secured, however loss ratio increases with grow of network crowding. Still, even by network load as high as 1.06, loss ratio

of multimedia packets does not exceeds 2% of total packets number, what ensures good quality of multimedia transmission.

## References


[1] R. M. Needham and A. J. Herbert. The Cambridge Distributed Computing System. Addison-Wesley, 1982.

[2] IEEE Std. 802.5. IEEE Standard for Token Ring, 1989.

[3] F. E. Ross. Overview of FDDI: The Fiber Distributed Data Interface. IEEE JSAC, 7(7), Sept. 1989.

[4] W. W. Lemppenau, H. R. van As, and H. R. Schindler. Prototyping a 2.4 Gb/s CRMA-II Dual-Ring ATM LAN and MAN. In Proc. 6th IEEE Wksp. Local and Metro. Area Net., 1993.

[5] ISO/IECJTC1SC6 N7873. Specification of the ATMR protocol (v.2.0), Jan. 1993.

[6] Fredrik Davik, Mete Yilmaz, Stein Gjessing, and Necdet Uzun. IEEE 802.17 Resilient Packet Ring Tutorial.

[7] Christophe Mathieu. Toward Packet Oadm. WDM product group, Alcatel-Lucent presentation, Dec. 2006.

[8] Dominique Chiaroni. Optical packet add/drop multiplexers: Opportunities and perspectives. Alcatel-Lucent R&I, Alcatel-Lucent presentation, Oct. 2006.

[9] Thaere Eido, Ferhan Pekergin, and Tulin Atmaca. Multiservice Optical Packet Switched networks: Modelling and performance evaluation of a Slotted Ring.

[10] T. Eido, D. T. Nguyen, and T. Atmaca. Packet filling optimization in multiservice slotted optical packet switching MAN networks. In Proc. of Advanced International Conference on Telecommunications, AICT'08, Athens, June 2008.

[11] F. Haciomeroglu and T. Atmaca. Impacts of packet filling in an Optical Packet Switching architecture. In Advanced Industrial Conference on Telecommunications, AICT, July 2005.

[12] M. Nowak and P. Pecka. Simulation and Analytical Evaluation of a Slotted Ring Optical Packet Switching Network. In Internet - Technical Development and Applications. Series: Advances in Intelligent and Soft Computing , Vol. 64 E. Tkacz, A. Kapczynski (Eds.), Springer Berlin / Heidelberg 2009

[13] IP packet length distribution. http://www.caida.org/analysis/AIX/plen hist/, June 2002.

[14] OMNeT++ homepage. http://www.omnetpp.org.

[15] Piotr Pecka. Obiektowo zorientowany wielowątkowy system do modelowania stanów nieustalonych w sieciach komputerowych za pomocą łańcuchów Markowa. PhD thesis, IITiS PAN, Gliwice, 2002.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 153 – 166

# Internet distance measures in goodput performance prediction

LESZEK BORZEMSKI MAREK RODKIEWICZ GABRIEL STARCZEWSKI

Institute of Informatics,
Wrocław University of Technology
50-370 Wrocław, Poland
*{leszek.borzemski, marek.rodkiewicz, gabriel.starczewski }@pwr.wroc.pl*

**Abstract:** Multiple regression is a popular mean of modelling an unknown function from experimental data. It can be successfully applied to the problem of goodput performance prediction that consists in forecasting application network throughput. We conducted an experiment in PlanetLab testbed where we implemented a system for performing active data transfer measurements to collect various data transfer characteristics that might have an influence on Internet goodput performance properties. Collected data has been analyzed to discover an usable knowledge for goodput performance prediction. A regression set might include many properties amongst which we focused in this paper on network distance measures. We studied RTT, IP hop number, autonomous system number and geographic distance. We constructed different regression sets both for transform regression and linear regression to find out what properties should they include in order to produce satisfactory goodput performance predictions. We show that in order to predict goodput performance we do not need many explanatory variables and both linear and transform regression produce similar results.

**Keywords:** goodput, PlanetLab, transform regression, autonomous systems, Internet performance knowledge discovery, network distance measures, Internet performance prediction.

## 1. Introduction

If we try to describe network performance then we often think in terms of capacity, available bandwidth or TCP throughput. These metrics can be applied both to a single hop and to a link between any given two end stations. Internet growth is undeniable and nowadays we are in urge demand of a metric that will express application performance in the Internet. Goodput can be one of these metrics. It resides in application layer of TCP/IP protocol stack and as such does not have any strong bounds with underlying measures as for example TCP throughput. One may think that if it is located in application layer then it is related to TCP throughput but it is not entirely true. It can happen that an application will make use of a network in a way that will push other competing

TCP flows down or the data flow will not suffice to push the network to its current maximum.

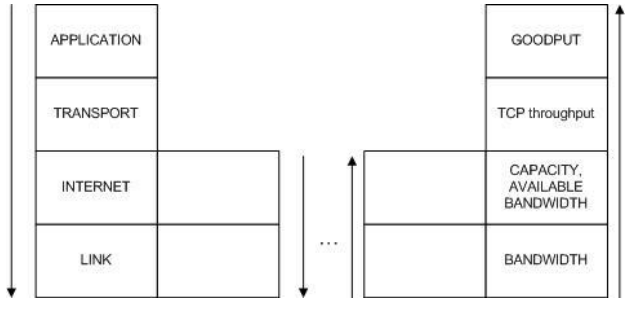



Fig. 1. Network metrics and their corresponding TCP/IP layers

Goodput is given by this simple equation :

$$g = \frac{d}{t}, \tag{1}$$

where $g$ – goodput, $d$ – useful application data, $t$ – transmission time.

By 2013 according to CISCO report [1] 70% of the whole traffic will be generated by peer-to-peer applications with tremendously growing video increase. We need an efficient manner to engineer global traffic so as to diminish growth impact and preserve usability and availability of Internet services. Goodput seems to be the best solution as it describes application performance. We may use goodput in detour routing, overlay network architecting and local peer selection in peer-for-peer networks. In this paper we will try to answer the question - if multiple regression can be a satisfactory goodput performance predictor? Even if the answer is yes, then still we shall think of such factors like: ISP-friendliness and non-invasive information retrieval for the purpose of forecasting. Moreover, gathering data - in particular by active probing impacts the network - therefore it is beneficial for us to know what metric is the most valuable one.

## 2. Related work

Goodput prediction is a relatively new area of research. The researches used to pay more attention to other network metrics such as capacity or available bandwidth. TCP throughput prediction is probably closest to goodput prediction. We may list three distinct approaches and groups of predictors in that area:

- active probing predictors
- predictors using historical data

- hybrid predictors

In case of active probing some number of time-spaced packets is sent and based on the dispersion of packets the forecast is made. Predictors using historical data usually tend to by quite simplistic – in that group fall various kinds of moving averages and median-based methods. It seems that the most accurate are predictors constituting the third group. Based on the history they built the approximation of an unknown goodput function in multidimensional space and before the transfer they send probes to identify the point in the space to obtain the value of the function, which is a current prediction. Here we include for instance linear and non-linear multiple regression.

Work [2] predicts TCP throughput by sending periodic probes of constant size to servers. It exploits correlation between file size and throughput and uses it to built a linear model in which throughput relates to slope of the line. The main contribution of [3] is using knowledge of TCP flow patterns to predict future TCP throughput. The throughput is measured in the intervals of RTT, next this constitutes time series of throughput values that is analyzed to make a forecast. This research in principle try to differentiate the behavior of TCP by the way a throughput changes. In pattern-based approach authors use time series of throughputs measured in fixed intervals equal to RTT time that is estimated at the beginning of prediction. [4] joins together active measurements and history based data to be inputs to Support Regression Vector method which can have multiple inputs and use all of them to make a prediction. The input parameters include queuing, available bandwidth and packet loss. [5] starts with a simple analytical formula extends the model to include history based and active measurements. That results in the hybrid model where at the centre is a regression model – either Holt-Winters or EWMA. In [6] the formula based versus machine-learnt models of TCP throughput predictions are compared. The methods compared include PFTK, SQRT, decision/regression trees and neural network architectures. In all cases methods based on the regression frameworks outperform formula based models and simultaneously have results which are quite close to each other in terms of square error.

Some research try to predict only single path properties rather than goodput in order to design a scalable and low-overhead solution that could be used statically in p2p applications. The most successful projects over time have been iPlane [7] and coordinate-based system Vivaldi [8]. The difficulty in that case is that knowing latency or packet loss does not necessarily lead to accurate goodput estimate and relying on stationariness of the Internet, negates its burstable nature while active probing alleviates this problem to some extent.

There is also a difference between the applications of the methods. Not all of them can be used in environments that are limited in terms of either network resources or calculation capabilities. ISPs head for solutions that avoid

overburdening the network and increasing outgoing and incoming traffic thus prefer history based approaches. However, in closed distributed systems where active probing is not an issue more effective solutions will be welcome. The aim is to construct universal and scalable solution that is easy to incorporate in existing networks and application deployments.

A concept of coordinates-based systems emerged recently and shows some good properties in case of delay and loss rates expressed that way. If it was a metric space we would posses a distance function making things simpler to calculate. In case of goodput it would have to be some kind of inverse though. Mathematical definition of distance exploits a notion of metric space. Metric space is defined as an ordered pair $(M,d)$ where $M$ is a non-empty set and $d$ is a metric (distance) on this set. The conditions that have to be met are defined via the following axioms:

$$d(x,y)=0 \text{ if and only if } x=y \tag{2}$$

$$d(x,z)\geq d(x,y)+d(y,z) \text{ (triangle inequality)} \tag{3}$$

Because the Internet is based on packet switched networks, the packets do not have to follow the shortest path thus the triangle inequality is not satisfied (except for geographic distance). What we call a network distance: RTT, autonomous system number on the path or IP hop number on the path is merely an approximation of the metric. It may be more accurate to speak of the shortest path when we talk about ASs or IP hops number as the Internet can be also perceived as a graph, which lattices are ASs, ISPs, IP addresses or hosts. The difficulty in constructing such a structure pushes as towards predictions using machine learning algorithms.

## 3. Experiment setup

We conducted our experiment in PlanetLab testbed. PlanetLab is a world-wide organization and research network providing academic institutions with access to hosts. Its aim is to enable planetary-scale services development and research that without managed and open environment must be impeded. PlanetLab handles over nodes that belong to user slice with a purpose of active and passive measurements. A derivative of PlanetLab - MeasurementLab - is currently on track with common effort of PlanetLab society and Google as a complimentary solution for passive measurements only.

We designed a prototype of the system for automatic Internet performance knowledge discovery. This implementation could gather data online and update users with current state of the chosen Internet portion. We used 20 nodes located on 5 continents during 3 weeks time span. Out of 20 we picked up 16 for further analysis. During that time we transferred more than 1TB of data with 30 000

downloads. On each node we installed HTTP client and server. Transfer scheduling occurred through our central management station. We enabled concurrent transfers that did not interfere with one another. The application was coded in python, we used PostgreSql and IBM DB2 databases as well as PHP application management tools suite. Our transfers went through 75 different autonomous systems making total of 423 AS-paths.

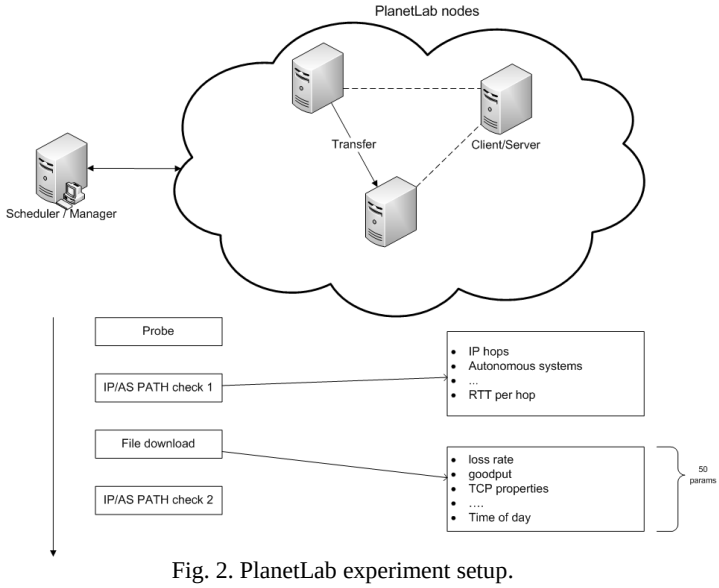



Fig. 2. PlanetLab experiment setup.

## 4. Results

We have defined 125 features describing transfer properties (Tab. 1). With that number of features we needed to have a good selection algorithm. We decided to use mutual information based approach implemented in maximum relevance minimum redundancy [9] algorithm in order to limit the dimensionality of data. We excluded features that describe transfers but are difficult or impossible to obtain in practice. That left us with 67 features (Tab. 1).

Tab. 1. Features measured during transfer.

| **Feature** | **Feature** |
|---|---|
| transfer goodput | day of transfer |
| Probe goodput | hour of transfer |
| integer alias for client IP | number of ases on the path before real transfer |
| integer alias for server IP | number of ases on the path after real transfer |
| Probe file size | number of hops on the path before real transfer |
| lookup time before probe transfer | number of hops on the path after real transfer |
| connect time before probe transfer | average client load in last 15 minutes |

| | |
|---|---|
| total time before the probe transfer is about to begin | average client load in lat 5 minutes |
| pretransfer time plus server overhead | average client load in last minute |
| total time of probe transfer including name lookup etc.. | average server load in last 15 minutes |
| total amount of bytes downloaded for the probe | average server load in last 5 minutes |
| number of packets sent by the server during probe transfer | average server load in last minute |
| number of packets sent by the client during probe transfer | average goodput of last 5 transfers within 2 hours time span |
| number of ACKs sent by the server during probe transfer | average goodput of last 5 transfers within 2 hours time span for similar transfers |
| number of ACKs sent by the client during probe transfer | geographic distance between hosts |
| number of pure ACKs set by the server during probe transfer | number of stop packets sent by the client during probe transfer |
| number of selective ACKs sent by the server during probe transfer | number of tcp init packets sent by the server during probe transfer |
| number of selective ACKs sent by the server during probe transfer | number of tcp init packets sent by the client during probe transfer |
| number of duplicate ACKs sent by the server during probe transfer | idle time of the server during probe transfer |
| number of duplicate ACKs sent by the client during probe transfer | idle time of the client during probe transfer |
| number of data packets sent by the server during probe transfer | tcp throughput of the server during probe transfer |
| number of data packets sent by the client during probe transfer | tcp throughput of the client during probe transfer |
| number of data bytes sent by the server during probe transfer | probe transfer time |
| number of data bytes sent by the client during probe transfer | probe data transfer time |
| number of packets retransmitted by the server during probe transfer | number of ACKs sent by the client during probe transfer |
| number of packets retransmitted by the client during probe transfer | day of probe transfer |
| number of bytes retransmitted by the server during probe transfer | hour of probe transfer |
| number of bytes retransmitted by the client during probe transfer | rtt before probe transfer |
| number of packets sent by the server that arrived out of order during probe transfer | rtt after probe transfer |
| number of packets sent by the client that arrived out of order during probe transfer | number of asses on the path before probe transfer |
| number of pushed packets by the server during probe transfer | number of asses on the path after probe transfer |
| number of packets pushed by the client during probe transfer | average tcp window scale option for the server |

| | |
|---|---|
| maximum segment size of packets sent by the server during probe transfer | average tcp window scale option for the client |
| maximum segment size of packets sent by the client during probe transfer | server loss rate for the probe transfer |
| minimum segment size of packets sent by the server during probe transfer | client loss rate for the probe transfer |
| minimum segment size of packets sent by the client during probe transfer | probe tcp throughput |
| average segment size of packets sent by the server during probe transfer | file size of the real transfer |
| average segment size of packets sent by the client during probe transfer | number of stop packets sent by the server during probe transfer |

After mRmr initial selection we were left with 20 most relevant features. From this set we used backward selection to further reduce the cardinality of regression set yet preserve good predictability characteristics. The order of importance of the features is presented in the table above:

Tab. 2. Importance order of the features

| Nr | Correlation | Corr. Feature | mRmr |
|---|---|---|---|
| 1 | 0.91 | Average goodput of last 5 similar transfers | Average goodput of last 5 similar transfers |
| 2 | 0.9 | Average goodput of last 5 transfers | Number of IP hops |
| 3 | 0.6 | Probe goodput | Geographic distance |
| 4 | -0.5 | RTT measured before transfer | Average goodput of last 5 transfers |
| 5 | -0.4 | Geographic distance | Probe goodput |
| 6 | -0.32 | Number of IP hops | Server alias |
| 7 | -0.22 | Number of ASes on the path | Number of ASes on the path |
| 8 | 0.16 | Probe throughput | Number of ACKs sent by the server during probe transfer |
| 9 | -0.16 | Pretransfer time | Number of packets sent by the server that arrived out of order |
| 10 | -0.16 | Probe transfer time | RTT measured before transfer |
| 11 | -0.15 | Average segment size of packets sent by the server during probe transfer | Client alias |
| 12 | -0.15 | Average segment size of packets sent by the client during probe transfer | Number of selective ACKs sent by the client during probe transfers |
| 13 | 0.07 | Server load during last 5 minutes before probe transfer | Average segment size of packets sent by the server during probe transfer |
| 14 | 0.05 | Pretransfer time plus server overhead | Server probe throughput |
| 15 | 0.05 | Number of packets retransmitted by the server during probe | Number of data packets send by the client during probe transfer |

| | | | |
|---|---|---|---|
| | | transfer | |
| 16 | 0.05 | File size | Hour of transfer |
| 17 | 0.04 | Server loss rate | Client probe throughput |
| 18 | 0.04 | Number of packets retransmitted by the client during probe transfer | File size |
| 19 | 0.04 | Client loss rate | Server loss rate |
| 20 | 0.03 | Number of bytes retransmitted by the client during probe transfer | Load on the client in the last minute |

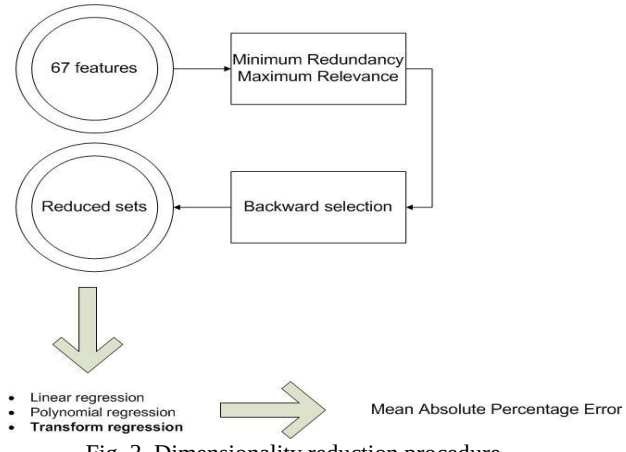



Fig. 3. Dimensionality reduction procedure.

At this point we had 20 features in regression sets, next step required us to decrease their number.

### 4.1 Reducing the dimensionality of data

Our aim was to minimize number of features required for exact prediction. The fewer features are used for forecasting, the better. We decided to further investigate both reduced sets to obtain a smaller yet equally effective explanatory variables set. Let us denote $regression\_set$ as a set of 20 variables chosen in previous step. This set is constructed for each regression. We could approach the problem in two different ways, namely we could either look for the best set $R \subseteq regression\_set$ so that:

$$R^* = \arg\min_R MAPE(R), \tag{4}$$

or assume that we will also be fond of reducing the dimensionality of a set in turn for slight error prediction growth. MAPE stands for mean absolute percentage error.

$$R^{+} = \arg\min_{R} card(R) \wedge MAPE(R) \leq MAPE(regression\_set) + \gamma\,, \tag{5}$$

where $\gamma$ is a permitted error increase. To find such a set we could check all subsets of *regression_set* or use backward selection as and heuristic approach. We used backward stepwise regression with mRmr and Spearman's rank correlation coefficient to assign ranks to variables. We assumed $\gamma = 0$. Here we tried to look at the rankings of the variables paying special attention to distance measures and order of their inclusion.

### 4.2 Importance of measures

In [10] we found out that regression can be an efficient goodput predictor. We noticed that distance measures are particularly important in our previous research [11]. Surprising is the fact that mRmr chooses IP hops number to be the most relevant feature in the set whereas according to correlation it is RTT. This can be because correlation analyses only two variables at once and mRmr indicates which one would introduce more information to current regression set. For example if we had to choose two variables only for prediction mRmr would point to IP hops and distance but if we had to choose only one it would probably not choose IP hops number. What is worth mentioning here is that RTT and distance can be calculated with acceptable error from snapshots of the Internet without direct active probing. We can do the same for ASes but the error tends to be bigger in that case.

| **Rank** | **1** | **2** | **3** | **4** |
|---|---|---|---|---|
| Correlation | RTT | distance | IP hops number | ASes number |
| mRmr | IP hops number | distance | ASes number | RTT |

Tab. 3. Measure importance order

### 4.3 Stepwise regression

Our regression algorithms were included in IBM Data Mining system [12]. We investigated and its implementation in this advanced data mining product. In case of linear regression it uses adjusted Spearman correlation coefficient to rank features. In that case we may override algorithm settings and make algorithm use our set of explanatory variables. In case of transform regression we may only specify a set from which these variables will be drawn. If the underlying selection algorithm performs well it should be close to our findings. At each step we removed a variable with the smallest score. If the prediction error increased we kept the variable, otherwise we removed it from a regression set. The figures

below represent prediction improvement for 12000 records. We used mRmr and spearman correlation for variables selection for linear and transform regression.

Feature selection helps us reduce both dimensionality of data and prediction error. Maximum relevance minimum redundancy approach allows us to get better results than using Spearman's correlation. Going further, we decided to take a look at an error change as a function of measure (RTT, IP hops, ASes and distance). We used mean absolute percentage error for that purpose (MAPE). The results are presented in Fig. 5.

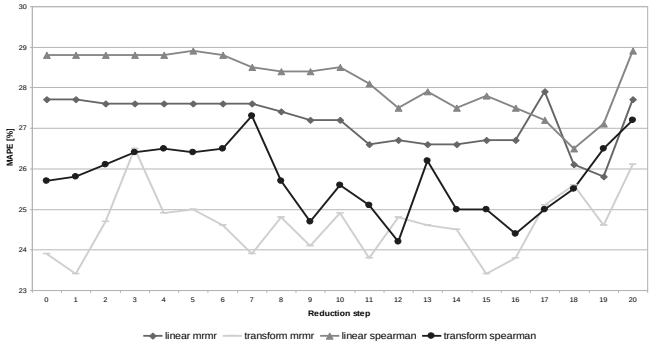



Fig. 4. Prediction error after successive steps.

In Fig. 5a we can see that there is a visible error growth with the number of autonomous systems on the path. However, in case of IP hops the error tends to diminish with its increase. In Fig 5c and 5d there is no apparent trend but in the middle of the scale the error seems to be smaller. It can be caused by the number of observation for each discretized bucket. There are more observations in the middle than at the borders of the scales. In all cases MAPE oscillates at about the same level which implies that if the nodes lay further (a measure has higher values) the prediction error does not change significantly.

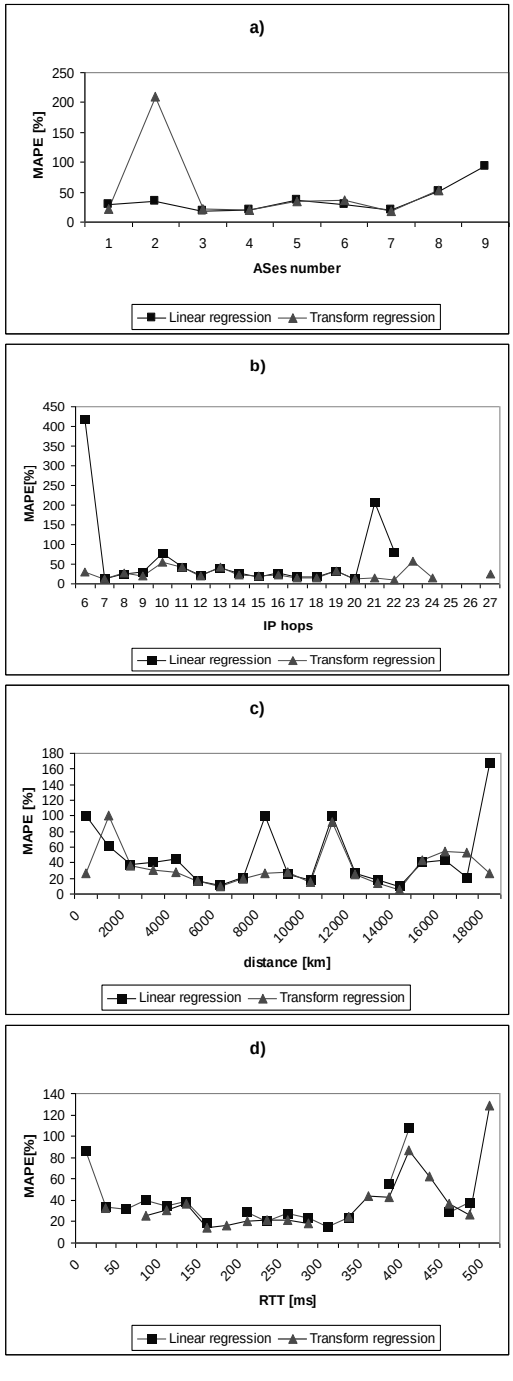



Fig. 5. Prediction error as a function of measure.

At the end we compared convergence speed for: initial sets, sets of 20 most important features and final sets with respect to the number of measurements.

| Regression | Card. | 1k | 3k | 6k | 9k | 12k | 15k | 18k | 21k | 24k | 27k |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Linear | 67 | 137.2 | 61.5 | 51.3 | 45.5 | 40.4 | 40.6 | 38.6 | 37.8 | 41.4 | 40.8 |
| Transform | 67 | 56 | 41.1 | 35.9 | 31.1 | 29.1 | 30.5 | 29.8 | 27.1 | 29 | 29.2 |
| Linear | 20 | 43.6 | 33.7 | 32.3 | 28.5 | 30.3 | 28.6 | 28.8 | 28.6 | 28.3 | 28.2 |
| Transform | 20 | 35.4 | 30.9 | **27.4** | 27.7 | 23.9 | **24.3** | **24.5** | 24.7 | 24.5 | 24.3 |
| Linear | 6 | **34.5** | 30.3 | 27.6 | 26.3 | 25.8 | 27 | 26.4 | 26.3 | 26.1 | 26.1 |
| Transform | 12 | 36.9 | **29.9** | 29.5 | **24.9** | **23.4** | 24.9 | **24.5** | **24.1** | **24.4** | **24** |

Tab. 4. Mean Absolute Percentage Error [%] for different regression sets.

In case of both transform regression and linear regression we can see an improvement over previous sets. Linear regression produces slightly worse predictions but uses only six variables. The prediction error decreased about 3-7 % for linear regression. The number of features decreased from 20 to 6. In case of transform regression the results are similar but with considerable reduction of regression set cardinality after mRmr feature selection. Our research confirmed the conclusions from previous study [11] that both IP hops and AS number on the path are important variables in goodput prediction. The best set for linear and transform regressions contained the features as shown in Tab.5.

| Explanation | linear | transform |
|---|---|---|
| moving average over previous 5 transfers | X | X |
| number of IP hops on the path | | X |
| geographic distance | | X |
| moving average over previous 5 similar transfers | X | X |
| probe goodput | X | X |
| number of autonomous systems on the path | X | X |
| number of ACKS sent by client during probe transfer | X | |
| number of packets that arrived out of order during probe transfer | X | X |
| round trip time | | X |
| average TCP segment size during probe transfer | | X |
| hour of transfer | | X |
| file size | | X |
| probe loss rate | | X |

Tab. 5. Best regression sets.

## 5. Conclusions and future work

Practically the simplest way to predict goodput is using moving averages or their derivatives. Alternatively we may take advantage of regression. We examined transform and linear regression to find out that linear regression gives similar results to transform regression but is far less complex. Transform regression performed better – perhaps this difference would be more apparent with greater number of hosts/transfers. We selected a regression set with maximum relevance minimum redundancy approach. It helps both select an effective regression set and reducing the dimensionality of data what was confirmed by our study. Prediction error in our experiment oscillated around 25%. Taking into consideration possible application in the future we should answer the question - is it good enough? That depends, in some applications more important than small error is preservation of order of hosts that provide us with the same service such as the same copy of a file. Nevertheless linear regression can be useful in certain situations. Examples can include closed service-oriented systems. Such infrastructure does not embrace the whole Internet. Instead it usually contains a few hundred nodes or so. An interesting characteristic of using regression is that the prediction error is at the same level even if network measure has high values.

## Acknowledgements

This work was supported by the Polish Ministry of Science and Higher Education under Grant No. N516 032 31/3359 (2006-2009).

# References

[1] Hyperconnectivity and the Approaching Zettabyte Era, Cisco report, 2009.

[2] Bustamante F.,Dinda P., Dong L.Y, Characterizing and predicting TCP throughput on the Wide Area Network, Tech. Rep. NWU-CS-04-34, Northwestern University, Department of Computer Science, 4, 2004.

[3] Huang T., Subhlok J., Fast Pattern-Based Throughput Prediction for TCP Bulk Transfers, Proceedings of the Fifth IEEE International Symposium on Cluster Computing and the Grid (CCGrid'05) - Volume 1, 2005, 410 – 417.

[4] Barford P., Mirza M. Zhu X., A Machine Learning Approach to TCP Throughput Prediction, ACM SIGMETRICS Performance Evaluation Review, Volume 35, Issue 1, 2007, 97 – 108.

[5] Hu C., Pucha H., Overlay TCP: Ending End-to-End Transport for Higher Throughput, http://www.sigcomm.org/sigcomm2005/poster-119.pdf.

[6] Geurts P., Khayat E., Leuc G., Machine-learnt versus analytical models of TCP throughput, Computer Networks: The International Journal of Computer and Telecommunica-tions Networking, Volume 51, Issue 10, 2007, 2631-2644.

[7] Anderson T., Krishnamurthy A. et al. iPlane: An Information Plane for Distributed Services, OSDI '06: Proceedings of the 7th symposium on Operating systems design and implementation, 2006, 367-380

[8] Cox R., Dabek F., Kssshoek F., Morris R., Vivaldi: A Decentralized Network Coordinate System, Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2004, 15-26

[9] Peng, H.C., Long, F., and Ding, C., Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, 1226–1238, 2005.

[10] Borzemski L., Starczewski G., Application of transfer regression to TCP throughput prediction. 1st Asian Conference on Intelligent Information and Database Systems, ACIIDS 2009.

[11] Borzemski L., Rodkiewicz M., Starczewski G., AS-path influence on goodput prediction. Information Systems Architecture and Technology: Advances in Web-Age Information Systems. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2009. 35-45.

[12] IBM Intelligent Miner for Data. www.ibm.com.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 167 –180

# BitTorrent Based P2P IPTV Traffic Modelling and Generation

ARKADIUSZ BIERNACKI[a] THOMAS BAUSCHERT[b] THOMAS MARTIN KNOLL[b]

[a] Institute of Computer Science
Silesian University of Technology
*arkadiusz.biernacki@polsl.pl*

[b] Chemnitz University of Technology
Electrical Engineering and Information Technology
Chair of Communication Networks
*{thomas.bauschert, knoll@etit.tu-chemnitz.de}@etit.tu-chemnitz.de*

**Abstract:** This paper is a research proposition which aims at the creation of a software framework for the simulation of a P2P BitTorrent-based IPTV system. Using this framework we are able to analyse the statistical characteristics of the P2P IPTV traffic (mean, variation, autocorrelation) and to determine the IPTV traffic flow distribution. This in turn should be used to evaluate the capacity consumption in the underlying IP network so as to support IP traffic engineering actions (in the underlay IP network)[1].
**Keywords:** P2P, IPTV, BitTorrent.

## 1. Introduction

Since the 1950s, television has been a dominant and pervasive mass media; it is watched across all age groups and by almost all countries in the world. Over the last years, many technological advances were produced by trying to meet user needs and expectations in such a widespread media. For example, the large number of users that concurrently watch TV initiated the use of IP multicast by network operators to provide Internet TV (IPTV) services with low transmission costs. Currently we witness how traditional media is converging with newer Internet-based services (e.g. Joost, Zatoo, Livestation, and BBC's iPlayer).

Traditional IPTV service based on simple client–server approach is restricted to small group of clients. The overwhelming resource requirement makes this solution impossible when the number of user grows to thousands or millions. By multiplying the servers and creating a content distribution network (CDN), the solution will scale only to a larger audience with regards to the number of

[1] The work was sponsored by the Polish Ministry of Science and High Education within the grant N516 035 32/3938

deployed servers which may be limited by the infrastructure costs. Finally, the lack of deployment of IP-multicast limits the availability and scope of this approach for a TV service on the Internet scale.

Therefore the use of the peer-to-peer overlay paradigm (P2P) to deliver live television on the Internet (P2P IPTV) is gaining increasing attention and has become a promising alternative [1]. An overlay network is a layer of virtual network topology on top of the physical network (e.g. Internet), which directly interfaces to users. The primary use of such networks so far has been to exchange data over the Internet as a whole [2], although they are also used for data backup [3], caching [4] or computing [5]. Overlay networks allow both networking developers and application users to easily design and implement their own communication environment and protocols on top of the physical network by using its infrastructure and by eliminating or distributing the maintenance costs. P2P technology does not require support from Internet routers and network infrastructure, and consequently is cost-effective and easy to deploy. While traditional P2P file distribution applications are targeted for elastic data transfers, P2P streaming focuses on the efficient delivery of audio and video content under tight timing requirements. P2P-based IPTV service delivery is a new emerging area in this category, see Fig. 1a.

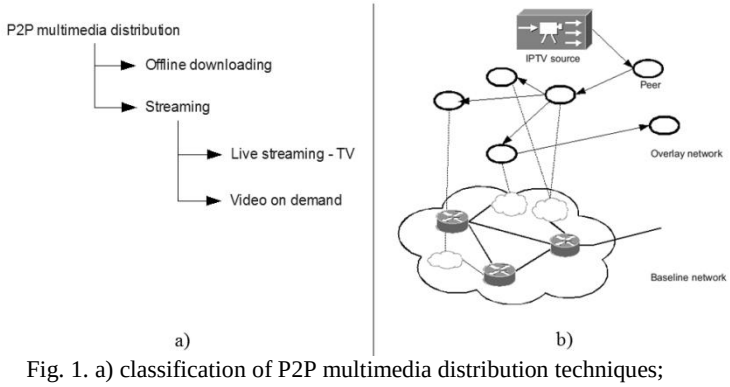



Fig. 1. a) classification of P2P multimedia distribution techniques;
b) P2P IPTV network architecture.

From the technical point of view a local user (or peer) acts both as receiver and supplier of the IPTV data. Connected to “upstream” peers the user receives IPTV data chunks and forwards them to other “downstream” receivers watching the program, which in turn can forward it further down in a hierarchy of peers, see Fig. 1b. Consequently, such an approach has the potential to scale with group size, as greater demand also generates more resources.

The success of P2P IPTV stems from two key characteristics. Firstly, to its ability to distribute resource consumption among the participating entities, thus avoiding the bottlenecks of a centralized distribution. Secondly, to its ability to avoid performance deterioration and service unavailability by enforcing cooperation. Moreover, by using the existing Internet infrastructure as a medium and by exploiting user participation for the creation of the content distribution network, P2P IPTV technologies have innovative potential by making any TV channel from any country globally available and allowing Internet users to broadcast own TV content at low costs. The raising popularity is also confirmed by the amount of new P2P IPTV applications that become continuously available and by the fact that the traffic generated by such applications has recently increased significantly [6]. The most popular P2P-based IPTV frameworks include among others PPLive [7], SOPCast [8] and Coolstreaming [9]. Large volumes of multimedia content from hundreds of live TV channels are now being streamed to users across the world.

## 2. Motivation

P2P applications are posing serious challenges to the Internet infrastructures. One of the major problems is QoS provisioning for P2P services. Unlike file sharing, the live media needs to be delivered almost synchronously to a large number of users with minimum delay between source and receivers. One solution is the application of traffic engineering (TE) methods by Internet Service Providers (ISPs) to improve the perceived QoS of a P2P IPTV system. However the prerequisite to provide better QoS is to measure and/or model the P2P traffic flowing through the network in order to identify potential bottlenecks in the system.

P2P IPTV has drawn interest from many researchers and related traffic studies have concentrated on measurements of real world data traces and their statistical analysis [10][11]. This approach has significant advantages with respect to the reliability of the extracted results, but it is characterized by inflexibility: there is no control over the participating peer characteristics and major protocol variations cannot be studied without first implementing and then deploying them. Moreover, the trace collection process is cumbersome and the data gathered may be incomplete. Additionally, the measurements present a snapshot of traffic without yielding information on how certain system parameters influence the traffic. Such influence factors, among others, might be: 1) the dynamics of the P2P IPTV system wrt. continuous arrivals and departures of nodes (churn); 2) the peer node performance; 3) the IPTV traffic demand (that significantly depends

on the TV channel popularity); 4) the P2P overlay network topology; 5) the P2P protocol (e.g. peer selection, chunks scheduling); 6) the video codec used.

To avoid those disadvantages of real traffic measurements one can create an traffic model. Such a model may either be obtained by analytical methods or by simulation. However, large-scale distributed systems are quite complex and to build an accurate analytical model might not be feasible. Therefore usually analytical models are more or less simplified to be tractable. This leads to models that describe the system at a very coarse-grained level and with many uniformity assumptions. Moreover, the use of analytical models for P2P systems is hindered due to the highly dynamic character of the system: peers dynamically enter and leave the overlay, establish and tear down connections, decide on the preferred pieces of a data stream and chose to exchange data or not with others peers.

Taking into account the above statements, for P2P IPTV traffic studies, simulative methods appear to be the most promising alternative. Simulators allow fast prototyping, provide the possibility to perform large scale experiments, and offer a common reference platform for experimentation.

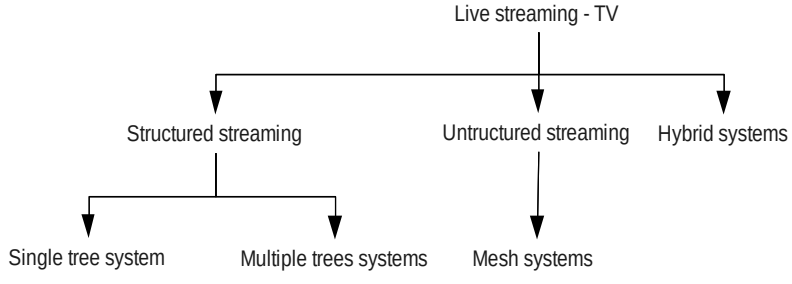



Fig. 2. Classification of P2P live TV streaming systems.

## 3. Planned works

In the project we propose to build an P2P IPTV simulator in order to investigate the traffic behaviour of the P2P IPTV system depending on specific P2P network mechanisms and parameter settings. The challenge lies in combining relevant features from workload modelling, network architecture (topology, protocols), and user's behaviour into a consistent description of a P2P system. The key output of the simulation is the P2P IPTV traffic characteristic. It comprises amongst others the mean traffic intensity, the variation, the autocorrelation and possibly the degree of self-similarity.

In order to build a comprehensive P2P network simulator we have to take into account the behaviour of the P2P system in three main categories: individual peer behaviour, overlay topology generation, and distribution protocol.

The individual peer behaviour can be characterized by state models describing the idle, down, and active states. The model also includes the method how a peer submits and responds to queries and its uptime and session duration. Each of the peers has a certain amount of capacity available. The amount may be different for each of the peers.

The video encoding method will have a considerable impact on the generated traffic. The process of video encoding is quite complicated [12]. Depending on the used encoder there may be huge differences in the output traffic characteristics [13][14]. To avoid an excessive parametrization of our simulation, we assume that only the average traffic intensity is dependent on the choice of the video encoder. All other statistical parameters of the video stream are assumed to remain constant and will be set to some well-known values published e.g. in [15] for popular video codecs.

Peers form an dynamic overlay network topology on top of a transport network. Upon joining the network, peers establish links to a number of other peers in the network and disband the links while leaving. Query and control messages are passed on the links between the peers. The topology of P2P IPTV networks is determined by the specific topology formation method, see Fig. 2.

Distribution algorithms used in non-proprietary P2P IPTV systems are based on protocols used in P2P file sharing networks. One widely used protocol is BitTorrent. It allows users to distribute large amounts of data without demanding not as much network resources from their computers as in case of standard Internet hosting. When applied for P2P IPTV, additional adaptations are necessary to enable data streaming.

Knowing the traffic generated by individual peer nodes we can aggregate it taking into account the current P2P network topology and the distribution protocol. In this way we are able to obtain the characteristics of the overall P2P traffic flows.

Fig. 3 describes the inputs and outputs of the proposed P2P IPTV simulator.

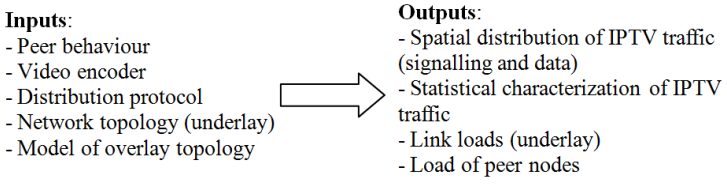



Fig. 3. Inputs and outputs of the proposed P2P IPTV simulator.

P2P traffic can be broadly classified into two categories: signalling and data traffic. The signalling traffic is confined to TCP connection set-up, search queries and query replies. It heavily depends on the type of the P2P network (e.g. structured or unstructured) and used P2P protocol. The data traffic in the P2P IPTV system corresponds to the TV data streams and is much larger compared to the signalling traffic although the signalling traffic packets are sent more frequently [11]. To obtain a comprehensive view of the network behaviour, we will take into account both types of traffic.

The main target of the project is the creation of a software framework for the simulation of a P2P BitTorrent-based IPTV system. Using this framework we are able to analyse the statistical characteristics of the P2P IPTV traffic (mean, variation, autocorrelation) and to determine the IPTV traffic flow distribution. This in turn should be used to evaluate the capacity consumption in the underlying IP network so as to support IP traffic engineering actions (in the underlay IP network).

## 4. Previous works

The popularity of P2P applications has led to some research activities wrt. P2P network simulators. However, to our knowledge, there are currently no simulative implementations of P2P IPTV systems. Nonetheless, there are several popular open-source solutions which are able to simulate P2P systems, some of them also implement the Bittorent (BT) protocol. Many of the ideas, which are used in those implementations may be adopted for the development of P2P IPTV system simulators. Generally P2P simulators range from protocol specific to general frameworks.

One of the earliest attempts to simulate BT-like scenarios is the swarming simulator described in [16]. The simulator does not implement the full BT protocol, and development seems to have stopped. Additionally, the simulator abstracts the BT network entities in a rather unorthodox way, making the simulator more complex and difficult to extend. Another BT-specific simulator is the one used for the work presented in [17]. It is written in the C# language and implements the majority of the BT mechanisms. The use of Microsoft's .NET runtime environment makes platform independence an issue. However, as with [16]. development of the simulator seems to largely have stopped. GPS [18] is a Java-based simulator which incorporates a GUI and logging features, in addition to the core simulation components. The simulator has been used to model the BT protocols, primarily the peer protocol (also known as the peer wire protocol). An implementation of a BT simulator, intended for investigating

modifications to the BT system to provide streaming media capabilities, is presented in [19]. In [20] authors describe and analyse a full featured and extensible implementation of the BT protocol for the OMNeT++ simulation environment. They also show enhancements to a conversion tool for a popular Internet topology generator and a churn generator based on the analysis of real BT traces.

An ambitious general framework is OverSim [21], which extends the OMNeT++ simulation framework. OverSim provides a modular system for implementing both structured and unstructured P2P systems with several P2P protocols such as Chord, Pastry and GIA. However there is no support for BT. Various types of other protocols that are used in underlay networks are provided i.e. transport protocols such as TCP or UDP.

PeerSim [22] is a Java P2P simulator comprising of two distinct simulation engines. One is an efficient cycle-based engine, which does not take into account many parameters in the protocol stack. The other is a more accurate event-based engine, and thus is significantly slower but allows for more realistic simulations.

Here we have described only a few most popular simulators - for a more comprehensive review of P2P simulators see [23].

## 5. Research methods

### 5.1 Implementation issues

In our simulations will use parameters that are obtained from analysis and measurement of the P2P IPTV application GoalBit [24]. GoalBit is capable of distributing high-bandwidth live-content using a BT-like approach where the stream is decomposed into several flows sent by different peers to each client. The system has also a built-in mechanism of perceived quality measurement. Another advantage of the choice of Goalbit is that it is open source and that its specification is well documented. Goalbit has been actively developed and is its references are present in academic papers. Other popular P2P IPTV systems and applications are either based on proprietary protocols like mentioned earlier (e.g. PPLive [7], SOPCast [8] and Coolstreaming) or their specifications are not well documented (e.g. StreamTorrent [25], SwarmTV [26]) or their development works have been stopped (e.g. DistribuStream [27], VidTorrent [28]).

The live media can be captured using different kind of devices (such as a capture card, a webcam, another http/mms/rtp streaming, a file, etc.). It can be encoded to many different formats (MPEG2/4-AVC, VC1, ACC, VORBIS, MPGA, WMA, etc.) and stored in various containers (MPEG-TS, ASF, OGG,

MP4, etc). The streaming encapsulation, defined as GoalBit Packetized Stream (GBPS) is based on the BT protocol. GBPS takes the stream and generates fixed size pieces (chunks), which are later distributed among peers in the GoalBit system. Finally (at the client side) the GoalBit player reproduces the video streaming by consuming the pieces obtained through the GoalBit Transport Protocol (GBTP).

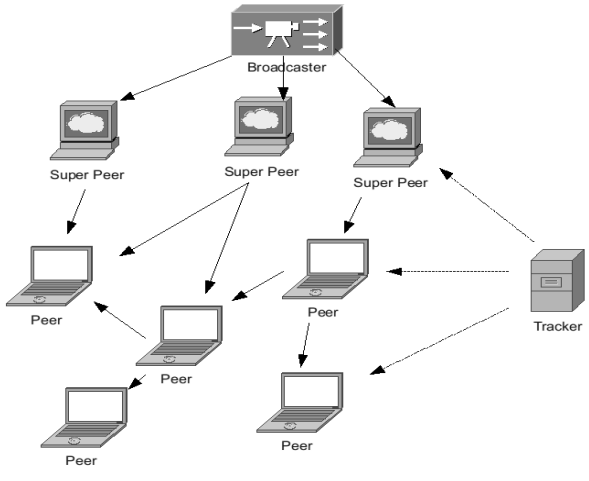



Fig. 4. Components of the Goalbit Architecture (source [24]).

There are four different types of components in the GoalBit network, see Fig. 4. The broadcaster, the super-peers, the peers, and the tracker. The broadcaster is responsible for the content to be distributed i.e. for getting it from some source and to put it into the platform. This critical component plays a main role in the streaming process. If it leaves the network, the streaming obviously ends. The super-peers are highly available peers with large capacity. Their main role is to help in the initial distribution of the content - they get the data streams from the broadcaster and distribute it to the peers which are the final users of the system. The Peers, representing the large majority of nodes in the system, are the final users, who connect themselves to the streams for playout. Peers exhibit a very variable behaviour over time, in particular, they connect and disconnect frequently. The last element, tracker, has the same role than in the BT network. It is in charge of the management of the peers in the system. For each channel the tracker stores a reference to the peers connected to it.

Usually, the following steps are involved in the live content generation and encoding: 1) Content capturing: the content (audio and video) is captured from an analogue signal. 2) Encoding of the analogue signal: in this step the audio and

video signals are separately coded to some known specification (e.g.: MPEG2/4-AVC, WMA). As result at least 2 elementary streams are generated (one for the video and one for the audio). 3) Multiplexing and synchronization of the elementary streams: this step consists of the generation of a single stream of bytes that contains all of the audio, video and other information needed in order to decode them simultaneously. 4) Content distribution: the encoded single stream is distributed using some transport protocol e.g. RTP, HTTP.

### 5.2 Implementation issues

Taking into account the review of existing P2P simulators presented in the previous chapter it is obvious that few simulators are designed as more general tools for system building and evaluation. Even more there is almost no simulator code sharing among the researchers and little standardization of the common practices [29]. Having the choice between adapting already existing BT simulation frameworks, however with poor documentation and without support from a community or creating an own simulation framework from the scratch, we incline towards the second solution. For that, we plan use the open-source simulation tool OMNeT++ propped by dedicated P2P libraries like OverSim [30][21].

OMNeT++ is a open-source, component-based, modular, open-architecture discrete event simulation environment. Simple modules are written in C++ and defined in the NED-language. NED is a simple language developed to easily insert modules into the simulator. Modules interact with each other by means of messages sent through gates and over channels. The simulation executable can also be compiled as a command-line variant to get higher simulation speeds. Statistics are conveniently managed through specific classes provided by the OMNeT++ framework designed to collect simulation data. The modular architecture and structure of the OMNeT++ framework allows to quickly and easyly extend the core P2P algorithms. Furthermore, the fact that OMNeT++ is written in C++ yields the advantage of platform independence and allows us to run simulations on a wide variety of operating systems and architectures.

The simulation of an large scale P2P IPTV scenario (overlay and underlay) of turns out to be a very complex task. Packet level simulations do not scale well enough to be applied to P2P networks with thousands of peers and taking the effects on the underlying IP network into account. We therefore propose an approach to build the simulation in two stages, see Fig. 5. The first stage is simulating the behaviour at the overlay network layer only. In the second stage we plan to introduce more details by additionally regarding the components (routers, transport links) of the underlying network. By this two step approach we

avoid problems related to simulation model construction, performance, and accuracy. In the first stage the system complexity will be kept low. The simulation will be based on pure OMNeT++ API libraries. In the second stage we go deeper towards a realistic modelling of the network components. We will use the INET framework which is an open-source communication networks simulation package for OMNeT++. The INET Framework contains models for several Internet protocols like UDP, TCP, SCTP, IP, IPv6, Ethernet, PPP and several other protocols as well as models of networking devices like routers, workstations and servers. Due to the open-source character of GoalBit some parts of its code may also be implemented directly in the simulator.

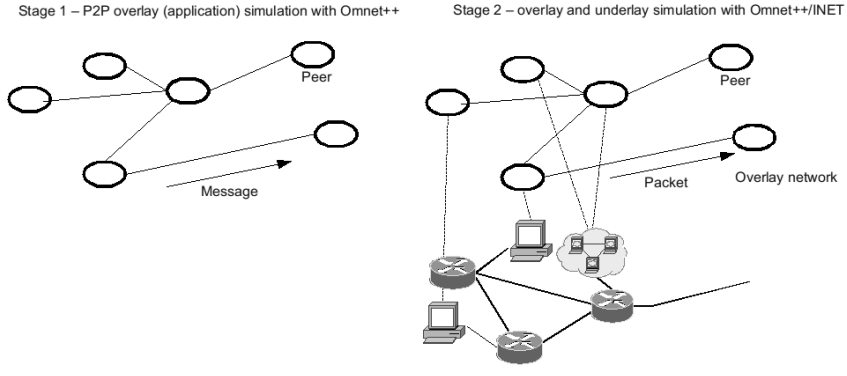



Fig. 5. P2P IPTV simulator implementation stages.

To generate the network topology we intend to use the BRITE generator [31], which allows to export its results into OMNeT++. For the video traffic generation we will use either a specialized multimedia generator [32] or any of the available general network traffic generators with parameters tuned for video traces.

Regarding the simulator design we opt for the following: simplicity for eliminating simulation complexity, modularity for supporting abstractions and facilitating future add-ons, and extensibility for encouraging the model evolution by community contributions. Most of the implementation specific characteristics (i.e., the features left open in the specification), are to be encapsulated and abstracted in order to produce a simulation package that is both concrete and extensible. Thus simulations will run without touching the source code, simply by editing the corresponding configuration files (e.g. .ini and .ned files), while at the same time maintaining the ability to change many aspects of the model behaviour. A large number of parameters will be accessible through an external initialization file and loaded during the simulation start-up. These parameters will

be: amount of different system components used in the simulation run; behaviour of different components; delay and bandwidth of the links between the components; parameters of specific algorithms (peer selection, data piece selection, topology generation, video encoding).

The simulator will collect various statistical traffic data. In a further work we may will also add a GUI to handle the statistics more conveniently.

## 6. Conclusions

In the paper we presented is a research proposition which aims at the creation of a software framework for the simulation of a P2P BitTorrent-based IPTV system. The P2P IPTV simulator might be used by ISPs to support traffic management actions in their (underlying) IP networks. Currently traffic estimations are based on past experiences of ISP or customer subscription information. Alternatively traffic statistics can be obtained by measuring the traffic in the operational network. For the computation of the traffic loads it is required to combine measurement data from multiple locations in the network. By using the traffic and demand values obtained by our P2P IPTV simulator the costly and time intensive measurements of traffic statistics in operational ISP networks can be significantly reduced. Another potential benefit of the proposed P2P IPTV simulator is the possibility to easily investigate novel methods for enhancing quality and efficiency of the P2P IPTV system.

# References

[1] J. Liu, S.G. Rao, B. Li, i H. Zhang, "Opportunities and Challenges of Peer-to-Peer Internet Video Broadcast," *Proceedings of the IEEE*, vol. 96, 2008, s. 11–24.

[2] K. Aberer i M. Hauswirth, "An Overview on Peer-to-Peer Information Systems," *Workshop on Distributed Data and Structures*, WDAS, 2002.

[3] M. Landers, H. Zhang, i K.L. Tan, "PeerStore: better performance by relaxing in peer-to-peer backup," *Peer-to-Peer Computing, 2004. Proceedings. Proceedings. Fourth International Conference on*, 2004, s. 72-79.

[4] T. Stading, P. Maniatis, i M. Baker, "Peer-to-peer caching schemes to address flash crowds," *Proc. 1st International Workshop on Peer-to-Peer Systems (IPTPS)*, Springer, 2002.

[5] D.S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins, i Z. Xu, *Peer-to-peer computing*, Technical Report HPL-2002-57, HP Lab, 2002, 2002.

[6] A. Sentinelli, G. Marfia, M. Gerla, i L. Kleinrock, "Will IPTV Ride the Peer-to-Peer Stream?," *IEEE Communications Magazine*, 2007, s. 87.

[7] "PPLive," *www.pplive.com*.

[8] "SOPCast," *www.sopcast.com*.

[9] X. Zhang, J. Liu, B. Li, i Y.S. Yum, "CoolStreaming/DONet: A data-driven overlay network for peer-to-peer live media streaming," *Proceedings IEEE INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, 2005.

[10] F. Liu i Z. Li, "A Measurement and Modeling Study of P2P IPTV Applications," *Computational Intelligence and Security, 2008. CIS'08. International Conference on*, 2008.

[11] T. Silverston, O. Fourmaux, A. Botta, A. Dainotti, A. Pescapé, G. Ventre, i K. Salamatian, "Traffic analysis of peer-to-peer IPTV communities," *Computer Networks*, vol. 53, 2009, s. 470–484.

[12] T. Wiegand, G.J. Sullivan, G. Bjontegaard, i A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Transactions on circuits and systems for video technology*, vol. 13, 2003, s. 560–576.

[13] M. Dai i D. Loguinov, "Analysis and modeling of MPEG-4 and H. 264 multi-layer video traffic," *IEEE INFOCOM*, 2005, s. 2257.

[14] G. Van der Auwera, P.T. David, i M. Reisslein, "Traffic characteristics of H. 264/AVC variable bit rate video," *IEEE Communications Magazine*, vol. 46, 2008, s. 698–718.

[15] F. Wan, L. Cai, i T.A. Gulliver, "A simple, two-level Markovian traffic model for IPTV video sources," *IEEE Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008*, 2008, s. 1–5.

[16] P. Korathota, "Investigation of swarming content delivery systems," University of Technology, Sydney, 2003.

[17] A.R. Bharambe, C. Herley, i V.N. Padmanabhan, "Analyzing and improving a bittorrent network's performance mechanisms," *Proceedings of IEEE Infocom*, 2006.

[18] W. Yang i N. Abu-Ghazaleh, "GPS: a general peer-to-peer simulator and its use for modeling BitTorrent," *Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, 2005, s. 425–434.

[19] K. De Vogeleer, D. Erman, i A. Popescu, "Simulating bittorrent," *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, 2008, s. 2.

[20] K. Katsaros, V.P. Kemerlis, C. Stais, i G. Xylomenos, “A BitTorrent module for the OMNeT++ simulator,” London, UK: 2009, s. 361-370.

[21] I. Baumgart, B. Heep, i S. Krause, “Oversim: A flexible overlay network simulation framework,” *IEEE Global Internet Symposium, 2007*, 2007, s. 79–84.

[22] M. Jelasity, A. Montresor, G.P. Jesi, i S. Voulgaris, *PeerSim*, http://peersim.sf.net: .

[23] S. Naicken, B. Livingston, A. Basu, S. Rodhetbhai, I. Wakeman, i D. Chalmers, “The state of peer-to-peer simulators and simulations,” *ACM SIGCOMM Computer Communication Review*, vol. 37, 2007, s. 95–98.

[24] M.E. Bertinat, D.D. Vera, D. Padula, F. Robledo, P. Rodriguez-Bocca, P. Romero, i G. Rubino, “GoalBit: The First Free and Open Source Peer-to-Peer Streaming Network,” *LANC '09: Proceedings of the 5th international IFIP/ACM Latin American conference on Networking*, New York, USA: ACM, 2009.

[25] B.B. Gnecco i R. Meier, *StreamTorrent*, http://streamtorrent.sourceforge.net, .

[26] *SwarmTv*, http://swarmtv.nl, .

[27] T. Arcieri, *DistribuStream*, http://faq.clickcaster.com, .

[28] D. Vyzovitis i I. Mirkin, *VidTorrent*, Viral Communications http://viral.media.mit.edu, .

[29] S. Naicken, A. Basu, B. Livingston, i S. Rodhetbhai, “A Survey of Peer-to-Peer Network Simulators,” *Proceedings of the 7th Annual Postgraduate Symposium (PGNet '06)*, 2006.

[30] *Omnet++*, www.omnetpp.org, .

[31] A. Medina, A. Lakhina, I. Matta, i J. Byers, “BRITE: An approach to universal topology generation,” Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2001. Proceedings. Ninth International Symposium on, 2001, s. 346-353.

[32] P. Nain,i D. Sagnol, *ALLEGRO, a multimedia traffic generator*, INRIA, http://www-sop.inria.fr.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 181–194

# Scenarios for Virtual QoS Link implementation in IP networks

PIOTR WIŚNIEWSKI PIOTR KRAWIEC

Institute of Telecommunications
Warsaw University of Technology
Nowowiejska 15/19, 00-665 Warsaw, Poland
P.Wisniewski.4@stud.elka.pw.edu.pl
pkrawiec@tele.pw.edu.pl

**Abstract:** This paper deals with scenarios for establishing Virtual Links (VL) in IP networks. The VL is established between specialized overlay nodes (S-Nodes) at the top of best effort IP network and it aims at improving the packet transfer characteristics provided by underlying IP network. For this scope, VL exploits ARQ selective repeat scheme for recovering packet losses as well as the playout buffer mechanism for assuring constant packet delays. In this paper, we evaluate three scenarios for establishing VLs. The first scenario assumes that VLs are directly established between edge S-Nodes. In the second scenario, we introduce additional intermediate S-Node and split original VL into smaller ones, for shortening ARQ loop. In the last scenario, we propose to replace the additional S-Node with an intermediate transit overlay node (T-Node) with simplified functionality. Applying T-Nodes improves the efficiency of VL by increasing the number of possible retransmissions, which can be performed by overlay nodes, maintaining at the same time delay restrictions required by the VL.

**Keywords:** : QoS virtual link, Quality of Service, overlay network, selective repeat ARQ

## 1. Introduction

Quality of Service (QoS) in IP networks, despite the intense research in recent years, is still not implemented on a global scale. One of the reasons is the lack of appropriate QoS mechanisms in equipment currently used in networks. Although we can use Generic Routing Encapsulation (GRE) tunnels or Multiprotocol Label Switching (MPLS) paths to provide controlled connectivity between selected

This work was partially founded by the Polish Ministry of Science and Higher Education as the grant no. N N517 056635

routers, packet transfer characteristics in established in this way Virtual Connections (VC) may be unreliable. It means, that packets may experience variable transfer delay and packet loss ratio may be at high level. The cause of such disruptions is that packet transmitted along these VCs shares the same network resources, as buffers and link capacities, with other packets in the network.

The presence of such hurdles induces to find another potential way towards QoS IP networks. An approach, which is often taken into account, is the concept called Service Overlay Networks (SON) [1]. The main intention of introducing SON is the possibility of uncomplicated deployment of new capabilities such as multicast ([2][3]), reliability ([4][5]) or anonymity ([6][7]), on the top of the underlying network, without the need of modification of existing infrastructure. One of the value-added services of SON can be the improvement of rough packet transfer characteristics offered by VC established in the underlying network.

The approaches for assuring QoS by exploitation overlay network presented so far can be divided into two groups. In the first one, from among many available paths offered by underlying network with different QoS parameters, overlay nodes try to find the best overlay path to satisfy given QoS requirements and perform appropriate traffic control in overlay network, to not deteriorate QoS guarantees offered by VCs created in underlying networks ([8][9]).

The second group relies on the improvement of packet transfer characteristics offered by underlying best effort network. This task is performed by specialized mechanisms implemented in overlay nodes. In particular, the proposed approaches focus on improving the packet loss ratio offered by the underlying network by applying the Automatic Repeat reQuest (ARQ, [10][11]), Forward Error Correction (FEC, [12][13]) or combination of the both above mentioned mechanisms (so called Hybrid ARQ, [14]).

In this paper we focus on the solution presented in [15], called Constant Bit Rate Virtual Link (shortly VL). The VL is established with means of specialized overlay nodes (S-Nodes), which enhance the Hybrid ARQ scheme with the playout buffer mechanism. It allows us to control both packet losses and the packet delay variation offered by the underlying network.

We consider three scenarios for establishing VLs. The first scenario assumes, that VLs are established directly between edge S-Nodes. In the second scenario, we introduce additional intermediate S-Node and split VL into smaller ones, to shorten ARQ loop. In the third scenario, we propose an approach for establishing the VL with so called intermediate transit overlay nodes, shortly T-Nodes. The T-Node is characterized by simplified functionality in comparison with S-Node. It exploits ARQ mechanism only for improving packet loss characteristics. Notice, that the approach relying on a split of ARQ loop is known from the literature, but in our

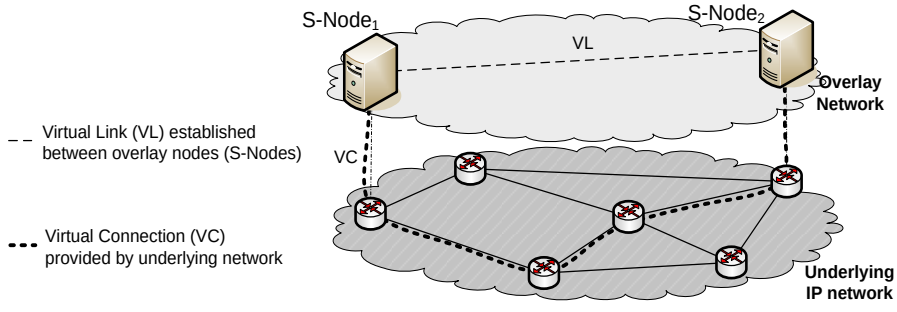



Fig. 1. Overlay network with Virtual Link

case, we are taking into account issues specific to the VL. In the VL, the number of permissible packet retransmissions is limited by allowed transfer delay. Therefore we propose an algorithm to calculate the moment, when T-Node should give up retransmitting a lost packet, because there is no chance the packet will be received by destination S-Node in the assumed time interval. What is worth mentioning, the proposed algorithm does not require synchronization between overlay nodes.

The rest of the paper is organized as follows. Section 2 describes the details of the QoS link concept. In Section 3 we present three scenarios for virtual QoS Link implementation and explain proposed approach to use T-Nodes. In Section 4 exemplary numerical results related to investigated scenarios are reported. Section 5 summarizes the paper and draws some conclusions.

## 2. Virtual Link concept

The VLs are the links connecting two points in IP network using VCs as underlying path for transferring packets (see fig. 1). The motivation for VL is to improve the packet transfer characteristics of VC, typically expressed by the values of QoS metrics such as: IP Packet Transfer Delay (IPTD), IP Packet Delay Variation (IPDV) and IP Packet Loss Ratio (IPLR) [16]. Since VC is usually provided in best effort IP network, $IPTD_{VC}$ can be variable, $IPDV_{VC}$ can be far from zero and $IPLR_{VC}$ can be at the high level (e.g. $10^{-2}$).

The aim of VL is to provide packet transfer characteristics similar to offered by physical synchronous link, those are: (1) constant $IPTD_{VL}$, (2) $IPDV_{VL}$ zero or very close to zero and (3) $IPLR_{VL}$ at very low level. Establishing such VLs gives us possibilities to e.g. design Service Overlay Network with strict QoS guarantees or circuit emulation service ([17]).

To improve the packet transfer characteristics of VC to these required by VL, S-Nodes use ARQ retransmission mechanism for reducing packet losses in con-

junction with the playout buffer to assure constant packet transfer delay.

Fig. 2 depicts the concept of the VL. The incoming packets arrive to the VL with the maximum rate $C_{VL}$. The VL starts the transmission of the packets by putting them to the Selective Repeat ARQ mechanism, where the copies of particular packets are stored in ARQ retransmission buffer (ARQ RTX). To each packet, we add the timestamp with information about packet's arrival time to VL. This timestamp is used later for recovering traffic profile at the output of the receiving S-Node. Optionally, if there are no other packets waiting for transferring in the sending S-Node, transmission of the last arrived packet is repeated. This functionality, called as the Packet Duplication Mechanism (PDM), helps to improve packet loss ratio and avoid possible retransmissions.

The receiving S-Node controls the sequence of the incoming packets to detect lost or duplicated ones. If a packet is lost, then the sending S-Node retransmits it based on the received retransmission request or time-out expiration, after checking if the retransmitted packet has a chance to be received in the assumed packet transfer delay for VL, $IPTD_{VL}$. After successful reception of the packet, the receiving S-Node puts it into the playout buffer, which role is to delay the packet departure from VL for giving chance for retransmissions of lost packets and to mitigate the variable packet transfer time in VC. We take the consecutive packets from the playout buffer maintaining the packet inter-departure delays by using timestamps, which are appointed to the sending S-Node. If a packet arrives too late to maintain the traffic profile, the playout buffer simply drops it. Notice, that sending and receiving S-Nodes do not need to share the same absolute clock. A more detailed description of VL is presented in [15].

One can conclude that VL may work correctly at the cost of increased packet transfer delay ($IPTD_{VL} > IPTD_{VC}$) and decreased capacity ($C_{VL} < C_{VC}$). The capabilities for improving $IPLR_{VL}$ comparing to $IPLR_{VC}$ essentially depend on the $Dmax$ value, defined as:

$$Dmax = IPTD_{VL} - minIPTD_{VC}. \tag{1}$$

The $Dmax$ is the maximum time a packet may spend in the playout buffer to recover the traffic profile. At the same time, it expresses the maximum allowed time for lost packet retransmissions. The sending S-Node submits the packet to transmission only if the difference between the running time and the packet arrival time is less than $Dmax$.

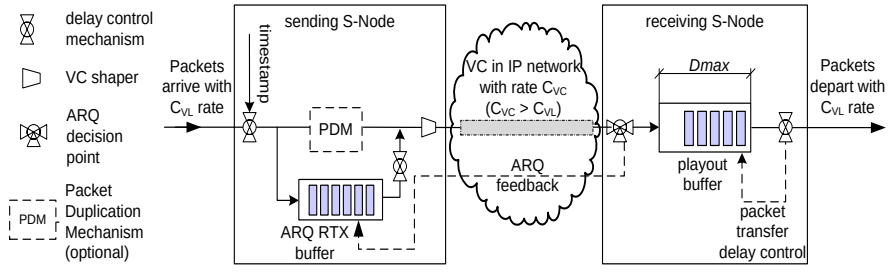



Fig. 2. Virtual Link concept

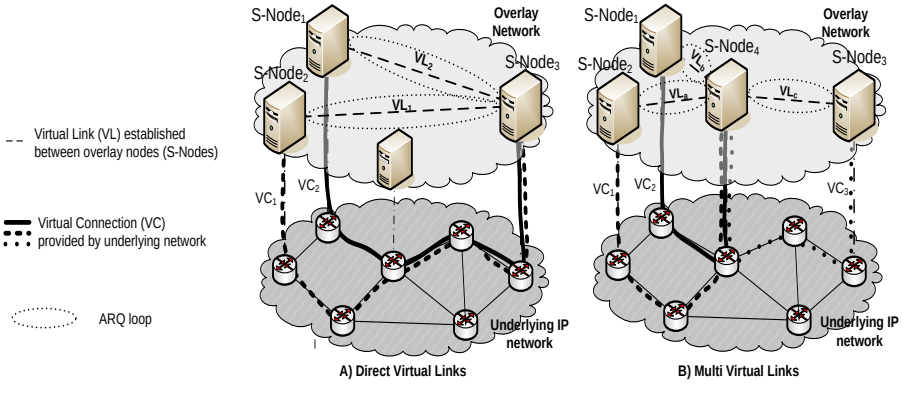



Fig. 3. Virtual Link implementation scenarios

## 3. Scenarios for Virtual Link implementation

The basic scenario for virtual link implementation consists in setting direct VL between investigated nodes, e.g. border routers (see fig. 3A). We call such established VL "direct Virtual Link". Direct VL may include to many physical links. This may lead to relatively long RTT (Round Trip Time) perceived on VL. Since the efficiency of the ARQ mechanism depends on RTT (by efficiency we mean loss recovery time interval), the long RTT limits VL's effectiveness.

To deal with the problem of long ARQ loops (long RTT), we may set a few smaller VLs instead of unique longer VL (see fig. 3B). We call such a scenario "multi Virtual Link". Fig. 3 presents examples of virtual links established between S-Node$_1$, S-Node$_2$ and S-Node$_3$ as direct VLs (fig. 3A) or as multi VLs (fig. 3B). Notice that the second case allows the potential profits from the traffic aggregation on shared VL$_3$ (i.e. multiplexing gain).

However, splitting long VL into smaller VLs is not an optimal solution, because every intermediate S-Node introduces the additional delay to recover input

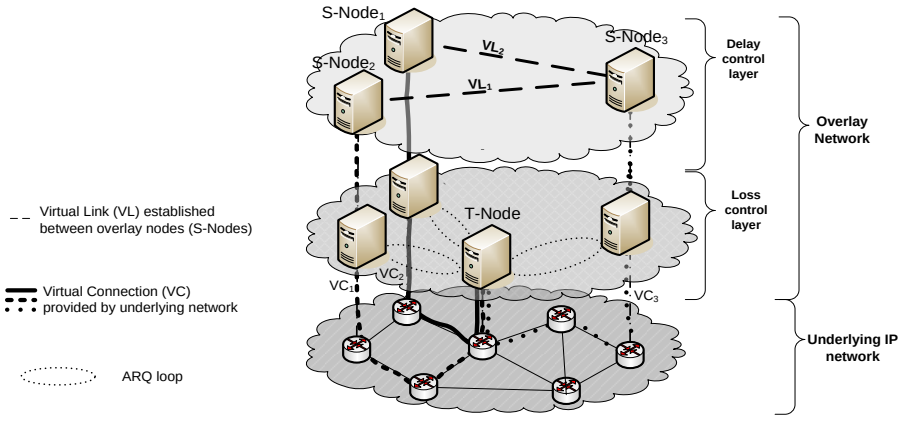



Fig. 4. Semi-direct Virtual Link implementation scenario

traffic profile. Notice, that delaying packets in playout buffer is not a requisite in the intermediate nodes, since profile needs to be reconstructed only at the edge S-Nodes. If we want to maintain end-to-end $IPDT_{VL}$, then splitting VL into smaller ones requires appropriate splitting the end-to-end $Dmax$. Consequently, each "smaller" VL has available only a part of the end-to-end $Dmax$ for recovering lost packets, what finally may diminish the benefits of splitting VL.

### 3.1. Semi-direct Virtual Link with T-Nodes scenario

In order to take advantage of splitting long ARQ loop in the case of Virtual Links, we introduce the intermediate transit nodes, T-Nodes (see fig. 4). The T-Node exploits only a part of functionality of S-Node, which is responsible for improving packet loss ratio. If a packet is lost, the T-Node performs local retransmission, in contrary to end-to-end retransmission performed in direct VL scenario. Moreover, the T-Node does not buffer packets for traffic profile recovery of an input stream, but immediately forwards them (even out of order) to the next overlay node without wasting time in playout buffer; as it happened in the case of multi VL. The whole remaining time budget for considered packet, can be exploited for potential retransmissions and delay variability compensation at the further part of given virtual link. Because packet sequence numbering is made by every overlay node: S-Node as well as T-Node, between the sending S-Node and the receiving S-Node there could be one or more T-Nodes. The VL with intermediate T-Nodes we call "semi-direct VL".

By introducing T-Nodes, we divide original VL functionality into two layers: loss control layer (LCL) and delay control layer (DCL - see Fig. 5). Delay control

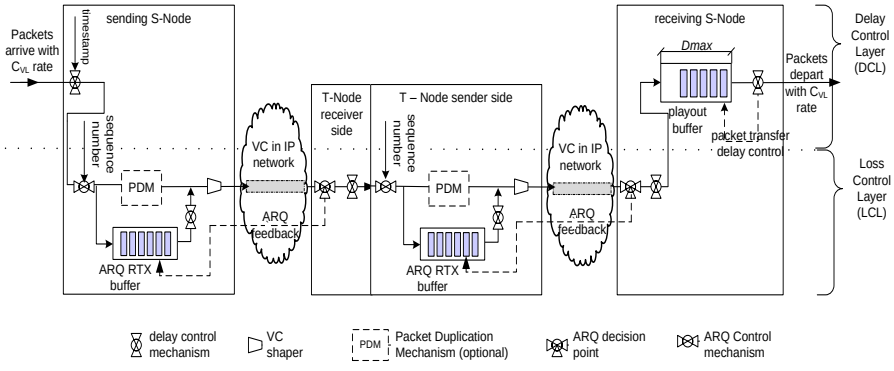



Fig. 5. Virtual Link with T-Node concept

layer is implemented only in the end nodes of connection (which are S-Nodes) and it is responsible for maintaining $IPDV_{VL}$ on zero level (or close to zero). On the other hand, LCL is implemented in every overlay node (S-Nodes as well as T-Nodes) and it is responsible for reducing the packet loss ratio. To determine if any packet is lost, LCL exploits sequence numbers, but retransmission of lost packet is decided on the basis of timestamps from DCL. Since both layers work independently, one overlay node could work as S-Node for some connections and, at the same time, could work as T-Node for other connections.

Let us remark that the introduction of T-Nodes not only effectively shorten the ARQ loops, but also enables traffic aggregation because each ARQ loop controlled by T-Node has own sequence number space. What is worth mentioning, the traffic aggregation at intermediate nodes (S-Nodes as well as T-Nodes) increases overall efficiency of overlay network due to multiplexing gain.

In direct VL scenario, the maximum allowed time in S-Node for recovering the lost packets, denoted as drop interval $d$, is equal to $Dmax$. But in the case of the T-Node, a part of $Dmax$ may be already consumed for packet retransmissions or delay variability compensation on VC between sending S-Node and current T-Node. In order to increase the efficiency of VL mechanism, we propose to calculate the optimal value of $d$ for each packet handled by T-Node. Taking into account the timestamp $t_s(k)$ appointed in the source S-Node and receiving T-Node timestamp $t_r(k)$, we calculate the moment $t_d(k)$, before which the $k^{th}$ packet can exit from T-Node and it still keeps the chance to reach a destination S-Node in the assumed time interval limit ($IPTD_{VL}$), as pointed in formulae 2 and 3.

For the first packet arrival to given T-Node the rule is:

$$t_d(0) = t_r(0) + Dmax, \tag{2}$$

whereas for each following packet the rule is given by:

$$t_d(k) = min[t_d(k-1) + t_s(k) - t_s(k-1); t_r(k) + Dmax]. \quad (3)$$

If $k^{th}$ packet exists from T-Node after $t_d(k)$, it is surely useless for the receiving S-Node, because it arrives too late from the point of view of the traffic profile recovery process. Therefore, $t_d(k)$ determines the moment, when $k^{th}$ packet should be dropped in T-Node because it has no chance to be received by destination S-Node in the assumed time interval limit. On the other hand, for times lower or equal to $t_d(k)$, T-Node can perform retransmissions, if it recognizes $k^{th}$ packet as lost. This means that drop interval for $k^{th}$ packet is given by:

$$d(k) = now - t_d(k), \quad (4)$$

where $now$ denotes current time at given T-Node. Note that our approach uses the difference between sender's timestamps to obtain drop interval, therefore it does not require synchronization between overlay nodes.

### 3.2. Related works

The approaches relying on a split of one multihop ARQ loop into cascade of smaller loops, have been already investigated in literature. Particularly in case of heterogeneous paths, which are established through wireless and wired domains, and therefore consist of links with vastly different characteristics. The example is a set of Split TCP techniques, analyzed e.g. in [18], [19] or [20]. However, authors generally have not considered in their studies the problem of assuring strict packet transfer delay guarantees by controlling a number of retransmissions.

In the context of overlay networks, that issue was investigated by Amir et al. in [10] and [11]. They propose dividing the end-to-end path into a number of overlay hops and recover the losses only on the overlay hop on which they occurred. To bound packet end-to-end transfer delay, they limit number of possible retransmission to fixed value equals one in every overlay hop.

Our solution differs from the presented above, because it allows variable number of possible retransmissions for each packet in every ARQ loop, depending on previous retransmissions of given packet and its current transfer delay. Thanks to it, we can improve efficiency of ARQ mechanism, maintaining at the same time constant packet transfer delay required by the VL.

## 4. Numerical results

In this section, we present exemplary numerical results to illustrate the capabilities of VLs established according to the proposed scenarios, i.e., direct VL,

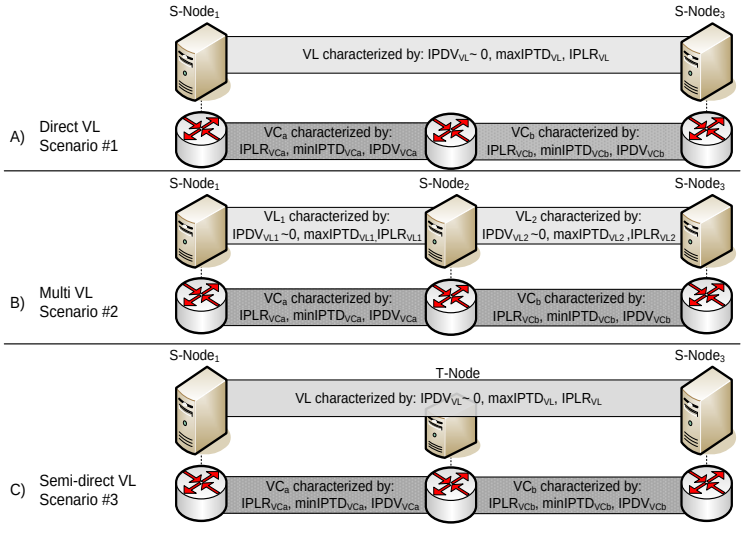



Fig. 6. Simulation scenarios for case 1 and case 2

multi VL and semi-direct VL. We describe performance of VL by the values of $IPTD_{e2e}$, $IPDV_{e2e}$ and $IPLR_{e2e}$, metrics measured between edge S-Nodes. The VLs are created above VCs characterized by capacity $C_{VC}$, loss ratio $IPLR_{VC}$ (independent packet loss probability), $minIPTD_{VC}$ that constitutes the constant part of packet transfer delay on VC, and variable part of packet transfer delay $IPDV_{VC}$. We assume that packets on VC are randomly delayed according to uniform distribution from range from $minIPTD_{VC}$ to $(minIPTD_{VC} + IPDV_{VC})$. In all the simulations, the capacity of the VL, $C_{VL}$, equals $0.6\,Mbps$, while $C_{VC} = 1\,Mbps$ (for each VC). Furthermore, we set the same quality for forward and reverse VCs.

The details of the traffic conditions are the next: we feed the VL with CBR traffic of $C_{VL}$ rate and the packet size equals $200\,B$ (it corresponds to the size of the G.711 VoIP packet with $160\,B$ payload and $40\,B$ of RTP/UDP/IP headers). In all the simulation cases we assume that desired $IPLR_{e2e}$ provided by the VL should be not greater than $10^{-3}$. This value is the upper bound on packet loss ratio defined in ITU-T Y.1541 ([16]) for VoIP connections.

We obtained the presented numerical results from simulation in C++. For a single test case, we simulated at least 4 millions of packets in each of the 10 iterations and calculated the mean values with the corresponding 95% confidence intervals. However, the confidence intervals were not given if they were negligible.

We performed three simulation cases. In the first experiment, we created a network topology consisting of three nodes connected by two VCs: $\mathrm{VC}_a$ and $\mathrm{VC}_b$

| | case 1 | | case 2 | |
|---|---|---|---|---|
| | $VC_a$ | $VC_b$ | $VC_a$ | $VC_b$ |
| $IPLR_{VC}$ | $5.0 \cdot 10^{-2}$ | $1.0 \cdot 10^{-3}$ | $5.0 \cdot 10^{-2}$ | $5.0 \cdot 10^{-2}$ |
| $minIPTD_{VC}$ [ms] | 5 | 25 | 5 | 5 |
| $IPDV_{VC}$ [ms] | 15 | 5 | 15 | 15 |
| scen. #0: $IPTD_{e2e}$ [ms] | 42.3 | | 29.0 | |
| scen. #0: $IPDV_{e2e}$ [ms] | 19.5 | | $28.5 \pm 0.2$ | |
| scen. #0: $IPLR_{e2e}$ | $5.1 \cdot 10^{-2}$ | | $9.8 \cdot 10^{-2}$ | |
| scen. #1/2/3: $IPTD_{e2e}$ [ms] | 120 | | 150 | |
| scen. #1/2/3: $IPDV_{e2e}$ [ms] | ~0 | | ~0 | |
| scen. #1: IPLRe2e | $4.9 \cdot 10^{-2}$ | | $2.8 \cdot 10^{-3}$ | |
| scen. #2: IPLRe2e | $2.5 \cdot 10^{-3}$ | | $2.1 \cdot 10^{-3}$ | |
| scen. #3: IPLRe2e | $4.2 \cdot 10^{-4}$ | | $4.4 \cdot 10^{-5}$ | |

Table 1. Simulations results for case 1 and case 2.

(see fig. 6). We compare three scenarios for establishing VL between two edge overlay nodes (S-$Node_1$ and S-$Node_3$) and one reference scenario:

- Scenario #0: Reference scenario - we do not create any virtual link between S-$Node_1$ and S-$Node_3$; the packets are transferred directly through VCs.
- Scenario #1: Direct VL between S-$Node_1$ and S-$Node_3$ (see Fig. 6A).
- Scenario #2: Multi VL between S-$Node_1$ and S-$Node_3$ that consists of two "smaller" VLs (see fig. 6B).
- Scenario #3: Semi-direct VL between S-$Node_1$ and S-$Node_3$ with one intermediate transit node T-Node (see fig. 6C).

In the simulation case 1, $VC_a$ was characterized by: $minIPTD_{VCa} = 5\,ms$, $IPDV_{VCa} = 15\,ms$. It means, that the packets transferred over $VC_a$ suffered relatively high delay variation (from $5\,ms$ to $20\,ms$) and high $IPLR_{VCa} = 5 \cdot 10^{-2}$. The second virtual connection ($VC_b$) was characterized by $minIPTD_{VCb} = 25\,ms$, $IPDV_{VCb} = 5\,ms$ and $IPLR_{VCb} = 10^{-3}$. We assumed that VLs should provide constant delay equal to $120\,ms$.

The results presented in table 1 show that VLs in each scenario (#1, #2 and #3) assured desirable delay transfer parameters: $IPTD_{e2e} = 120\,ms$ and negligible $IPDV_{e2e}$. However, direct VL (scenario #1) did not improve packet loss ratio comparing to reference scenario #0. The cause is the too small value of parameter $Dmax$ ($Dmax = IPTD_{e2e} - minIPTD_{VCa} - minIPTD_{VCb} -$

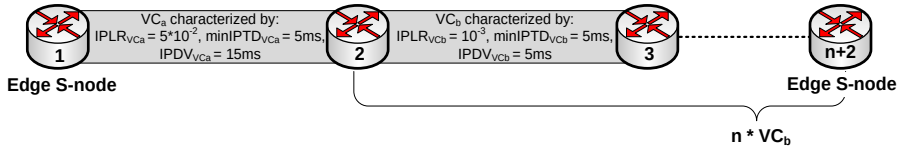



Fig. 7. Simulated network topology for case 3

$transmission\_time$), which does not allow to perform necessary end-to-end retransmissions required to reduce loss rate significantly. In scenario #2, we set up two VLs: $VL_1$ and $VL_2$, with parameters $Dmax_{VL1}$ and $Dmax_{VL2}$, respectively. To keep assumed $IPTD_{e2e}$, we introduced the restriction: $Dmax_{VL1} + Dmax_{VL2} = Dmax$. Since $IPLR_{VCa} > IPLR_{VCb}$ and $IPDV_{VCa} > IPDV_{VCb}$, then we set $Dmax_{VL1} > Dmax_{VL2}$ what allows that $VL_1$ performs more local retransmissions on lossy $VC_a$ than the number of retransmission that $VL_2$ could perform on $VC_b$. This causes an $IPLR_{e2e}$ reduction by about one order of magnitude in comparison to reference scenario #0 (and scenario #1).

Establishing VL according to scenario #3 (semi-direct VL) enables optimal exploiting of the time interval $Dmax$ since T-Node does not buffer packets. This allows for maximum number of local retransmissions during $Dmax$ interval and causes packet loss ratio reduction by about two orders of magnitude in comparison to scenario #1 and by about one order of magnitude in comparison with scenario #2. Notice that in this simulation case only semi-direct VL achieved the desired $IPLR_{e2e} < 10^{-3}$.

In the simulation case 2 we assume that $VC_a$ and $VC_b$ possess the same packets transfer characteristics as $VC_a$ from the first simulation case (see table 1). We assume $IPTD_{VL} = 150\,ms$, what allows for meanly two retransmissions of lost packets in the case of direct VL and results in loss ratio reduction by about two orders of magnitude. Since VCs are the same, we set $Dmax_{VL1} = Dmax_{VL2} = 0.5 \cdot Dmax$ for VLs established according to scenario #2. Such $Dmax$ splitting allows that the mechanisms of $VL_1$ and $VL_2$ may perform as well meanly two retransmissions.

VL established accordingly to scenario #3 allows that lost packets could be retransmitted meanly 4 times what results in an IPLR reduction by about four orders of magnitude in comparison with reference scenario #0 and two orders of magnitude in comparison with scenario #1 and #2. Notice that also in this simulation case only semi-direct VL achieved the desired $IPLR_{e2e} < 10^{-3}$.

In the simulation case 3 we tested, how long VL can be established in each investigated scenario, on the path that consists of one $VC_a$ and a few $VC_b$ (see fig. 7). The $VC_a$ was characterized by $IPLR_{VCa} = 5 \cdot 10^{-2}$, $minIPTD_{VCa} =$

| No. of $VC_b$ links | scen. #1: $IPLR_{e2e}$ | scen. #2: $IPLR_{e2e}$ | scen. #3: $IPLR_{e2e}$ |
|---|---|---|---|
| 1 | $\mathbf{1.1 \cdot 10^{-4}}$ | $\mathbf{1.3 \cdot 10^{-5}}$ | $\mathbf{4.8 \cdot 10^{-7} \pm 1.5 \cdot 10^{-7}}$ |
| 2 | $1.7 \cdot 10^{-3}$ | $\mathbf{1.5 \cdot 10^{-4}}$ | $\mathbf{2.6 \cdot 10^{-6}}$ |
| 3 | $2.8 \cdot 10^{-3}$ | $1.9 \cdot 10^{-3}$ | $\mathbf{2.1 \cdot 10^{-5}}$ |
| 4 | $7.0 \cdot 10^{-3}$ | $6.6 \cdot 10^{-3}$ | $\mathbf{2.2 \cdot 10^{-4}}$ |
| 5 | $5.4 \cdot 10^{-2}$ | $1.9 \cdot 10^{-2}$ | $1.8 \cdot 10^{-3}$ |

Table 2. Simulations results for case 3.

$5\,ms$, $IPDV_{VCa} = 15\,ms$, whereas $VC_b$ was described by $IPLR_{VCb} = 10^{-3}$, $minIPTD_{VCb} = 5\,ms$ and $IPDV_{VCb} = 5\,ms$. We demand that VL guarantees the desired $IPLR_{e2e}$ below $10^{-3}$ and constant $IPTD_{e2e}$ equal to $150\,ms$.

Direct VL (scenario #1) is set between edge S-Nodes. Multi VL (scenario #2) consists of 2 VLs, the first one between nodes 1 and 2, while the second one is set between node 2 and n+2. Semi-direct VL is set between nodes 1 and n+2 with intermediate T-Node set on node 2. We set intermediate S-Node (scenario #2) or intermediate T-Node (scenario #3) on node 2 in order to isolate the lossy $VC_a$ with high packet transfer delay variation.

To achieve desired packet transfer characteristics, (i.e. constant $IPTD_{e2e} = 150\,ms$ and $IPLR_{e2e} < 10^{-3}$) the direct VL can be established from node 1 to node 3 at the most. Introducing another S-Node into the path and splitting direct VL into two VLs enables to establish multi VL between nodes 1 and 4 (one $VC_a$ and two $VC_b$), at the most. This means that it is one hop longer than direct VL. Semi-direct VL with one T-Node enables to extend Virtual Link to node 6 (one $VC_a$ and four $VC_b$), that is three hops longer that standard direct VL and two hops longer than multi VL.

## 5. Conclusions

In this paper we proposed the method to establish QoS Virtual Links in overlay network using transit overlay nodes. Function of T-Nodes is to split ARQ loops and perform merely local retransmissions of lost packets. Although T-Nodes do not possess delay control functionality (DCL) of S-Nodes, they exploit timestamps to determine if a lost packet may be retransmitted or not. T-Nodes take advantage of allowed time for packet losses and traffic profile recovering (denoted as $Dmax$) in a more efficient way than intermediate S-Nodes.

The proposed and investigated solution of establishing semi-direct VL with T-Nodes enables radical reduction of loss ratio in comparison to direct VL and multi VL scenarios and, in this way, it extends the range of possibilities for VL exploiting. T-Nodes can be used to increase efficiency of both single VLs and overlaying network as they enable traffic aggregation.

## References


[1] Duan, Z., Zhang, Z.L., Hou, Y.T.: *Service overlay networks: SLAs, QoS, and bandwidth provisioning*, IEEE/ACM Transactions on Networking 11(6), Dec. 2003, pp. 870–883.

[2] Castro M. et al.: *Scribe: A large-scale and decentralized application-level multicast infrastructure*, IEEE Journal on Selected Areas in Communicastions, vol. 20, no. 8, Oct. 2002, pp. 1489–1499.

[3] Pendarakis D. et al.: *ALMI: An application level multicast infrastructure*, in Proc. of 3rd USENIX Symposium on Internet Technology and Systems, USA, Mar. 2001.

[4] Andersen D. et al.: *Resilient Overlay Networks*, Proc. 18th ACM Symp. on Operating Systems Principles (SOSP), Banff, Canada, Oct. 2001.

[5] Andersen D. et al.: *Improving Web Availability for Clients with MONET*, In Proc. of 2nd USENIX NSDI, (Boston, MA), May 2005.

[6] Clarke I., Sandberg O., Wiley B., Hong T. W.: *Freenet: A Distributed Anonymous Information Storage and Retrieval System*, Int. Workshop on Design Issues in Anonymity and Unobservability, LNCS, Vol. 2009 (2001), pp. 46–66.

[7] Dingledine R., Mathewson N., and Syverson P.: *Tor: The second-generation onion router*, in Proc. 13th USENIX Security Symposium, San Diego, CA, Aug. 2004.

[8] Zhi L., Mohapatra P.: *QRON: QoS-Aware Routing in Overlay Networks*, IEEE Journal on Selected Areas in Communications, Volume 22, No. 1, Jan. 2004, pp. 29–40.

[9] Cai Z. et al.: *IQ-Paths: Predictably High Performance Data Streams across Dynamic Network Overlays*, Journal of Grid Computing, Vol. 5, No. 2, Jun. 2007, pp. 129–150.

[10] Amir Y. et al.: *1-800-OVERLAYS: Using Overlay Networks to Improve VoIP Quality*, In: Int. Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV), Jun. 2005, pp. 51–56.

[11] Amir Y. et al.: *An Overlay Architecture for High-Quality VoIP Streams*, IEEE Transactions on Multimedia, Volume 8, No. 6, Dec. 2006.

[12] Cen Z. et al.: *Supermedia Transport for Teleoperations over Overlay Networks*, Networking 2005, Volume 3462/2005, Waterloo Ontario Canada, 2005.

[13] Shan Y., Bajic I. V., Kalyanaraman S., Woods J. W.: *Overlay Multi-hop FEC Scheme for Video Streaming over Peer-to-Peer Networks*, Signal Processing: Image Communication, Volume 20, Issue 8, Sept. 2005, pp. 710–727.

[14] Subramanian L. et al.: *OverQoS: An Overlay based Architecture for Enhancing Internet QoS*, In: Network System Design and Implementation NSDI'04, San Francisco, 2004.

[15] Burakowski W., Śliwiński J., Bęben A., Krawiec P.: *Constant Bit Rate Virtual Links in IP Networks*, In Proc. of the 16th Polish Teletraffic Symposium 2009, Łódź, Poland, Sept. 2009.

[16] ITU-T Rec. Y.1541 *Network Performance objectives for IP-based services*, 2002.

[17] Bryant S., Pate P. (eds.): *Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture*, IETF RFC 3985 (Informational), Mar. 2005.

[18] Kopparty S., Krishnamurthy S. V., Faloutsos M., Tripathi S. K.: *Split TCP for Mobile Ad Hoc Networks*, in Proc. of the IEEE GLOBECOM 2002, Taipei, Taiwan, Nov. 2002

[19] Bakre A., Badrinath B. R.: *I-TCP: Indirect TCP for Mobile Hosts*, in Proc. of the 15th ICDCS, June 1995, pp. 136–143.

[20] Chakravorty R., Katti S., Crowcroft J., Pratt I.: *Flow Aggregation for Enhanced TCP over Wide-Area Wireless*, in Proc. of the INFOCOM 2003, Apr. 2003, pp. 1754–1764.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 195–212

# Measurement and Analysis of Live-Streamed P2PTV Traffic

Philipp Eittenberger, Udo R. Krieger [a] Natalia M. Markovich [b]

[a]Faculty Information Systems and Applied Computer Science
Otto-Friedrich-Universität, D-96045 Bamberg, Germany,
Email:udo.krieger@ieee.org

[b]Institute of Control Sciences
Russian Academy of Sciences
Moscow, Russia

**Abstract:** We investigate the traffic characteristics arising from the IPTV streaming service SopCast realized by means of a peer-to-peer overlay network. We perform a careful measurement study in a typical home environment and analyze the gathered traces to determine the structure of the P2PTV application, the employed protocol stack and its related message flows. Our data analysis illustrates the features of the dissemination approach, partly validates former findings and provides new teletraffic models of the applied transport service.

**Keywords:** IPTV, peer-to-peer overlay, SopCast, traffic characterization

## 1. Introduction

Currently, the deployment of video content distribution and television services over packet-switched networks employing Internet protocols abbreviated by the term IPTV is a major service component of the triple-play portfolio developed by major network operators and ISPs worldwide. The deployment is fertilized by DSL-based high speed access to the Internet in

*The authors acknowledge the support by the projects BMBF MDA08/015 and COST IC0703.

residential networks and powerful stationary and mobile terminals including multicore PCs, Laptops, netbooks, and 3rd generation handhelds or 4th generation mobile terminals.
From a network perspective the key QoS issues concern the scalability of the transport and the resulting quality of experience of the customers covering both the access and the end-to-end transport delays, the loss of video data as well as the perceived quality of a received sequence of video frames. Apart from traditional client-server architectures new peer-to-peer (P2P) overlay network concepts have been implemented to cope with the enormous bandwidth demand of flash crowds of Internet spectators (cf. [12]). This P2P overlay technology is able to shift parts of the dissemination cost away from the content service provider towards the transport service provider and the users. Integrated into new video portals, the video dissemination systems can offer additional features without any need of set-top boxes exploiting the capabilities of Web 2.0 or 3.0 technology such as news updates, discussion forums, immediate ratings of show events, voting services or user chats. Considering peer-to-peer IPTV (P2PTV) and the transport of recorded videos, the very first streaming services have included numerous Chinese offers such as Coolstreaming, TVAnts and PPLive (cf. [1], [4], [10]). The latter support stored and live video contents with rates between 100 to 800 kbps. In recent years, several new P2PTV providers like Joost, Zattoo and SopCast have popped up (cf. [7], [9]). Their emerging service offers grew out of the experience gained in related P2P projects like Kazaa and Skype.
We have selected a representative subset of these IPTV services, already recorded a moderate number of traces of prototypical streaming applications in 2007 and at the beginning of 2009 and performed a statistical analysis of the peer-to-peer transport service (cf. [7]). In this paper we focus on the P2PTV system SopCast. To improve the understanding of the transport mechanisms of a P2PTV network, we investigate the structure of the overlay and the properties of the packet traffic generated during a SopCast session. For this reason we also analyze the inter-arrival times (IATs) between packets of the flows exchanged downlink towards a stationary peer of interest and develop corresponding teletraffic models.
Similar measurement studies of the four different P2PTV applications PPlive, PPStream, SopCast, and TVants have been performed by Ciullo et al. [3], Hei et al. [4], [5], Liu et al. [8], Silverston et al. [13], and Tang et al. [14] among others. Regarding SopCast the most detailed and informative study by Tang et al. [14] used a PlanetLab implementation to reveal the overlay structure by means of a distributed measurement approach. Inspired

by this concept, we try to analyze the SopCast structure by a passive single-point measurement, to validate Tang's results and to develop new workload models related to the packet level for the exchanged control and video traffic. The rest of the paper is organized as follows. In Section 2. we provide an overview of IPTV services. In Section 3. the measurement and data collection settings are described. In Section 4. the data analysis and modeling of typical P2PTV packet traffic arising from SopCast sessions is discussed. Finally, some conclusions are drawn in Section 5..

## 2. The IPTV concept

Internet Protocol Television or TV over Internet called IPTV denotes the transport of live streams and recorded movies or video clips by means of advanced packet-switched Internet technologies and is a quickly emerging service of the evolving multimedia Internet and next generation networks with their stationary and mobile terminals. ITU-T [16] has vaguely defined the term as multimedia services which are transported by IP-based networks and provide a required level of quality of service, quality of experience, security, interactivity and reliability. It intends to identify scenarios, drivers and relationships with other services and networks to isolate corresponding requirements and to define an appropriate framework architecture. The latter may follow the traditional design of current content distribution systems (see Fig. 1).

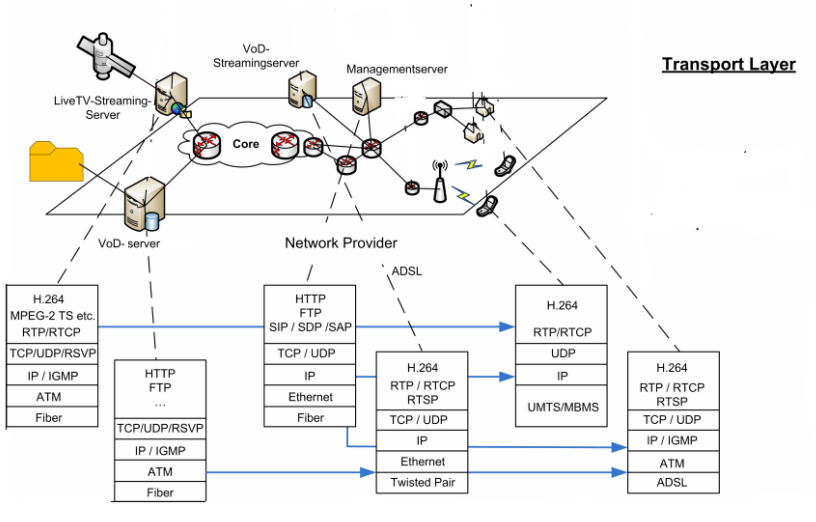



Fig. 1. Transport model of a content distribution system with associated protocol stacks.

The types of content distribution are rather diverse. Following Hossfeld et al. [6], [15] (see Fig. 2), they can be classified according to their real-time properties, the type of use, the dissemination processes established towards the viewing entities, and the connection patterns in the overlay architecture of the application infrastructure which stores and offers the content. The second item allows us to distinguish among video-on-demand (VoD), network based virtualized and personalized online video recorders (nPVR) governed by an electronic program guide (EPG) or the user's interactivity and Live-TV streaming systems. The last item generates the categories of broadcast, multicast and peer-to-peer overlay networks among viewing entities which can offer streaming and download functionalities for requested or recorded video contents. The application architecture can rely on the traditional client-server paradigm or follow a peer-to-peer principle to disseminate the content.

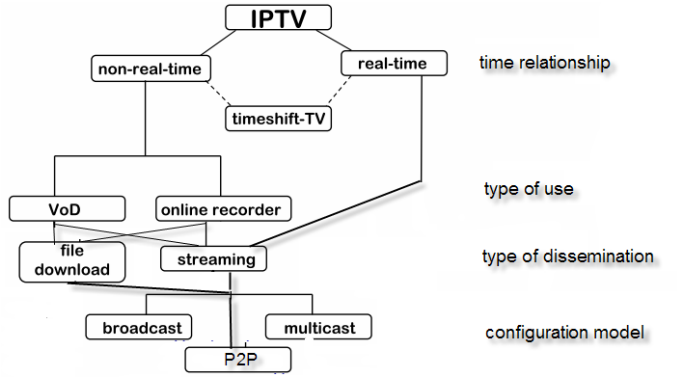



Fig. 2. Classification by the type of the content distribution (cf. [6])

The P2P design of the corresponding systems follows the classical layout of overlay networks with additional streaming functionality. It means that the management, control and primary content distribution is provided to the overlay peers by a highly available cluster of layer 1 servers. After the client has contacted the super node and control servers at layer 1 the seeder list of the feeding peers is distributed by the latter. We refer to the literature for further discussion of the underlying concepts and their first measurements, see e.g. [4], [5], [9], [10], [14] (also [7], [11], [13]).

## 3. Measurement of Live-Streamed P2PTV Traffic

To achieve an efficient performance engineering of a next generation Internet with new triple-play services, we need to study the features, advantages and drawbacks of the content distribution by a peer-to-peer overlay network and to analyze the associated traffic streams of the new services. Therefore, we have performed an extensive measurement study of the P2PTV platforms

Joost, Zattoo, PPLive and SopCast. Our preliminary results concerning the first three systems have been discussed in [7] (for similar studies see [4], [5], [8], [13], [14]). Thus, we focus in this paper on the service offered by SopCast. In this section we discuss our measurement setting of an ADSL access to the Internet by a typical German customer premises network and describe the data collection and preprocessing of prototypical P2PTV traces.

### 3.1. Measurement setting

During the second quarter of 2009 a measurement study has been carried out by the Computer Networks Laboratory of Otto-Friedrich University Bamberg, Germany, to collect traces of representative SopCast sessions. It has focussed on two typical scenarios covering both a wireline and a wireless access to the Internet.
The test bed depicted in Fig. 3(a) has been developed to study first of all the basic operation of the P2PTV streaming system SopCast subject to different access technologies in the typical home environment of a customer.
In the wireline scenario the SopCast client running on a Lenovo Thinkpad T61 Laptop with 2 GHz Intel Core 2 Duo T7300 processor, 3 GB RAM and Windows Vista 64 Bit is connected across an Ethernet link by a home router with ADSL modem to the ISP's public point-of-presence and then by the Internet to the peers of the SopCast overlay cloud. The latter provides an asymmetric Internet access of a representative residential network with a maximal download rate of 6016 kbps and an upload rate of 576 kbps where the actual ADSL rates are smaller due to the impact of damping on the access line and multiplexing effects. In the wireless scenario the client is running on a desktop PC IBM Thinkcentre with 2.8 GHz Intel Pentium 4 processor, 512 MB RAM, Windows XP Home, and attached to the corresponding ADSL router by a Netgear WG111 NIC operating the IEEE802.11g MAC protocol over a wireless link.
Our measurement study aims to determine the following basic issues of the SopCast dissemination concept related to 4 orthogonal criteria:

- *Overlay structure:* The overlay network is analyzed to determine the number of peers per P2PTV session, their geographical distribution, the preference distribution among peers and the realization of a tit-for-tat load distribution strategy.

- *Protocol stack:* We analyze the protocol features applied for control and video content traffic and their related message exchanges. In

particular, the signaling and management message exchange including overlay network establishment, maintenance and control is studied.

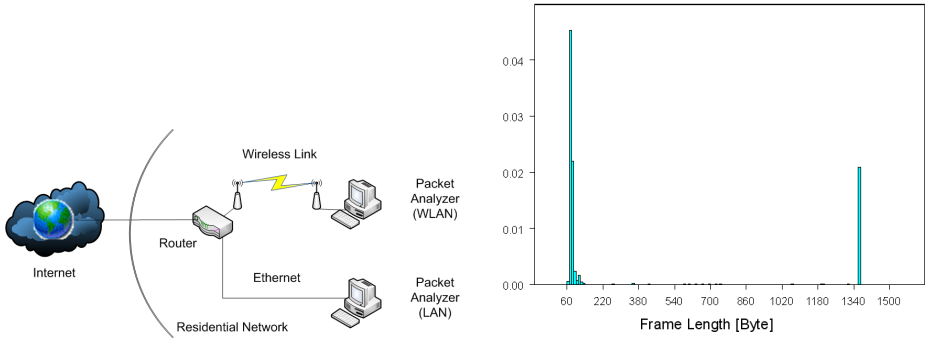
- *Overhead analysis:* The overall amount of transported traffic, its video and control proportion and the resulting overhead are investigated.
- *Traffic characteristics:* The structure of the transferred packet streams including inter-arrival times (IAT) and packet lengths are analyzed. Further, the individual peer connections and the overall traffic stream of a P2PTV session are investigated and include an analysis of the distinct inbound and outbound traffic portions of a selected P2PTV client.

In our further study we focus on the scenario of a wireline access to the Internet.

### 3.2. Collection and preprocessing of the streaming data

Traces arising from live-streamed soccer matches during representative SopCast sessions of approximately 30 minutes have been gathered by Wireshark (cf. [21]). The observation point was the stationary host of the SopCast client with the private address 10.59.1.106 connected by Ethernet to the gateway router. In a preprocessing phase the traces have been cleaned from all non-streaming traffic components and analyzed by Capsa Professional (cf. [17]). In general, the latter sessions have generated a traffic volume of 136 to 138 MB.
Both TCP and UDP are employed by SopCast as transport protocols to perform the topology discovery and maintenance tasks of the peer-to-peer mechanism in the overlay network as well as the control and delivery tasks of the video content dissemination among the involved peers. Preprocessing has also revealed that in both scenarios SopCast uses approximately 1000 different ports to operate the delivery of the video content in terms of smaller chunks (cf. [14]). In the considered private environment without any stringent restrictions on the admitted flows by a firewall only 0.04 - 0.05 % of the traffic volume is generated by TCP to manage the registration and authentication tasks during the set-up of a session and the video selection phase while all other tasks are handled by UDP flows. Considering the directional distribution of the traffic flows it has been recognized that SopCast generates an upward/downward byte ratio of approximately 1:4 whereas the older P2P streaming protocol PPLive has generated for comparable streaming sessions a two times higher ratio of 7:13 and a bigger TCP proportion of 6 - 7 %.
The visual inspection of the Ethernet frame lengths of all flows instantiated

during a session between the observed SopCast client and the communicating peers by a scatter plot and the histogram (see Fig. 3(b) for typical session data) illustrates the dominance of the atoms at the seven characteristic footprints 62, 70, 84, 86, 90, 94 and 1362 bytes by UDP payloads of 20, 28, 42, 44, 48, 52 and 1320 bytes in agreement with [14].




(a) (b)

Fig. 3: Test bed of the wireline and wireless Internet access (a) and a histogram of the frame lengths during a typical SopCast session (b).

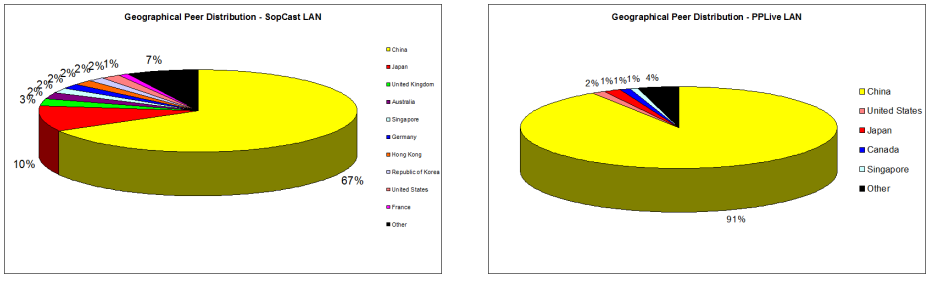
### 3.3. Peer distribution

The preprocessing of the traces by means of MaxMind [20] and GeoIP [19] APIs has revealed the correlation of the IP address of the peers involved in SopCast sessions and the countries of the corresponding hosts. It confirms the strong dominance of Chinese peers (see Fig. 4(a) for a typical constellation). Compared to PPLive the European user community is a bit larger (see Fig. 4(b)).

## 4. Analysis and Modeling of SopCast Traffic

Inspired by the detailed investigation and insights of Tang et al. [14] that were derived from a distributed measurement approach, we intend to validate and extend their corresponding modeling based on our single-site observation.

SopCast establishes a peer-to-peer overlay network on top of the packet-switched TCP/IP transport network and applies a pull mechanism to dis-




(a) SopCast (b) PPLive

Fig. 4. Geographical peer distribution.

tribute the video content among all peers looking to a certain live TV channel or recorded video stream. The latter are described by an object space $\mathcal{O} = \{O_1, \ldots, O_m\}$. The peers form an overlay structure of common interest called *neighborhood community* (cf. [14]). They can be described by a set $\mathcal{P}^{(O_i)}(t) = \{p_1(t), \ldots, p_{n(t)}(t)\} \subset \mathcal{U}$ of peers depending on the viewed object $O_i$ and the observation time $t \geq 0$. We assume that the potential population of peers $p_i$ is large but bounded belonging to a finite universe $\mathcal{U}$. At each epoch $t$ the latter communication relation can be modelled by a *neighborhood graph* $G_N = (V_N, E_N)$, $V_N = \mathcal{P}^{(O_i)}(t)$ (cf. [14]) which dynamically evolves in time since peers may join or leave this community of common interest. In the following we omit the superscript and time parameter since we focus on one specific content at a certain time $t$ or the average behavior over an observation interval $[0, t]$.

### 4.1. Classification and structure of the message flows

We disregard the initialization phase of contacting the SopCast Web page and the registration, authentication and video selection steps which generate TCP traffic flows. In our data analysis we focus exclusively on the following dissemination phase which generates UDP flows. According to the pull mechanism of the overlay system we can classify the UDP traffic into control flows which are arising from the topology generation and maintenance tasks of the overlay structure and UDP flows of video content. The latter transfer a segmented and streamed video object by sequences of replicated chunks which are exchanged among the peers $p \in \mathcal{P}^{(O_i)}$ and reassembled by an observed *home peer* $p_1$ playing out the content to the user.

The control traffic is used to establish and maintain the overlay network. Investigations of Tang et al. [14] have revealed that the generated overlay topology $E_N$ of the *undirected* neighborhood graph $G_N$ can be determined by the communication connections $\{p_i, p_j\} \in E_N$. They are established by means of the UDP transport protocol along a transport path in the underlying router network between the two peers $p_i, p_j \in V_N$. We have validated that the latter association is indicated by the exchange of a Hello set-up message identified by the unique UDP payload of 52 bytes (or, equivalently, a 94 bytes long frame captured on the physical link which is caused by an overhead of 42 bytes due to the headers of Ethernet (14 bytes), IP (20 bytes) and UDP (8 bytes)). The confirmation in a two-way handshake procedure is done by a UDP packet of 80 byte payload or 122 bytes on the wire. Further control information with 58, 60 or 94 bytes UDP payload (100, 102, 136 bytes on wire) can follow.
During their lifetime established connections are maintained by a heartbeat signal. The latter is identified by a keep-alive message exchange of 42 bytes UDP payload (84 bytes on wire) which is acknowledged by the receiver with a normal 28 bytes message. These control flows contribute substantially to the overall message exchange.
A normal control message exchange is confirmed by UDP packets with 28 bytes long acknowledgments (70 bytes on wire). Further unknown control traffic is generated with a payload of 20 bytes. Peer list information is obviously exchanged by UDP packets with 958 bytes payload (1000 bytes on wire).
Our one-site measurements could clearly reveal this flow structures. In this way they confirm Tang's [14] results that are generated by a distributed monitoring approach of higher complexity without a need.

### 4.2. Hierarchical modeling concept for control and video traffic

As pointed out we have to distinguish during a session the control and content flows exchanged between two related peers $p_i, p_j \in V_N$ in opposite directions. This video data exchange defines a *directed video graph* $G_V = (V_V, E_V)$, $V_V \subseteq V_N$ (cf. [14]). $e = (p_i, p_j)$ means that a stream of requested video chunks called *micro-flow* is traveling from peer $p_i$ to $p_j$ on request of $p_j$. If we disregard the exchange of messages and set $\{p_i, p_j\} \in E'_V$ if $(p_i, p_j) \in E_V$ is given, we can embed the resulting undirected video graph as subgraph $G' = (V_V, E'_V)$ into $G_N$.
In SopCast the chunk request of a peer is indicated by a UDP message with 46 bytes payload (88 bytes on wire) followed by a 28 bytes ACK packet.

Our investigations have validated this finding of Tang [14]. Furthermore, it seems that messages of 48 bytes payload (90 bytes on wire) act as trigger for a retransmission of certain chunk sequences.
The directed video graph allows us the describe the exchange of chunk sequences among the peers of the overlay structure by appropriate teletraffic models. For this purpose the video graph $G_V$ is extended to a *flow graph* (cf. [2, Chap. 26.1, p. 644ff]). To create it, we add both a capacity function $c : V_V \times V_V \longrightarrow \mathbb{R}_0^+$ and a flow function $f : V_V \times V_V \times T \longrightarrow \mathbb{R}_0^+$. It assigns to each micro-flow $(p_i, p_j)$ from $p_i$ to $p_j$ its throughput as flow rate $f(p_i, p_j) \geq 0$ and an attribute $t \in T = \{t_1, \ldots, t_h\}$ which determines the flow type. It allows us to distinguish between the different types of control and content flows. The latter rate is recorded by averaging the monitored exchanged packet or byte volumes in a monitoring period $[0, t]$. The capacity function $c$ of a link in the overlay network is determined by the bottleneck capacity of the underlying path $w(p_i, p_j)$ in the router network and normally unknown. However, one can additionally measure it by appropriate bandwidth estimation techniques and determine the length (i.e. hop count) $l(p_i, p_j)$ and round trip delay $R(p_i, p_j)$ of the route in the transport network by path-pinging the corresponding hosts of the peers. Currently, this options are developed in a corresponding monitoring tool based on the JAVA API of Wireshark. To get it for both directions, it is however necessary to implement a measurement instrumentation in the peers feeding the home peer. This requires an expensive instrumentation of the complete infrastructure, for instance, by PlanetLab.
In summary, we obtain a hierarchical model in time and space. In the neighborhood graph $G_N$ of an object $O_i$ we can assign and determine by our measurements the size of $V_N$ and the sojourn time of a peer $p_i \in V_N$ as well as the lifetime $L(p_i, p_j)$ of each connection $\{p_i, p_j\} \in V_N$ exploiting the Hello and keep-alive flows. It can be used to determine the *churn rate*, i.e. the average number of peers changing the affiliation during the measurement interval (cf. [14]).
Further the related inter-arrival times of the connection set-up requests indicated by the 52 bytes Hellos can be determined for individual peer relations $(p_i, p_j) \in G_V$ and superimposed outbound flows $(p_i, V_V) \equiv \{(p_i, p_j) \in E_V \mid \forall p_j \in V_V\}$ or inbound flows $(V_V, p_i) \equiv \{(p_j, p_i) \in E_V \mid \forall p_j \in V_V\}$ of a home peer $p_i$.

### 4.3. Classification and characterization of P2P packet traffic

The weighted directed graph $G = (V_V, E_V, c, f)$ arising from the flow graph in a measurement period $[0, t]$ will enable us to analyze the structure of the flows of control and content traffic in a SopCast overlay network and to characterize their major properties. According to our experience the corresponding preliminary results presented in [1] based on simple estimates of the transferred volumes are not accurate enough. Therefore, improved bandwidth estimation techniques need to be applied along the routes between content exchanging peers. At present, we can only provide a preliminary data analysis reflecting these structural properties of a session since the bandwidth estimation analysis is under development and only partial results are available yet.

#### 4.3.1. Hierarchical peer structure of a SopCast session

An analysis of our traces has shown that connections to more than thousand different peers are established during the lifetime of a video session which embed an observed home peer into a dense mesh of the overlay network (see Fig. 5).

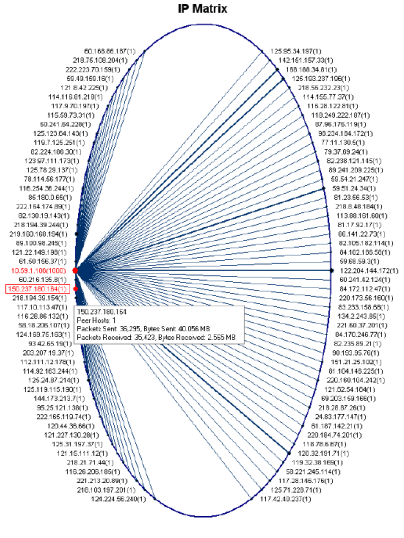



Fig. 5: Partial view of the IP matrix of hosts connected to the home peer at 10.59.1.106.

The investigation of the transferred accumulated volumes among the peers of the video graph $G$, in particular the number of packets and the byte volumes flowing inbound and outbound to a home peer, illustrated in our example by the client at 10.59.1.106 , reveal the hierarchical structure of the overlay. This criterion allows us to distinguish three levels of peers associated with a home peer in $G$ during a session, namely *super peers*, *ordinary peers* and *supplementary peers*. The first provide more than 10 % of the volume exchanged with a home peer and consist of less than 5 peers with high upload capacity, the second group offers between 1% and 10% and the third group less than 1 % (see Fig. 6).

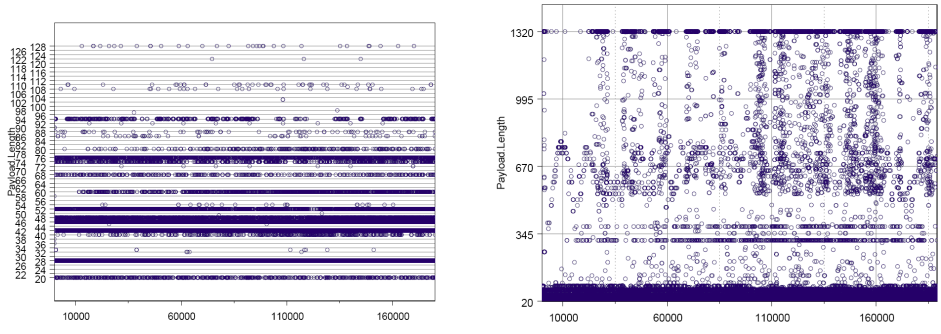
In our example the home peer is located in a residential network with limited access capacity of the ADSL line in the range of 500 kbps. Such nodes do not contribute significantly to the exchange of chunk flows as illustrated by Fig. 6. One can see that the contacts with the super peers are dominating the exchanged volumes confirming former findings (see [14]). They demand the development of an adequate teletraffic model of the downloaded chunk streams. Moreover, we notice that a tit-for-tat strategy is not applied by the normal or supplementary peers during the dissemination of video data.

### 4.3.2. Control flows of a SopCast session

The scatter plot of the frame lengths of all Ethernet packets captured during a representative TV session and the corresponding histogram of the lengths of all exchanged frames illustrates the dominance of the control flows transfering ACKs at 28 bytes UDP payload, the keep alives at 42 bytes, the video requests at 46 bytes, compared to the majority of chunks with a maximal size of 1320 bytes carrying the video content (see Figs. 7(a), 7(b)).
Furthermore, it supports the already proposed concept (cf. [14]) to distinguish control traffic by a UDP payload length less than 150 bytes (192 bytes on wire) and to classify higher packets with payload length beyond 150 bytes as content traffic. Hereby traffic fulfilling some specific functions like the peer list exchange which we suppose to arise at 958 bytes UDP payload may be misclassified. We are convinced that due to its sporadic nature the resulting mistake in traffic classification at the packet or flow levels can be disregarded.
An analysis has also shown that super peers are contacted by a requesting peer (i.e. *home peer*) in a periodical schedule of around 10.26 ms. Normal peers feeding less to other peers are obviously contacted on the basis of a ran-

| Name | Percentage Inbound | Percentage Outbound | Bytes | Packets |
|---|---|---|---|---|
| 10.59.1.106 | 83.314% | 16.686% | 138.013 MB | 369,347 |
| 150.237.180.164 | 1.858% | 29.024% | 42.621 MB | 71,718 |
| 122.204.144.172 | 1.428% | 22.237% | 32.661 MB | 55,485 |
| 219.160.168.194 | 0.851% | 12.133% | 17.919 MB | 32,515 |
| 218.194.39.154 | 0.332% | 2.598% | 4.044 MB | 11,926 |
| 120.32.191.71 | 0.086% | 1.083% | 1.614 MB | 3,229 |
| 168.188.34.81 | 0.081% | 1.000% | 1.492 MB | 3,062 |
| 59.51.24.34 | 0.067% | 0.864% | 1.285 MB | 2,547 |
| 125.193.237.196 | 0.068% | 0.806% | 1.206 MB | 2,549 |
| 125.24.87.214 | 0.779% | 0.052% | 1.146 MB | 2,267 |

Fig. 6: Top 10 hosts feeding and fed by the home peer at 10.59.1.106 during the lifetime of a session.




(a) outbound control traffic

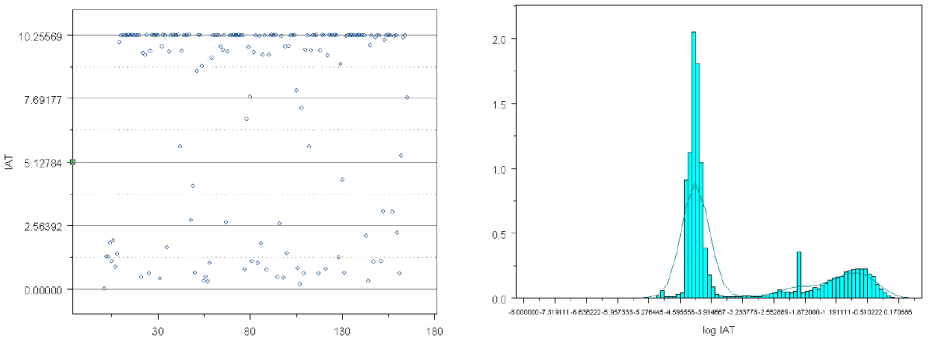
(b) overall outbound traffic

Fig. 7. Scatter plot of the packet lengths of a typical flow from the home peer to a neighbor.

dom schedule. Consequently the overall keep-alive traffic flowing outbound from an observed home peer to all its neighbors shows a random structure, too. In our example, for instance, we have observed a mean inter-arrival time between issued keep-alive requests of 51 ms, a median of $6.5 \cdot 10^{-5}$ and a standard deviation of 127 ms. Similar results apply to individual or aggregated keep-alive flows reaching the home peer.
At the IP packet level the outgoing aggregated keep-alive traffic of a home peer can be modelled in a first step by a renewal stream with bivariate inter-arrival time density of normal and gamma shaped components and independent constant IP packet sizes of 70 bytes. An ln transformation of the measured inter-arrival times has revealed this structure (see Fig. 8(b)).

#### 4.3.3. Video traffic characteristics of a SopCast session

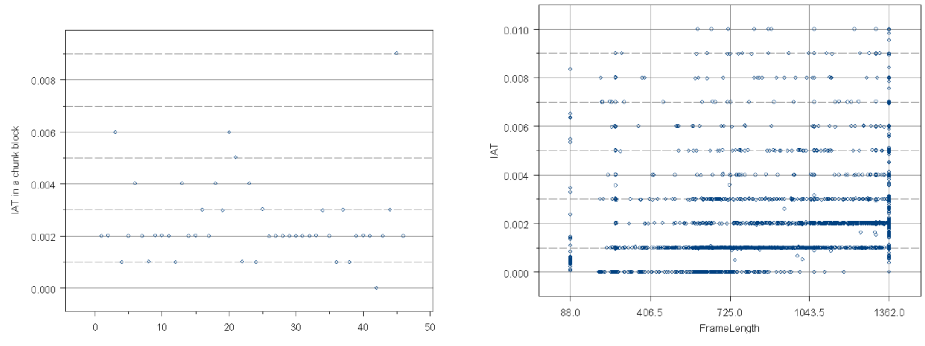
In the following we focus on the packet transfer processes feeding a peer on the downlink to the home environment. The reason is that it represents the more important direction of the information exchange which should be handled effectively by the transport infrastructure of a network operator. Studying a P2PTV session at the time horizon of the UDP flows from the perspective of an initiating peer $p_1$, we note that first several control flows are instantiated for the initial opening of several connections to the ordinary and super peers indicated during registration in the downloaded peer list. After the establishment of the neighboring graph the communication on the video graph is started. It implies the simultaneous existence of several




(a) (b)

Fig. 8: Scatter plot of the IATs of a keep-alive stream sent by the home peer at 10.59.1.106 to a super peer at 219.160.168.172 (a) and a histogram of the ln transformed IATs of the aggregated keep-alive stream sent outbound by the home peer (b).

micro-flows $f(p_j, p_1)$ which exchange video chunks between the monitored home peer $p_1$ and its neighbors $p_j$. Each micro-flow consists of multiple consecutive packet sequences. Each one comprises a pair of a video request on the uplink and the requested chunk packets on the downlink. Exchanged between the feeding peers $p_j$ and $p_1$, they create the superimposed inbound flow $(V_V, p_1)$ to the home peer. Similar to client-server traffic modeling, theses pairs can be identified by a request Ethernet packet of length 88 bytes which decomposes the incoming flow of a certain peer $p_j$ into several chunk sequences. They are answered by video contents with a maximal Ethernet packet length of 1362 byte, i.e. 1320 bytes UDP payload, and acknowledged by $p_1$ with packets of 28 byte UDP payload. This structure can be used to segment a micro-flow of incoming content or outgoing ACK packets from a home peer $p_1$ to any other peer $p_j \in V_V$ according to the request packet sequence of the home peer. Focussing on the dominant inbound content flows sent by super peers, e.g. 150.237.180.164, an investigation of the inter-arrival times has revealed that the chunks are sent at a major time horizon of 2 ms (see Fig. 9(a)). Due to buffer effects very small inter-arrival times can occur (see Fig. 9(b)). A logarithmic transformation of the inter-arrival times between chunks illustrates that at this transformed time scale a log-logistic inter-arrival time density may be proposed as first step to model the packet flow of video contents by analytic means. Taking into account the dominant maximal chunk size of 1320 bytes, a corresponding renewal stream with a




(a) (b)

Fig. 9: Scatter plots of the IATs of a chunk stream sent by to the home peer at 10.59.1.106 by a super peer at 150.237.180.164 (a) and the IAT vs. the frame length (b).

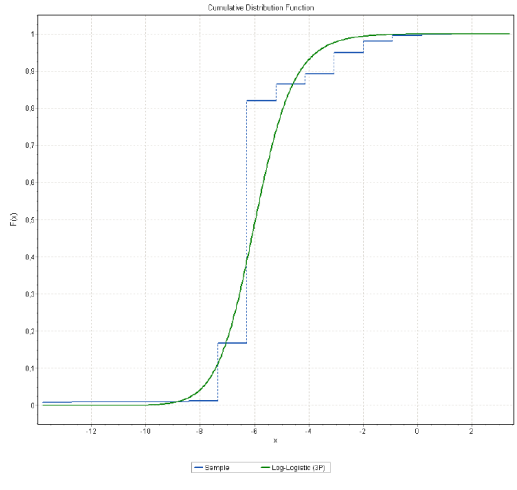



Fig. 10: Adaptation of a log-logistic distribution to the ln transformed IATs of a chunk stream sent by to the home peer at 10.59.1.106 by a super peer at 150.237.180.164.

constant packet size of 1348 bytes is therefore proposed as workload model at the IP layer (see Fig. 10).

In summary, the outcome of the SopCast overlay structure yields a hierarchical aggregated flow model of superimposed micro-flows in upward and downward direction to a home peer. However, the control and content flow components of the opposite directions are correlated due to the operation of

the push protocol. Thus a complete analytic modeling at the packet level seems to be beyond the capabilities of our current modeling techniques. As a consequence the packet process of the superimposed inbound flows towards a home peer should be described in a first step by an aggregated marked renewal stream with heavy-tailed inter-arrival times and the maximal packet lengths of the video chunks as rewards. Here the number of flows can be modulated by the actual number of active super peers.

## 5. Conclusions

Considering the current development of new multimedia services as an integral part of Web 2.0, application service providers and network operators have assigned the highest priority to the integration of triple-play services including VoIP, video streaming and video on demand as well as IPTV into the portfolio of their next generation networks. In this regard new video portals like Joost, Zattoo, PPLive or SopCast offering streaming of stored or live TV contents have become a rapidly emerging new commercial offer in the Internet.

In this paper we have investigated the traffic characteristics of the IPTV streaming service SopCast whose content dissemination and control planes are implemented by means of a peer-to-peer overlay network. We have followed the classical approach of a passive measurement at a single point in the overlay network and focussed on the data analysis and teletraffic modeling of the exchanged control and content flows at the packet level.

First, we have described the home scenario of our measurement study and then the analysis of the gathered traces to determine the structure of a SopCast session and the employed protocol stack with its related message flows. Confirming previous results, e.g by Tang et al. [14] and Ali et al. [1], we have realized that only a small fraction of peers mainly consisting of powerful machines with high upload capacity have contributed to the video dissemination. Moreover, we have seen that the UDP flows of the video data messages of a SopCast client are governed by a very complex distribution of the inter-arrival times between the datagrams. The packet length of the latter has major clusters of the UDP payload around 28, 42, and 52 bytes covering the control traffic and at 1320 bytes for the content traffic of transferred video chunks, respectively. Moreover, we have noticed that a tit-for-tat strategy is not applied by the peers during the dissemination of video data. Finally, we have proposed to model the IP micro-flows of video content within a session in a first attempt by a renewal stream with indepen-

dently marked packet sizes of 1348 bytes whose logarithmically transformed inter-arrival times are governed by a log-logistic distribution.
We are convinced that new video portals like SopCast or Zattoo that offer the access to video streaming of shows or live TV channels constitute a promising new revenue path for next generation Internet. The embedding into Web services providing the rich features of future Web technology will enable a fast growing demand in the coming years and the related peer-to-peer traffic on the transport network of a convergent next generation network with its stationary and mobile clients will generate many challenges for performance engineering.

## References

[1] S. Ali, A. Mathur, and H. Zhang. Measurement of commercial peer-to-peer live video streaming. In *Proc. of ICST Workshop on Recent Advances in Peer-to-Peer Streaming*, Waterloo, Canada, 2006.

[2] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein. *Introduction to Algorithms*. MIT Press and McGraw-Hill, 2nd ed., 2001.

[3] D. Ciullo, M. Mellia, M. Meo, and E. Leonardi. Understanding P2P-TV systems through real measurements. In *Proc. GLOBECOM 2008*, IEEE Computer Society, 2297–2302, 2008.

[4] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross. Insights into PPLive: A measurement study of a large-scale p2p IPTV system. In *Proceedings IPTV Workshop, International World Wide Web Conference*, 2006.

[5] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross. A measurement study of a large-scale p2p IPTV system. *IEEE Transactions on Multimedia*, 9(8), 1672–1687, 2007.

[6] T. Hoßfeld and K. Leibnitz. A qualitative measurement survey of popular Internet-based IPTV systems. In *Second International Conference on Communications and Electronics (HUT-ICCE 2008)*, Hoi An, Vietnam, 2008.

[7] U.R. Krieger, R. Schweßinger. Analysis and Quality Assessment of Peer-to-Peer IPTV Systems. In *Proc. 12th Annual IEEE International Symposium on Consumer Electronics (ISCE2008)*, April, 14-16th 2008, Algarve, Portugal, 2008.

[8] F. Liu, Z. Li: A Measurement and Modeling Study of P2P IPTV Applications. In *Proceedings of the 2008 International Conference on Computational Intelligence and Security 1*, 114–119, 2008.

[9] C. Maccarthaigh. *Joost Network Architecture*. Technical Presentation, April 2007.

[10] N. Magharei, R. Rejaie, and Y. Guo. Mesh or multiple-tree: A comparative study of live p2p streaming approaches. In *Proc. INFOCOM 2007*, IEEE Computer Society, 1424–1432, 2007.

[11] J. Peltotalo, J. Harju, A. Jantunen, M. Saukko, and L. Väätämöinen. Peer-to-peer streaming technology survey. In *ICN '08 Proceedings of the Seventh International Conference on Networking*, IEEE Computer Society, Washington, USA, 342–350, 2008.

[12] A. Sentinelli, G. Marfia, M. Gerla, L. Kleinrock, and S. Tewari. Will IPTV ride the peer-to-peer stream? *Communications Magazine, IEEE*, 45(6), 86–92, 2007.

[13] T. Silverston, O. Fourmaux, A. Botta, A. Dainotti, A. Pescapé, G. Ventre, and K. Salamatian. Traffic analysis of peer-to-peer IPTV communities. *Computer Networks*, 53(4), 470–484, 2009.

[14] S. Tang, Y. Lu, J.M. Hernández, F.A. Kuipers, and P. Van Mieghem. Topology dynamics in a P2PTV network. In *Proc. Networking 2009*, 326–337, 2009.

[15] www.smoothit.org/Publications/Talks/hossfeld-HUT-ICE.pdf; last check on 12.07.2009.

[16] http://www.itu.int/ITU-T/IPTV/index.phtml

[17] http://www.colasoft.com/products/

[18] http://www.coolstreaming.us/hp.php?lang=en

[19] http://www.geoip.com; checked 15.07.2009

[20] http://www.maxmind.com; checked 15.07.2009

[21] http://www.wireshark.org/

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 213 – 222

# Remote Virtual Reality: Experimental Investigation of Progressive 3D Mesh Transmission in IP Networks

SŁAWOMIR NOWAK PRZEMYSŁAW GŁOMB

Institute of Theoretical and Applied Informatics
Polish Academy of Science
ul. Bałtycka 5, Gliwice, Poland
*{emanuel,przemg}@iitis.pl*

**Abstract:** This work presents selected early results of simulation experiments of 3D mesh progressive transmission in IP networks. The data transmission follows from a model of a Virtual Reality exploration process. The objective is to evaluate the possible impact of TCP/UDP transmission dynamics on the amount of data available in VR browsing client, which relates to browsing Quality of Experience. In four sections, the article presents a brief introduction, model description, experimental results, and conclusions.

**Keywords:** 3D mesh, progressive data representation, TCP transmission, Virtual Reality, OMNeT++.

## 1. Introduction

Virtual Reality (VR) techniques use computers to build, present, and allow for user interaction with simulated environment. Whether VR is based on real or artificial data, this form of interaction is reported as being in many ways similar to the exploration of natural environment [1]. It is therefore a natural medium for entertainment, education, data visualization, telepresence etc. The popularity of VR is growing steadily over the past years with improvement of hardware capabilities allowing for ever increasing user base and better Quality of Experience (QoE). The realization of VR service usually requires, beyond traditional multimedia data types, 3D object description, commonly being mesh of vertices [7]. This data type can occupy a substantial portion of the whole VR data, and its effective transmission is a complex problem [4].

From network perspective, the process of interaction in VR is quite different than video/audio transmission. While both share the necessity of adapting to network conditions (i.e. by using scalable streams [2] and rate distortion optimization), the interaction is dynamic in nature, as it is the user who decides

what part of the content is to be send over the network. This makes tasks such as traffic prediction, congestion control and buffering difficult.

As the quantity of 3D data is expected to grow, it is of importance to analyze the properties of its transmission. The key element for this is an accurate model of 3D data source. Unfortunately, due to many different used technologies and applications, formulation of such model is difficult. To the best of authors' knowledge, there is no universally accepted model of 3D data transmission yet.

We view the definition of VR exploration models and collecting experimental data an important step for such model construction. To this end, we propose a model of VR browsing, and present the results of early experiments investigating its behavior. Our VR model consists of a randomly placed collection of 3D objects (meshes). Each mesh is stored in progressive form [3], allowing for transmission of only part of the total data at a time, improving the quality of rendering presentation at client's end. The transmission of progressive 3D data [5,6] is a result of user moving about the VR – at regular time instants, his position changes, and, if needed, an update for each object is sent. The user motion is simulated by a random walk.

We focus on network aspect of the VR exploration. We observe the delay between time a position is updated, and the time all data required for updated position scene rendering is transmitted. Within this delay, the lack of data produces (on average) a drop of quality (below predefined limit). We investigate this behaviour in experiments with and without coexistent TCP/UDP traffic. The model was implemented using OMNeT++ [8] simulation framework with INET [9] TCP/IP model suite.

## 2. Model's architecture

The objective of our model is to simulate a simple remote VR exploration process, where position change results in sending update of 3D models. With each 3D object in VR stored in progressive form, the update size is relative to distance of user position to the models in VR.

The model uses a client-server architecture. The role of the server is to store 3D objects (meshes), keep a history of data transmitted to a client, and respond to client's request by sending data updates. The role of the client is to present (render) the 3D data, get user input, and transmit position change of the virtual observer to the server.

Each 3D mesh is stored in progressive form and has its own location (x,y) in the virtual world. The progressive form allows us to send an incomplete object (with only a small subset of vertices/triangles) and then update it when needed. The controlling parameter is the vertices/triangles density per display unit, as it directly relates to the perceived quality of object's rendering (an importan part of overall QoE). When user's position in the virtual world changes, so does the

distance to the object. If the distance decreases, the object will occupy a larger portion of client's rendering window. In that case, to sustain the quality, a update of object data must be send, to keep the displayed density on required level.

This scheme forces data transmission when needed, instead of requiring transmission of the whole world (including parts that may not be seen during exploration process). The basic algorithm of VR exploration is as follows:

1. Server is started, loads the objects and their positions;
2. Client is started, makes connection to the server, transmits initial position of the observer;
3. Server responds with initial data for object rendering resulting from initial position;
4. Client sends the observer's position update, resulting from observer's decision;
5. Server responds with update of object data (if needed);
6. If exploration is not finished, goto 4.

In the following simulation experiments, we incorporated the following browsing model in the client:

1. Produce the random position offset;
2. Delay a given number of milliseconds.

We are interested in the dynamic component of the browsing quality. While all of the requested data will be eventually successfully transmitted, at each time instant, there may be unfulfilled requests, resulting from data requested but not yet received. This quantity we denote by *hunger_factor*, and is observed in our experiments, as it relates to quality loss from the baseline at a given time.

The **`hunger_factor`** *hf* is expressed by (1):

$$hf = \sum_{i=1}^{n} \begin{cases} \dfrac{size_i}{dist_i^2} - bs_i, & (dist_i > md) \\ size_i - bs_i, & (dist_i \le md) \end{cases} \tag{1}$$

where:

$n$ – number of 3D objects within a VR scene;
$size_i$ – size of object $i$ in bytes;
$md$ – **`minimumDistance`** global parameter;
$dist_i$ – distance between $i$ objects and observer;
$bs_i$ – **`bytesSent`**, how many bytes of object $i$ was already sent to client and are stored in clients local database;

This model is obviously very simple, and is presented as a starting point for further investigation. In particular, we do not include more advanced (and human behavior based) browsing schemes and VR constructions; also, we ignore the information about which side user is facing. We feel that this simplification is justified, as it is difficult to point out the representative browsing behaviours or VRs, and in this model already several novel and complex network situations can be identified.

## 3. Experiments

The model and experiments were prepared in OMNeT++ discrete event-driven simulator. Its main advantages are simple process of simulation models creation, convenient interface and rich libraries. For TCP/IP simulation INET, an open-source package for the OMNeT++ was used, the simulation environment containing models for several Internet protocols: UDP, TCP, SCTP, IP, IPv6, Ethernet, PPP, IEEE 802.11, MPLS, OSPF, and others.

The TCP protocol is used for data transmission; a single TCP connection is sustained throughout the whole exploration time. The network topology is presented on Fig.1. The **`cli[0]`** and **`srv[0]`** host the client and the server of the VR exploration model; a single connection between them transmits 3D data. The **`cli[1]`** and **`srv[1]`** are connected with several "background" TCP and UDP transmissions. The hosts are connected to routers are connected with 100mbps links, the links between the routers are 10mbps (possible bottleneck).

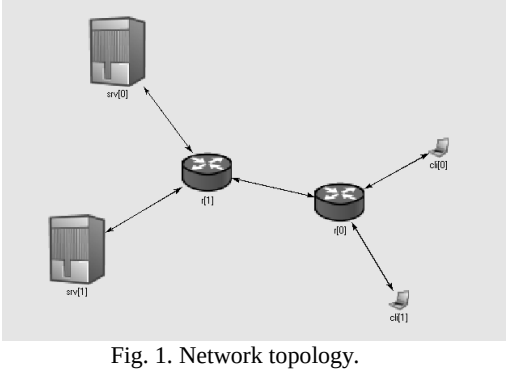



Fig. 1. Network topology.

To simplify the simulation evaluation we assume the **`minimumDistance`** parameter, the minimum distance below it the exact representation of a object have to send to the client. The exact representation (**`size`** parameter) of each object was 80 000[bytes]. As it is shown in the subsequent section the value of

**minimumDistance** significantly influences on the simulation results (esp. on the **hunger_factor** value).

In the simulation evaluation a single step of the observer in the VR is always 10[points] and new request for the data update was sent to the server every 1[sec].

We examined some different cases: "One object" case, Brownian motion and directed wandering for different values of the **minimumDistance** parameter and together with the concurrent TCP and UDP transmissions. Except the case of "One object" the simulations were carried out for 400 steps of the observer and **n**=50 objects uniformly arranged within the VR space (size 600 x 600 points). The summary results (selected **hunger_factor** statistics) are presented in Tab. 1.

### 3.1 "One object" case

A simplest case is where there is only one 3D object in the VR space, and the observer is moving closer to that object. There is no concurrent transmissions in the network. The resulting graph of **hunger_factor** (see Fig. 2) presents the generic idea of progressive 3D data, when the amount of data, needed to reconstruct the 3D scene is proportional to the squared distance between the object and the observer.

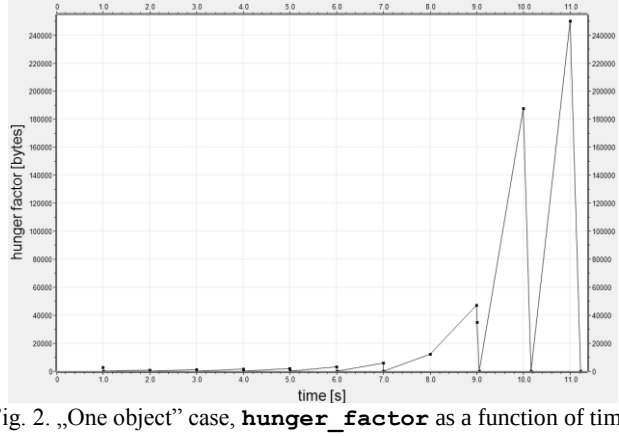



Fig. 2. „One object" case, **hunger_factor** as a function of time.

### 3.2 Case A and B, Brownian motion

Case A (**minimumDistance**=100) and B (**minimumDistance**=10) represents the example Brownian motion within the VR space (see figure 3).

There were concurrent transmission (one TCP and one UDP) but there was no congestion on the output queue in the bottleneck.

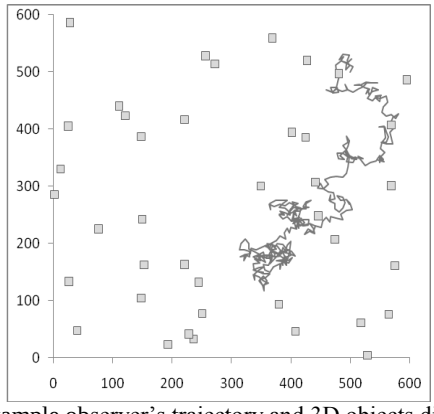
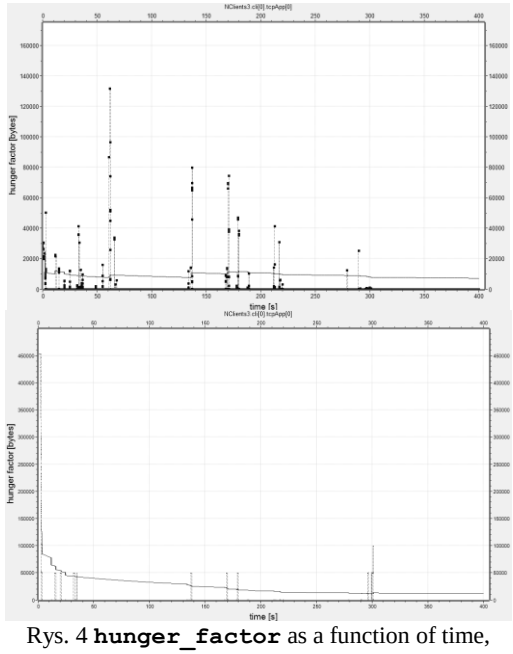



Fig. 3. Example observer's trajectory and 3D objects distribution for the Brownian motion (Case A and B).




Rys. 4 **`hunger_factor`** as a function of time,
left: case A (**`minimumDistance`**=100), right: case B (**`minimumDistance`**=10).

Because of the Brownian motion, the observer was moving within a limited, local area. At the beginning (before 200 step) the client gathered data in the local object's database and amount of sending data was significantly larger (Fig. 4). In

the case A objects are more detailed form the distance **`minimumDistance`**=100 and at times big "peaks" of **`hunger_factor`** are observed, when observer come to the close group of objects and as a result client requests detailed information of all of them. "Peaks" are significantly smaller in the case B, where **`minimumDistance`**=10.

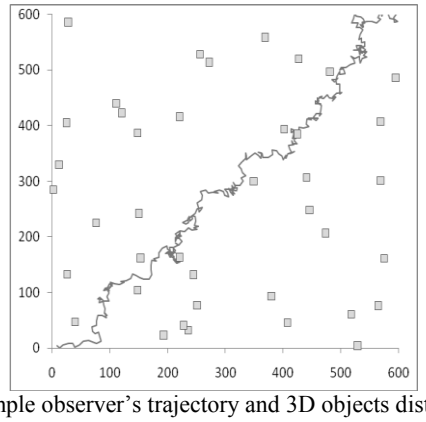



Fig. 5. Example observer's trajectory and 3D objects distribution for the directed wandering (Case C and D).

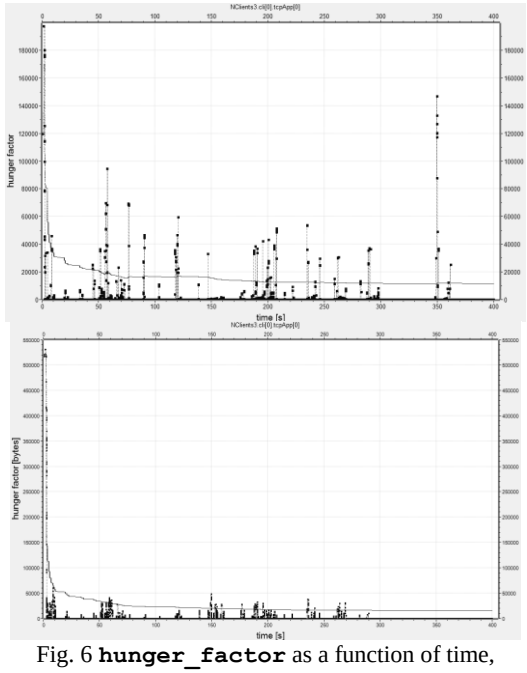



Fig. 6 **`hunger_factor`** as a function of time, left: case C (**`minimumDistance`**=100), right: case D (**`minimumDistance`**=10).

### 3.3 Case C and D, directed wandering

In cases C and D to Brownian motion we add directed vector [2,2], but the total length of a steep is still 10. The trajectory of the observer is shown on Fig. 5. Results for case C (**`minimumDistance`**=100) and case D (**`minimumDistance`**=10) is presented on Fig. 6.

### 3.4 Case E, congestion in the bottleneck

Previous cases assumed the convenient situation of lack of a heavy concurrent traffic. Occurrence of background transmission results of competing in bandwidth distribution in case of TCP or bandwidth limitation in case of UDP. As a result the quality of 3D scene reconstruction is decreased. The trajectory of the observer and object's distribution is the same as in cases C and D.

The configuration with the 100 TCP transmission were evaluated (file sending, random start). In case E the established sending window was not large enough to ensure the high quality of scene reconstruction. The **`hunger_factor`** parameter remained in a high level during simulation. We can conclude that in real application the process of scene reconstruction will be delayed and cannot keep up with position changes. Additionally server will be often send out-of-date data, according to previous position of the observer.

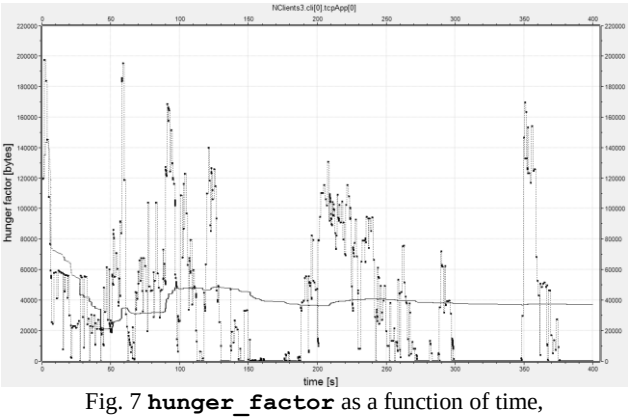



Fig. 7 **`hunger_factor`** as a function of time,
case E (**`maximumDistance`**=100).

| | hunger_factor[bytes] | | |
|---|---|---|---|
| | Max | Mean | Std Dev. |
| case A (minimumDistance=10) | 131 633 | 7 107 | 15 523 |
| case B (minimumDistance=100) | 453 030 | 11 531 | 49 820 |
| case C (minimumDistance=10) | 197 606 | 11 413 | 25 680 |
| case D (minimumDistance=100) | 531 095 | 14 980 | 49 904 |
| case E (minimumDistance=10) | 197 606 | 36 827 | 42 884 |

Tab. 1. Simulation results

## 4. Conclusions

This article presents early results of simulation experiments of 3D mesh progressive transmission. We use a client-server model of VR browsing. We've investigated several different cases of VR exploration, observing the "hunger for data" measure.

We observe that the nature of 3D traffic source is complex, and that advanced models will be needed for performance analysis experiments. In particular, irregular bursts of data were observed, and there is a notable "cost" (data that needs to be send) in the moment when user starts the exploration. Quality of Experience will be strongly affected by the Quality of Service mechanisms used, this problem requires further study; possibly advanced client with prediction of user movement would help here. Also, work needs to be done on defining objective measures for quality of VR exploration, both for 3D data presentation (especially incomplete or reduced quality) and interaction. Work on those problems will be continued.

## Acknowledgements

This work was partially supported with Polish Ministry of Science research project NN516405137.

# References

[1] Williams B., Narasimham G., Westerman C, Rieser J, Bodenheimer B., *Functional Similarities in Spatial Representations Between Real and Virtual Environments*, ACM Transactions on Applied Perception, Vol. 4, No. 2, Article 12, (2007).

[2] Głomb P, Nowak S., *Image coding with contourlet / Wavelet transforms and spiht* algorithm:an experimental study, proc. of IMAGAPP 2009, International Conference on Imaging Theory and Applications, Lisboa, Portugal, (2009).

[3] Skabek, K., Ząbik, L.:Implementation of Progressive Meshes for Hierarchical Representation of Cultural Artifacts, Communications in Computer and Information Science (2009).

[4] Li H., Li M., Prabhakaran B., *Middleware for Streaming 3D Progressive Meshes over Lossy Networks*, ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 2, No. 4, Pages 282-317, (2006).

[5] Kurose J. F., Ross K. W., *Computer Networking: A Top-Down Approach Featuring the Internet*, Addison Wesley, (2005).

[6] Hoppe, H.: *Progressive meshes*. ComputerGraphics (SIGGRAPH'96 Proceedings) 99-108. (1996).

[7] Nielsen, F.: *Visual computing: Geometry, graphics and vision*. Charles River Media (2005)

[8] OMNet++ homepage. http://www.omnetpp.org.

[9] INET homepage. http://inet.omnetpp.org.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 223–232

# Performance evaluation of a multiuser interactive networking system

ADAM JÓZEFIOK [a] KRZYSZTOF GROCHLA [b]
TADEUSZ CZACHÓRSKI[b]

[a]Silesian University of Technology, Institute of Informatics, PHD studies
ul. Akademicka 16, 44-100 Gliwice, Poland
*adam.jozefiok@list.pl*

[b]Institute of Theoretical and Applied Informatics
Polish Academy of Science
ul. Bałtycka 5, Gliwice, Poland
*{kgrochla,tadek}@iitis.pl*

**Abstract:** The article presents a work in progress aiming at performance evaluation of a large database system at an assurance company. The system includes a server with a database and a local area network with a number of terminals where the company employees run applications that introduce documents or retrieve them from the database. We discuss a model of clients activities. Measurements were collected inside the working system: the phases at each users application were identified and their duration was measured. The collected data are used to construct a synthetic model of applications activities which is then applied to predict the system behaviour in case of the growth of the number of users.

## 1. Introduction

The performance of large computer systems, serving thousands of users on daily basis is crucial to many company activities. Network speed limitation and computing power of the servers may limit the response speed, slowing down the work of many employees and introducing large costs. The evaluation of such a system should find a performance bottleneck, to localize and show the administrators the parts of the system (e.g. networking devices or servers) which should be replaced or improved. In this work we try to analyze a very large networking system, providing access to database servers in one of the largest polish insurance companies. The goal of the work is to identify the maximum number of users that

the system in current stage may serve and to identify what part (the server or the networking environment) should be improved to speed up the user service time.

The IT system being analyzed includes a multiple servers with a database and a local area network with a number of terminals where the company employees run applications that introduce documents or retrieve them from the database. Each server provides access to separate application, to simplify the model we consider just one of them. We discuss a model of clients activities. Measurements were collected inside the working system: the phases at each users application were identified and their duration was measured. The collected data are used to construct a synthetic model of applications activities which is then applied to predict the system behaviour in case of the growth of the number of users.

The investigated computer network is built on the Microsoft Windows software and called the Complex IT System (CITS). Each workstation at this network is called here a gate of CITS. The Complex IT System is a modern and innovative technological solution and considered as one of the greatest systems in the world. In Poland, the system performance about 100 million of accountancy operations annually and about 40 billion calculations in numerous specially created data centres.

At the considered installation there are 30 Cisco switches linked with a multimode fibre. Moreover, ain the local network there is connected about 10 servers for various purposes related to the maintenance of the whole environment. The main servers are responsible for data processing; each of them is responsible for the activity of one kind of application, but it often occurs that all servers supervise the activity of an individual application. The workstations working at the local area network communicate with the access switches arranged on each floor and these ones are linked to the core switches. All the core switches communicate with the central router. The router is joined via the Wide Area Network with the main router at the headquarters in Warsaw.

## 2. The activities of interactive applications

An interactive application is a program that enables the modification of data at the central database or the Data Warehouse. The interactive application is a client of the data base. The diagram of interactive applications activities is shown in Fig.1.

In the first phase a user chooses some data which he wants to find. A client connects with the server of database using the TCP protocol and a defined port, sending him a request. After a certain period of time, the server sends the information which is claimed by a client. A client can change the obtained data. In dependence on a kind of the application, the changes can require approval of an

application supervisor. Before starting work, a user is required to provide a suitable role he is going to work with. The role may be, for example: an approbant, a manager or a user.

The role in the application is granted at the request of the chief of the Department and verified by the Information Security Department, then it goes to the IT department where it is implemented. It should be noted that each employee working in a company and user of a computer network can log onto the system soleby using a special smart card.

At first login the interactive application begins. A user has to give the role in which he is going to work with. The logon process to the applications starts. Depending on the server load it may be shorter or longer.

In the next stage a user has to consider what kind of a payer he wants to download to the context. Choosing a payer is necessary, because all subsequent operations can be performed only after this choice. To select a payer, a user has to put his NIP number (Tax Identification Number), in the appropriate box or other assigned identifier, depending on legal status.

After that, the information is sent to the database and its resources are searched by the server. If a payer is found, the interactive application goes to its context, otherwise a user see the error message suggesting an incorrect application identifier. However, in most cases, a user has to give additional information to verify a downloaded payee. To download a particular document it is necessary to choose it from the drop-down list of the application. After clicking on the document it is downloaded from the database and then opened. If the user works on the context of a payer, who posses a large number of sent documents, there is a possibility to download the documents from the last several years at once. It is sufficient to select the appropriate boxes.

At the time between downloading the documents, a user analyses them and depending on the complexity of a researched case can take new ones. After the completion of the case a user closes all the analysed documents. If a case is fully completed, he also close a previously selected payer to the context.

At this stage, a user can select another payer to the context and start a new analysis of other documents, or can log out from the application and finish work on the computer. A logout application is an earlier closure of all documents, and a payer and then clicking the Logout button. After that a client becomes disconnected from the database.

## 3. The methods of collecting data to simulations

To enable the conduct of subsequent simulations, the project of the researched network (Figure 1) was setup. The project includes all activities occur from time the moment of logging on to the application of completing the work. According to the draft collecting of necessary data was started.

In the first step the time of logging on to applications for individual users was measured. During the research, the shortest time of logging lasted 0 seconds, and the longest 780 seconds, this is why in this range of time subsequent calculations were carried out.

In the next step calculations of probabilities were solved from the received data the histogram of login time was made presented the empirical distribution of the characteristic. In the first phase of the research, the histogram showed the characteristics of an exponential distribution. For the verification, the following hypotheses were assumed.

On the base of Kolmogorov - Smirnov statistics and the $\chi^2$ (chi square) statistic [2] the hypothesis of compatibility of distributions with exponential distrubution were rejected. The value p-value was less than the 0.05 level of significance. However we took it as a first approximation as the input to the simulation program written in OMNET++ [1].

A similar draft of activity was used for the rest of activities in the project:

- a number of active payers per day;
- the time between downloading a payer and a document;
- the gap between downloading documents;
- download time of a single document;
- the number of documents downloads for a single payer;
- service time of one payer;
- download time of one payer;
- log off time;

Figs. 2 – 11 present results of measurements.

## 4. Simulation model

The model of the server done with the use of OMNeT++ simulation system [1] is presented in Fig.12. In the simulation the 6-processor server was considered.

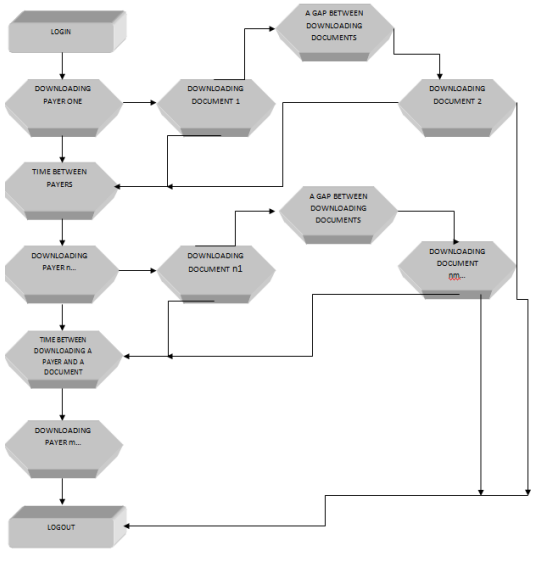



Fig. 1. Diagram of an application activities

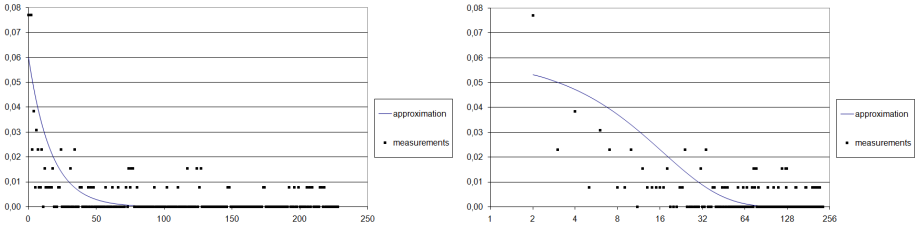



Fig. 2. Distribution of logon time –linear and logarithmic scale

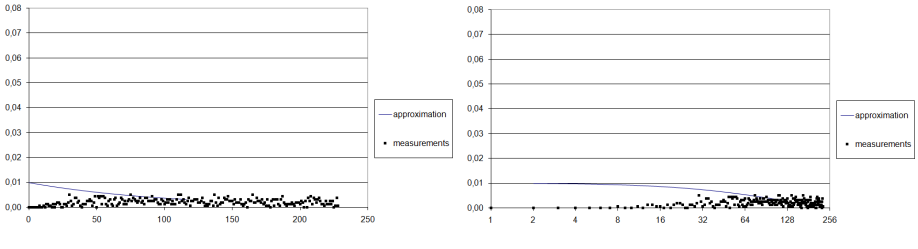



Fig. 3. Distribution of one payer service time – linear and logarithmic scale

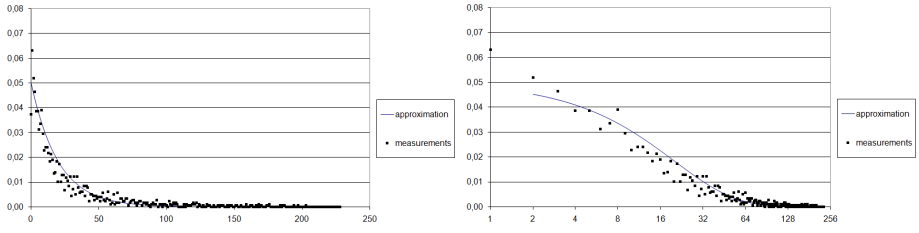



Fig. 4. Distribution of the download time of a single payer – linear and logarithmic scale

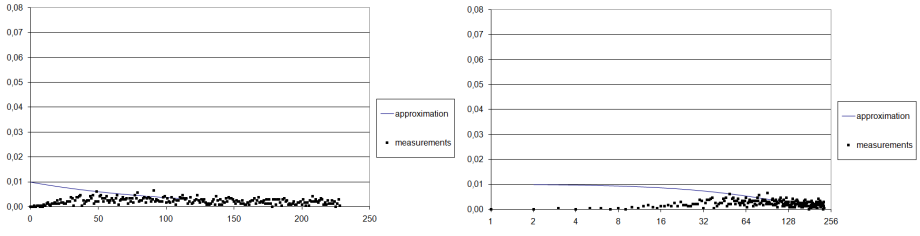



Fig. 5. The time between downloading payers Ű linear and logarithmic scale

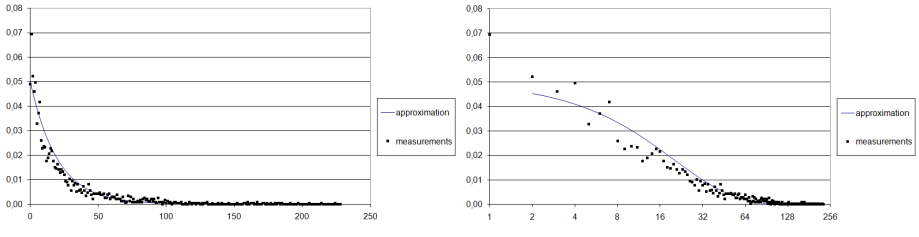



Fig. 6. Distribution of the time between downloading a payer and a document – linear and logarithmic scale

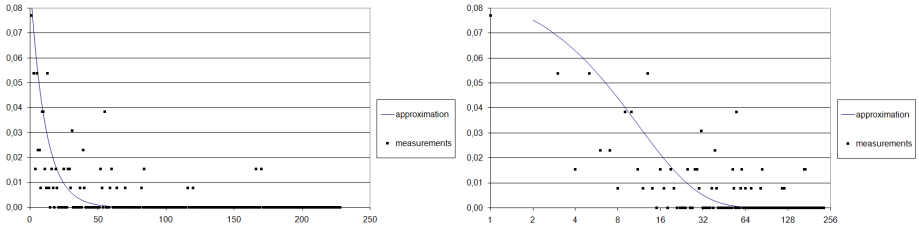



Fig. 7. Distribution of log off time – linear and logarithmic scale

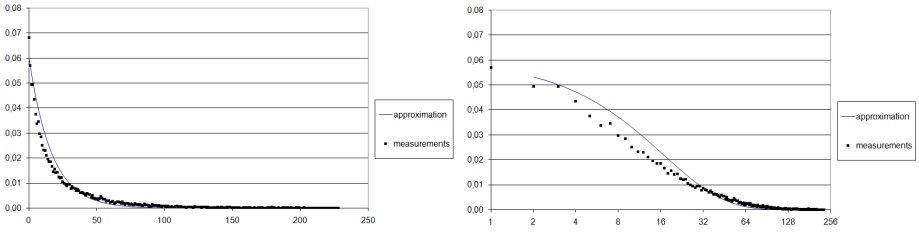



Fig. 8. Distribution of download time of documents – linear and logarithmic scale

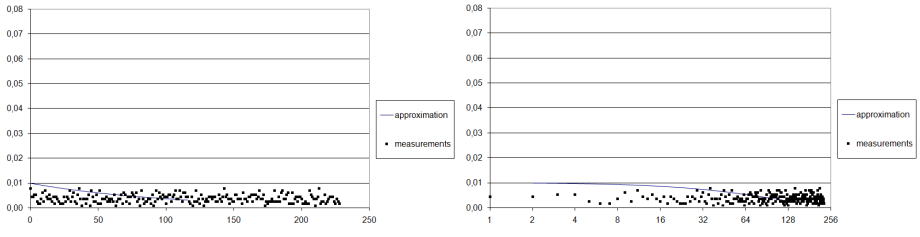



Fig. 9. The number of payers downloaded by a user during one day – linear and logarithmic scale

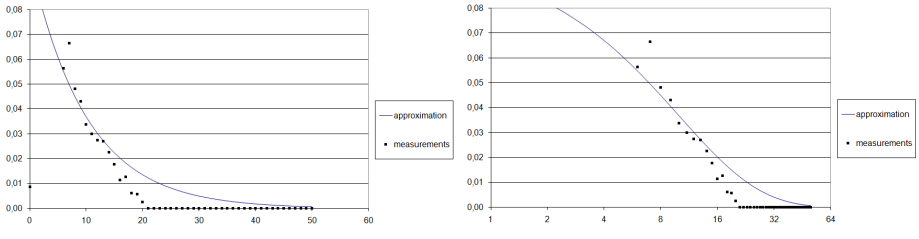



Fig. 10. Distribution of the number of document downloads for a single payer – linear and logarithmic scale

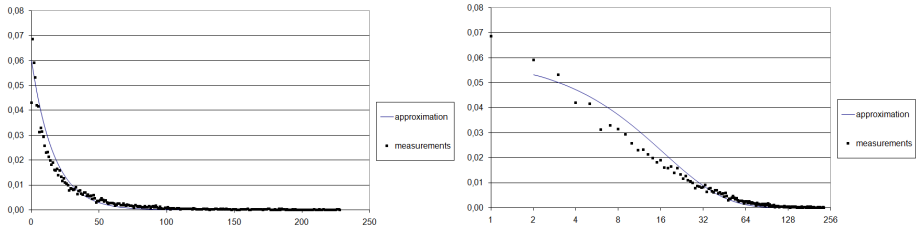



Fig. 11. Distribution of the gap length between downloading documents – linear and logarithmic scale

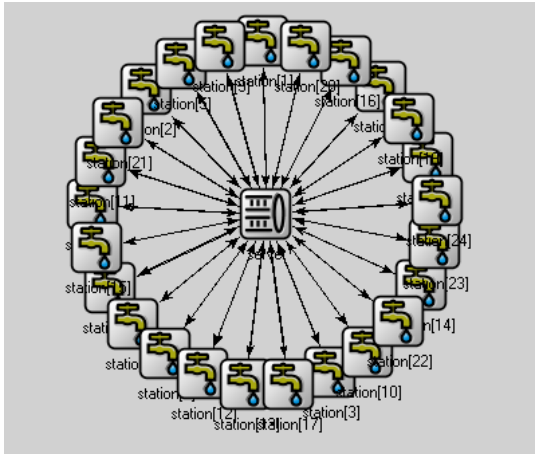



Fig. 12. Simulation model of the network made with the use of OMNET++

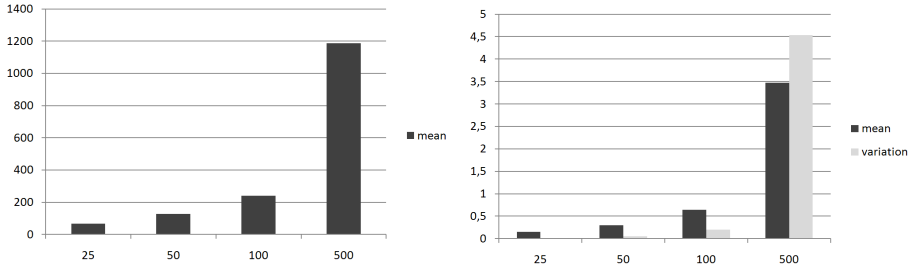



Fig. 13: Left: mean total time for retrieval of all documents related to one payer as a function of the number of active stations, right: mean and variation of one document retrieval time as a function of the number of active stations

Each of the processors provided access to the documents and served the clients according to the measured average service times, with exponential distribution.

Some simulation results are given below. Fig.13 presents mean total time for retrieval of all documents related to one payer as a function of the number of active stations as well as the mean and variation of one document retrieval time as a function of the number of active stations. Fig. 14 gives the distribution of a document retrieval time for a network of size 25 and 50 stations (left figure), and of 100, 500 stations (right figure ). The simulation shows, that with the current server efficiency the system may sustain up to 100 users.

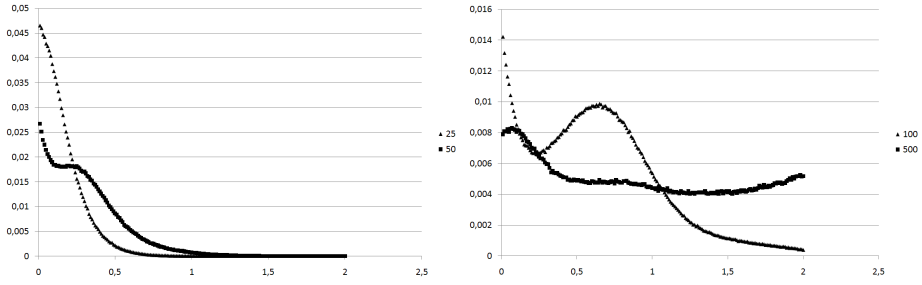



Fig. 14: Distribution of a document retrieval time for a network of size: 25 and 50 stations (left), 100, 500 stations (right)

## 5. Conclusions

The article reports our attempts to capture the characteristics of client activities in a large computer data base system and to construct a queueing model to predict the system performance while the number of customers is growing. The results are based on a simulation model, in the nearest future but we think also to use Markov models to evaluate the performance of the CITS, and to compare the outpot of models with synthetic times with the output of simulation model using real time histograms collected in the system.

## 6. Acknowledgements

This research was partially financed by the Polish Ministry of Science and Education grant N517 025 31/2997.

## References

[1] OMNET ++ site: http://www.omnetpp.org/

[2] Morris H. DeGroot, Mark J. Schervish *Probability and Statistics*, third edition, Addison Wesley, Boston 2002.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 233–246

# Analytical modeling of the multicast connections in mobile networks

DAMIAN PARNIEWICZ[a] MACIEJ STASIAK[a]
PIOTR ZWIERZYKOWSKI [a]

[a]Poznan University of Technology, Chair of Communications and Computer Networks
ul. Polanka 3, 60-965 Poznan, Poland, *piotr.zwierzykowski@put.poznan.pl*

**Abstract:** In this paper a new analytical method of blocking probability calculation in multi-service cellular systems with multicast connections is proposed. The basis for the discussed method is the full-availability group with traffic compression and the fixed-point methodology. The results of the analytical calculations were compared with the results of the simulation experiments, which confirmed the accuracy of the proposed method. The proposed scheme can be applicable for a cost-effective resource management in the UMTS/HSPA/LTE mobile networks and can be easily applied to network capacity calculations.

**Keywords** Analytical model, Multicast, UMTS, Iub interface

## 1. Introduction

With the rapid development and popularity of multimedia services in mobile networks of the second and the third generations, there has been a distinct increase in the interest in methods for dimensioning and optimization of networks carrying multi-rate traffic.

The problem of optimization and proper dimensioning of cellular networks is particularly important within the context of the ever-gaining popularity of multimedia services that use the MBMS standard (Multimedia Broadcast and Multicast Service) [1]. This standard envisages a possibility of setting up multicast connections for particular multimedia services. The application of the multicast mechanism in cellular networks constitutes, however, an additional challenge to be taken up by operators of cellular networks and networks designers.

The starting point for operators of cellular networks is to define a set of the KPI (Key Performance Indicator) parameters based on SLA (Service Level Agreement), which can be used as input data in the process of dimensioning and optimization of

networks. On the basis of the KPI parameters, it is possible to determine, for example, the blocking probability or the average throughput, which can be successively used in the evaluation of the Grade of Service (GoS).

The dimensioning process for the 3rd generation Universal Mobile Telecommunications System (UMTS) should make it possible to determine such a capacity of individual elements of the system that will secure, with the assumed load of the system, a pre-defined GoS level. With dimensioning the UMTS system, the most characteristic constraints are: the radio interface and the Iub interface. When the radio interface is a constraint, then, in order to increase the capacity, the access technology should be changed or subsequent branches of the system should be added (another NodeB). If, however, the constraint on the capacity of the system results from the capacity of the Iub interface, then a decision to add other nodes can be financially unfounded, having its roots in incomplete or incorrect analysis of the system. This means that in any analysis of the system, a model that corresponds to the Iub interface should be also included.

Several papers have been devoted to traffic modeling in cellular systems with the WCDMA radio interface, e.g. [2–8]. The relevant literature proposes only two analytical models of the Iub interface [9, 10]. In [9] the authors discuss the influence of the static and dynamic organization schemes of Iub on the efficiency of the interface. The static scheme assumed that the Iub interface is divided into two links: the first link carries a mixture of R99 (Release99) [11] traffic classes and the other link services HSDPA (High Speed Downlink Packet Access) [12–14]

traffic streams. In the dynamic organization scheme, it was assumed that the Iub interface resources are limited for R99 traffic while the resources for HSDPA traffic are unlimited. In all models, the influence of the compression mechanism for HSPA (High Speed Packet Access) traffic classes was not taken into consideration and the average throughput per a HSPA user was not discussed. In [10], an effective analytical model of the Iub interface carrying a mixture of R99 and HSPA traffic classes with adopted compression functionality was proposed. In the paper we will consider this model as a basis for modeling of HSPA traffic carried by Iub interfaces.

This paper discusses multicast mechanism implemented in the UMTS network on the basis of the Iub interface. The analytical model of the cellular system proposed in the paper is, to the best belief of the authors, the first model that takes into consideration the influence of multicast connections on the traffic effectiveness in a cellular network servicing multi-rate traffic.

The paper is divided into four sections: section 2 presents the analytical model used for modeling multicast connections in the UMTS network, section 3 includes the exemplary numerical results and the following section summarizes the consid-

erations presented in the paper.

## 2. Model of the system

Let us consider the structure of the UMTS network presented in Fig. 1. The presented network consists of three functional blocks designated respectively: User Equipment (UE), UMTS Terrestrial Radio Access Network (UTRAN) and Core Network (CN). The following notation has been adopted in Fig. 1: RNC is the Radio Network Controller, WCDMA is the radio interface and Iub is the interface connecting Node B and RNC.

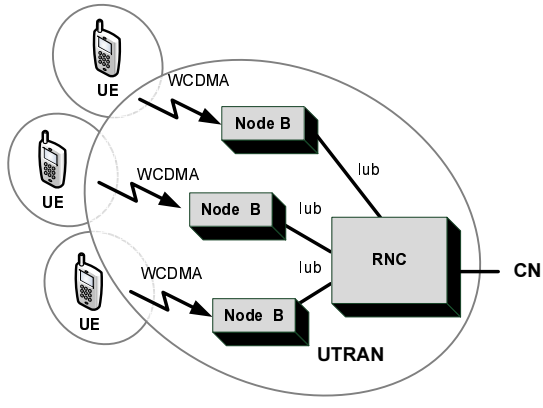



Fig. 1. Elements of the UMTS network structure.

### 2.1. Model of the system with multicast connections

Let us consider now the UMTS network servicing multicast connections. In the multicast connection one user is connected with a predefined group of users. It is relatively easy to define a number of varied scenarios of setting up multicast connections in the UMTS network with the radio interface and the Iub interface. Let us consider the two following exemplary scenarios:

- In the first scenario we assume that all users of a multicast connection are in the same cell. It means that to set up a connection, only the resources of one radio interface and one Iub interface are involved. Thus, a multiplication of connections is effected at the radio interface level, so carrying of multicast traffic has to be taken into consideration in the model of the radio interface.
- The second scenario assumes that the users of a multicast connection are in different cells. This means that radio resources of each of the cells service

one connection, whereas network resources between RNC and appropriate stations of NodeB are under heavy load of many connections[1]. Accordingly, in this scenario, so carrying of multicast traffic should be taken into consideration at the level of the Iub interface.

The multicast connection set up effected in one radio interface, proposed in the first scenario, can be taken into consideration through a multiplication of the number of traffic classes or through an increase in traffic intensity of appropriate traffic classes offered to the interface [1]. For analytical modeling of the first scenario, we can use one of the methods proposed in the literature of the subject, e.g. [2, 3, 6, 7]. Therefore, further in our considerations, we will not discuss the analytical model corresponding to this scenario.

In this paper we consider the second scenario, i.e. one in which we assume that each of the subscribers is associated with a different NodeB. This scenario requires then setting up several independent connections based on Iub interfaces. Let us assume that the Iub interface services a mixture of a number of different R99 and HSDPA traffic classes. Our further assumption is that the number of subscribers that are concurrently available is decisive whether blocking of a given connection ensues or not. This means that to model multicast connections in the discussed scenario, two analytical models, described in Section 2.2. and Section 2.3., can be worked out.

### 2.2. Analytical model of the system with multicast connections

We assume in this model that a multicast connection is lost if no connection with at least one of the demanded receivers can be set up. In line with the adopted definition, blocking in the multicast connection occurring in one link (between RNC and NodeB) influences the service acceptance of remaining connections that form the discussed multicast connection. In this case, service processes of multicast calls in individual links of a given multicast connection are interdependent.
The blocking probability of multicast calls of class $i$ in the link $r$ can be expressed as a function of the intensities of all traffic classes $M$ offered to the link $r$:

$$B_{i,r} = F[(a_{1,r}, t_1), \ldots, (a^e_{i,r}, t_i), \ldots, (a_{M,r}, t_M)], \tag{1}$$

where $a_{1,r}$ denotes the intensity of traffic of class 1 offered to the link $r$, whereas $a^e_{i,r}$ is the intensity of this part of the multicast traffic of class $i$ that is serviced in the link $r$ (this traffic is called effective traffic).
In order to determine the function (1) it is necessary then to determine effective traffic ($a^e_{i,r}$), i.e. this part of the total multicast traffic of class $i$ that is serviced in

[1]Each multicast call needs to be set up in many Iub links concurrently.

the direction $r$. Traffic $a_{i,r}^{e}$ forms such a part of the total multicast traffic of class $i$ (denoted by the symbol $a_i$)), which is not blocked in the remaining links participating in this connection (hence, it can be offered to the direction $r$). This dependence - in line with the fixed-point methodology [15]- can be written as follows:

$$a_{i,r}^{e} = a_i \prod_{z=1, z\neq r}^{L_i} (1 - B_{i,z}), \tag{2}$$

where $L_i$ is the number of links (resources of Iub interfaces) participating in the multicast connection of class $i$.
Function (1) can be determined on the basis of the modified Kaufman-Roberts distribution presented in the section devoted to the analytical modeling of the Iub interface (Section 2.4.).

Knowing the blocking probability of multicast calls of class $i$ in each of the participating links in this connection, we are in position to determine the total blocking probability of multicast calls of class $i$ in the system. The procedure for a determination of this probability depends on the adopted definition of blocking. In the model under consideration we assume that blocking of connection occurs if there are no free resources required for the service of a given connection in at least one of the demanded links. This dependence can be written as follows:

$$B_i = 1 - \prod_{z=1}^{L_i} (1 - B_{i,z}), \tag{3}$$

where $B_i$ is the blocking probability of multicast calls of class $i$ in the system.
To sum up, the iterative algorithm of determination of the blocking probability $B_i$ can be written in the form of the Algorithm 1.

### 2.3. Analytical model of the system with *k*-cast connections

Let us assume now that it is the number of accessible subscribers (i.e. those that can be concurrently reached) that is decisive in whether or not blocking of a given connection occurs. Our further assumption is that whether blocking of a given multicast connection of class $i$ occurs or not depends on the number of subscribers that can be concurrently reached. Multicast connections, for which the same definition of blocking has been adopted, will be called k-cast connections.

The identified exemplary situations in which blocking depends on whether there is a possibility of setting up a connection with three out of five demanded subscribers is presented in Figure 2. According to the adopted definition, k-cast connection ("3" out of 5") can be effected when at least three subscribers can be

**Algorithm 1** Iterative algorithm for blocking probabilities determination of multicast connections.

1. Setting the iteration number $l = 0$.
2. Determination of initial values: $\forall_{1 \le i \le M} \forall_{z \in L_i} B_{i,z} = 0$.
3. Determination of the values of the effective traffic $a^e_{i,r}$ on the basis of Eq. (2).
4. Increase in the iteration number: $l = l + 1$.
5. Determination of the values of blocking probabilities $B_{i,z}$ based on Eq. (10).
6. Determination of the values of blocking probabilities $B_i$ on the basis of Eq. (3).
7. Repetition of steps No. 3–6 until the assumed accuracy of the iterative process is obtained:

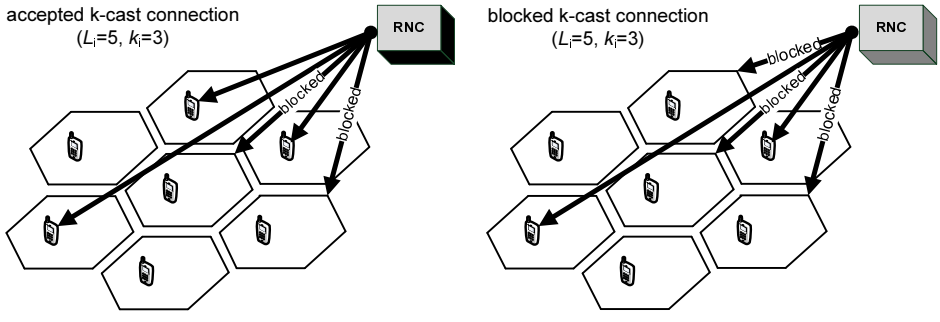
$$\forall_{1 \le i \le M} \left( \left| \frac{B_i^{(l)} - B_i^{(l+1)}}{B_i^{(l+1)}} \right| \le \xi \right),$$

where $B_i^{(l)}$ and $B_i^{(l+1)}$ are the appropriate values of blocking probabilities, obtained in iteration $l$ and $l + 1$, respectively.

reached. Figure 2a shows an exemplary instance when the resources required to set up a k-cast connection are in the three, from among five, demanded links. Thus, the connection will be successfully effected. If, however, the number of links that can service the k-cast connection ("3" from 5") of a given class is reduced to 2 (Figure 2b), then the admission control (located in, for example, RNC) will not allow the new call to be serviced. Let us assume now that the blocking probability of each of the component connections, i.e. included in a given k-cast connection, is the same and equals $B_i^*$. For example, to determine the blocking probability $k$ from among $L_i$ components of the k-cast connection of class $i$ – $P_B(k/L_i)$ – we apply the Bernoulli distribution:

$$P_B(k/L_i) = \binom{k}{L_i} (B_i^*)^k (1 - B_i^*)^{L_i - k}. \tag{4}$$

To determine the blocking probability of the considered k-cast connection, one has to consider all instances in which this connection will be blocked. In line with the adopted definition, a k-cast connection (in our particular example "3" from among 5") will be rejected if there is no possibility of setting up any, randomly chosen, three, or four, or all five required connections included in the given




(a) (b)

Fig. 2. Example of the accepted (a) and blocked (b) k-cast connection.

k-cast connection. Therefore, the total blocking probability $B_i^{k-cast}$ of a k-cast connection of class $i$ can be, in the considered case, written in the following way:

$$\begin{aligned} B_i^{k-cast} &= P_B(3/5) + P_B(4/5) + P_B(5/5) \\ &= \binom{5}{3}(B_i^*)^3(1-B_i^*)^2 + \binom{5}{4}(B_i^*)^4(1-B_i^*) + \binom{5}{5}(B_i^*)^5. \end{aligned} \quad (5)$$

In the general case we adopt that the maximum number of demanded directions in the k-cast connection of class $i$ is $L_i$ and that the minimum number of required connections is denoted by $k_i$. The k-cast class $i$ connection is blocked when there is no possibility of effecting $L_i - k_i + 1, \ldots, L_i$ connections. The blocking probability of a k-cast call of class $i$ can be determined on the basis of Eq. 4 in the following way:

$$B_i^{k-cast} = \sum_{z=L_i-k_i+1}^{L_i} \binom{L_i}{z}(B_i^*)^z(1-B_i^*)^{(L_i-z)}. \quad (6)$$

The blocking probability $B_i^*$ can be found as a blocking probability of a single connection within the multicast connection of class $i$. It can be done knowing that the multicast connection of class $i$ is non-blocked only if all $L_i$ related unicast connections are non-blocked (Eq. 3) :

$$B_i = 1 - (1 - B_i^*)^{L_i}, \quad (7)$$

where $B_i$ is the blocking probability of the mulicast call of class $i$ calculated on the basis of the Algorithm 1.

After simple modification, we can extract the value of $B_i^*$ by approximating real values of the blocking probability in a single link:

$$B_i^* = 1 - \sqrt[L_i]{1 - B_i}. \tag{8}$$

Recapitulating, it can clearly be seen that in order to determine the blocking probability of multicast calls and k-cast connections serviced in the cellular network, it is necessary to determine the blocking probability of each traffic class in each participating link in the connection. In line with the discussed scenario of connections, this value will be determined on the basis of the analytical model of the Iub interface.

### 2.4. Analytical model of the link (Iub interface)

The Iub interface in the UMTS network can be treated as a full-availability group (FAG) carrying a mixture of multi-rate traffic streams with and without compression property. Let us assume that the total capacity of the FAG is equal to $V$ Basic Bandwidth Units (BBUs). The group is offered $M = M_k + M_{nk}$ independent classes of Erlang traffic classes: $M_k$ classes whose calls can change requirements while being serviced and $M_{nk}$ classes that do not change their demands in their service time. It was assumed that class $i$ traffic stream is characterized by the Erlang traffic streams with the intensity $A_i$. The demanded resources in the group for servicing particular classes can be treated as a call demanding an integer number of (BBUs) [7]. The value of BBU, i.e. $R_{BBU}$, is calculated as the greatest common divisor of all resources demanded by traffic classes offered to the system. The occupancy distribution can be expressed by the modified Kaufman-Roberts recursion presented in [10]:

$$n\left[P_n\right]_V = \sum_{i=1}^{M_{nk}} A_i t_i \left[P_{n-t_i}\right]_V + \sum_{j=1}^{M_k} A_j t_{j,\min} \left[P_{n-t_{j,\min}}\right]_V, \tag{9}$$

where $\left[P_n\right]_V$ is the probability of state $n$ BBUs being busy, $t_i$ is the number of BBUs required by a class $i$ call: $t_i = \lfloor R_i / R_{BBU} \rfloor$, and ($R_i$ is the amount of resources demanded by class $i$ call in $kbps$) and $t_{j,\min} = \lfloor R_{j,\min} / R_{BBU} \rfloor$ is the minimum number of BBUs required by class $j$ call in the condition of the maximum compression. Formula (9) describes the system in the condition of maximum compression. Such an approach is indispensable to determine blocking probabili-

ties $B_i$ for a class $i$ call in the system with compression:

$$B_i = \begin{cases} \sum\limits_{i=V-t_i+1}^{V} [P_n]_V & \text{for} \quad i \in \mathbb{M}_{nk}, \\ \sum\limits_{i=V-t_{i,\min}+1}^{V} [P_n]_V & \text{for} \quad i \in \mathbb{M}_k, \end{cases} \tag{10}$$

where $V$ is the total capacity of the group and is expressed in BBUs ($V = \lfloor \frac{V_{Iub}}{R_{BBU}} \rfloor$, where $V_{Iub}$ is the physical capacity of group in kbps).

The average number of class $i$ calls serviced in state $n$ can be determined on the basis of Formula [16, 17]:

$$y_i(n) = \begin{cases} A_i \left[P_{n-t_i}\right]_V / \left[P_n\right]_V & \text{for } n \leq V, \\ 0 & \text{for } n > V. \end{cases} \tag{11}$$

Let as assume now that the full-availability group services a mixture of different multi-rate traffic streams with compression property [10]. This means that, in the traffic mixture, there are such calls in which a change in demands caused by the overload system follows unevenly.

The measure of a possible change of requirements is *maximum compression coefficient* that determines the ratio of the maximum demands to minimum demands for a given traffic class: $K_{j,\max} = \frac{t_{j,\max}}{t_{j,\min}}$, where $t_{j,\max}$ and $t_{j,\min}$ denote respectively the maximum and the minimum number of basic bandwidth units (BBUs) demanded by a call of class $j$. In the model we assume that all classes can undergo compression to a different degree.

We assume that the real system operates in such a way as to guarantee the maximum usage of the resources, i.e. a call of compressed class always tends to occupy free resources and decreases its maximum demands in the least possible way. The measure of the degree of compression in state $n$ is the quotient $\frac{V-Y^{nk}(n)}{n-Y^{nk}(n)}$, where $V - Y^{nk}(n)$ expresses the amount of resources available for calls with compression and $n - Y^{nk}(n)$ is the number of BBUs occupied by calls with compression (Fig. 3).

Let us consider now the average number of busy BBUs in the system occupied by class $j$ calls with compression:

$$Y_j^k = \sum_{n=0}^{V} y_j(n) \left[\xi_{k,j}(n) t_{j,\min}\right] \left[P_n\right]_V, \tag{12}$$

where $\xi_{k,j}(n)$ is the compression coefficient in the model:

$$\xi_{k,j}(n) = \begin{cases} K_{j,\max} & \text{for} \quad \frac{V-Y^{nk}(n)}{n-Y^{nk}(n)} \geq K_{j,\max}, \\ \frac{V-Y^{nk}(n)}{n-Y^{nk}(n)} & \text{for} \quad 1 \leq \frac{V-Y^{nk}(n)}{n-Y^{nk}(n)} < K_{j,\max}. \end{cases} \tag{13}$$

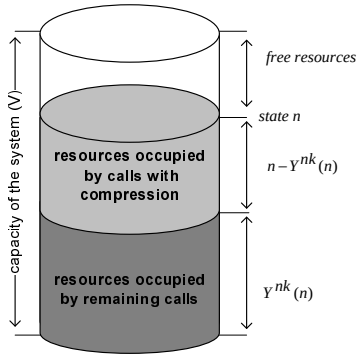



Fig. 3. Example of the system with compression.

In Formula (13), the parameter $Y^{nk}(n)$ is the average number of busy BBUs in state $n$ occupied by calls without compression:

$$Y^{nk}(n) = \sum_{i=1}^{M_{nk}} y_i(n) t_i. \tag{14}$$

## 3. Numerical results

Let us assume that our goal is to carry on with the traffic analysis of the access part of an UMTS network that includes RNC and 7 base stations (NodeB). The considered network services a mixture of R99 and HSDPA traffic, also including the Mobile TV service, which requires k-cast connections. According to the second scenario (Section 2.1.), multicast connections will be effected by Iub interfaces. In the case under discussion, to evaluate blocking probabilities of k-cast calls, the dependencies presented in Section 2.3. will be used.

The proposed analytical model of the UMTS system with multicast connections is an approximate model. Thus, the results of the analytical calculations were compared with the results of the simulation experiments. The study was carried out for users demanding a set of following traffic classes (services) in the downlink direction:

- class 1: Speech (VoIP, non-compressed, $t_1$=16kbps),
- class 2: Realtime Gaming ($t_{2,\min}$=10kbps, $K_{2,\max}$=1.5),
- class 3: FTP ($t_{3,\min}$=30kbps, $K_{3,\max}$=3),
- class 4: Mobile TV ($t_{4,\min}$=64kbps, $K_{4,\max}$=2),
- class 5: Web Browsing ($t_{5,\min}$=500kbps, $K_{5,\max}$=2.5).

In the presented study it was assumed that:

- $t_{BBU}$ was equal to 1 kbps,
- the system consisted of 7 Iub interfaces and each of the considered Iub interfaces carried traffic from three radio sectors, and a physical capacity of Iub in the downlink direction was equal to: $V_{Iub}$= 3 × 7,2 Mbps = 21,6 Mbps ($\cong$ 21 000 BBUs),
- the traffic classes were offered in the following proportions:

$$A_1 t_1 : A_2 t_2 : A_3 t_3 : A_4 t_4 : A_5 t_5 = 5 : 3 : 10 : 1 : 10.$$

  It was assumed that the main part of traffic was generated by FTP and Web Browsing services, followed by VoIP and Realtime Gaming services, while the smallest part of traffic came from Mobile TV service.

- Mobile TV was assumed to be the MBMS-based service which required 5 parallel connections and it was assumed that at least 3 connections had to be carried out by the system

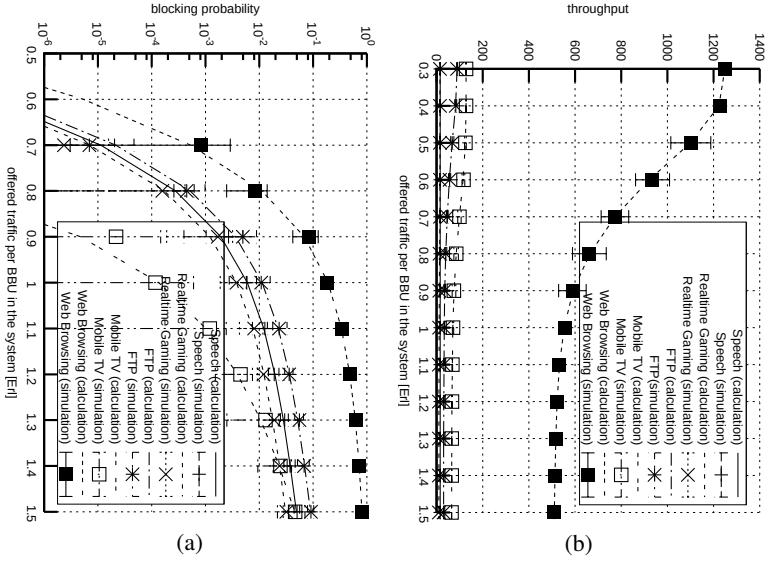



Fig. 4: Blocking probabilities (a) and average carried traffic (b) for particular classes carried by exemplary UMTS system.

Figure 4a shows the dependence of the blocking probability in relation to traffic offered per BBU in the system. The presented results were obtained for a minimum value of demanded resources for traffic classes with compression property.

It is easy to see that along with the increase in the load of the system, the blocking probability for all traffic classes offered to the system also increases. It is also easy visible that the least blocking probability is obtained for the traffic stream corresponding to the Mobile TV service (k-cast connection). In this case, the dependence of the blocking probability on the traffic intensity offered to the system is characterized by a stronger growing tendency. This is related to the way blocking for a k-cast connection was defined, where blocking occurs when there is no possibility of setting up connection concurrently in 4 or 5 links. This event exists statistically far more rarely than blocking events in a single link, while it has a considerable growing tendency with the increase in the load of the system.

Figure 4b shows the dependencies of the average throughput obtained for the calls that undergo compression effected by the load of the system. The results confirm a strong dependence between the average carried traffic (throughput) and the load of the system – the more overloaded system, the lower value of throughput [10].

The analytical results were validated by simulation experiments which were carried out on the basis of a simulation program specially designed for the purpose. In the conducted simulation results shown in Fig. 4, each point of the plot is the average value of the blocking probabilities obtained in 5 series. It was assumed that in particular simulation series $10^7$ of the incoming calls of the "oldest"[2] class were offered. The results of the simulations are shown in the charts in the form of marks with 95% confidence intervals calculated after the $t$-Student distribution. 95% confidence intervals of the simulation are almost included within the marks plotted in the figures.

## 4. Conclusion

The interest in the MBMS standard (Multimedia Broadcast and Multicast Service) [1] and in providing multimedia services with the application of multicast connections has a growing tendency. This sector of the service market can be expected to bring much profit to operators of cellular networks in the future and this is why appropriate planning procedures in developing and expanding the infrastructure of the network as well as optimization of already existing resources is so vital. This can ensure operators to successfully provide new multimedia services to the extent and quality that will satisfy prospective subscribers.

In the paper we propose a new blocking probability calculation method for cellular systems with mulicast connections. The proposed analytical method permits traffic analysis of cellular systems in which multicast-type connections are already

[2]The class which demands the highest number of BBUs.

used or are planned to be introduced. The method makes it possible to not only take into consideration the influence of the necessity of concurrent service on the effectiveness of the system (multicast) but also to take into account the variable bitrate of service (such as Web-browsing or FTP), as well as different definitions of blocking for multicast connections. The diversity of definitions of blocking can also be used in risk (cost) assessment evaluation of investments and the assumed level of the quality of service.

It is worth emphasizing that, despite the approximate character of the presented analytical method, the method is characterized by high accuracy, which has been confirmed by numerous simulation experiments. The method is also characterized by low computational complexity and can be easily applied in engineering calculations of capacities in the UMTS/HSPA/LTE systems.

## References

[1] A. Jajaszczyk, G. Iwacz and M. Zajączkowski, *Multimedia Broadcasting and Multicasting in Mobile Networks*, Wiley, Chichester, 2008.

[2] D. Staehle and A. Mäder, "An analytic approximation of the uplink capacity in a UMTS network with heterogeneous traffic," in *Proc. of 18th International Teletraffic Congress (ITC18)*, Berlin, 2003, pp. 81–91.

[3] V. B. Iversen and E. Epifania, "Teletraffic engineering of multi-band W-CDMA systems," in *Network control and engineering for Qos, security and mobility II*. Norwell, MA, USA: Kluwer Academic Publishers, 2003, pp. 90–103.

[4] I. Koo and K. Kim, "Erlang capacity of multi-service multi-access systems with a limited number of channel elements according to separate and common operations," *IEICE Transactions on Communications*, vol. E89-B, no. 11, pp. 3065–3074, 2006.

[5] V. G. Vassilakis and M. D. Logothetis, "The wireless Engset multi-rate loss model for the handofftraffic analysis in W-CDMA networks," in *Proc. of 19th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2008, pp. 1–6.

[6] M. Stasiak, P. Zwierzykowski, and D. Parniewicz, "Modelling of the WCDMA interface in the UMTS network with soft handoff mechanism,," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Honolulu, USA, December 2009.

[7] M. Stasiak, A. Wiśniewski, P. Zwierzykowski, and M. Głąbowski, "Blocking probability calculation for cellular systems with WCDMA radio interface

servicing PCT1 and PCT2 multirate traffic," *IEICE Transactions on Communications*, vol. E92-B, no. 4, pp. 1156–1165, April 2009.

[8] M. Głąbowski, M. Stasiak, A. Wiśniewski, and P. Zwierzykowski, *Performance Modelling and Analysis of Heterogeneous Networks*, ser. Information Science and Technology. River Publishers, 2009, ch. Uplink blocking probability calculation for cellular systems with WCDMA radio interface and finite source population, pp. 301–318.

[9] M. Stasiak, J. Wiewióra, and P. Zwierzykowski, "Analytical modelling of the Iub interface in the UMTS network," in *Proc. of 6th Symposium on Communication Systems, Networks and Digital Signal Processing*, Graz, Austria, July 2008.

[10] M. Stasiak, P. Zwierzykowski, J. Wiewióra, and D. Parniewicz, *Computer Performance Engineering*, ser. Lecture Notes in Computer Science. London, Springer, July 2009, vol. 5652, ch. Analytical Model of Traffic Compression in the UMTS network, pp. 79–93.

[11] H. Holma and A. Toskala, *WCDMA for UMTS. Radio Access For Third Generation Mobile Communications*, Wiley, Chichester, 2000.

[12] H. Holma and A. Toskala, *HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications*, Wiley, Chichester,, 2006.

[13] J. Laiho, A. Wacker, and T. Novosad, *Radio Network Planning and Optimization for UMTS*, 2nd ed., Wiley, Chichester, 2006.

[14] M. Nawrocki, H. Aghvami, and M. Dohler, *Understanding UMTS Radio Network Modelling, Planning and Automated Optimisation: Theory and Practice*, Wiley, 2006.

[15] F. Kelly, "Loss networks," *The Annals of Applied Probability*, vol. 1, no. 3, pp. 319–378, 1991.

[16] J. Kaufman, "Blocking in a shared resource environment," *IEEE Transactions on Communications*, vol. 29, no. 10, pp. 1474–1481, 1981.

[17] J. Roberts, "A service system with heterogeneous user requirements — application to multi-service telecommunications systems," in *Proc. of Performance of Data Communications Systems and their Applications*, North Holland, 1981, pp. 423–431.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 247–258

# Statistical analysis of active web performance measurements

MACIEJ DRWAL[a] LESZEK BORZEMSKI [a]

[a]Institute of Informatics
Wrocław University of Technology, Wrocław, Poland
{maciej.drwal | leszek.borzemski}@pwr.wroc.pl

**Abstract:** Web performance is an emerging issue in modern Internet. We present a study of the Web performance measurement data, collected during over a year of operation of *MWING* (Multi-agent Web-pING tool), our web server probing system. Both periodic and aperiodic factors influencing the goodput level in HTTP transactions are considered and analyzed for the predictive capability. We propose several hypotheses concerning the goodput shaping, and verify them for the statistical significance level, using the analysis of variance techniques. The presented work provides universal guidelines for designing advanced web traffic engineering applications.

**Keywords:** performance modelling, Internet, estimation, ANOVA

## 1. Introduction

The Internet builds up environment for such applications as every day information supply, communication and high performance computing. The core components of the global telecommunication network, created around the Internet Protocol (IP), cooperate mutually, in order to provide efficient platform for data packet delivery. Due to the open and evolving structure, number of coexisting protocols, services and different types of users, the Internet can hardly be analyzed as a strictly defined system, but more like a phenomenon, with its own intrinsic nature.

In this research our main concerns are techniques of HTTP goodput forecasting. We present the study of web performance measurement data, collected by *MWING* measurement tool [1]. We show that it is possible to perform reliable statistical inference from the previously collected HTTP transaction data.

This paper is organized in the following way. In the next section, we describe the performance data used for the analysis. In section 3., we present the results of variance analysis, suggesting periodic patterns in goodput level, as observed

on fixed paths. In the section 4., cross-dependencies between different clients are considered. Section 5. presents the regression analysis, complying both periodic and aperiodic goodput predictors. Lastly, section 6. contains a brief summary.

## 2. Description of the experimental design

The *MWING* project [1] (which stands for *multi-agent web-ping* tool), an extended version of the *Wing*, is a distributed performance measurement system. The key idea was to have a configurable platform for Internet active measurements of web servers. One of the major advantage of *MWING* is that each client operates on a dedicated machine, and accesses the Internet without any interfering traffic. This results in the clean throughput/goodput measurements, eliminating unwanted noise in the data.

A main aim of MWING system is to provide a distributed infrastructure for collecting experimental data through performing intentional active and passive network-wide measurement experiments which could be interpreted, processed and systematized to deliver information and knowledge about Internet performance and reliability issues.

MWING has a multi-agent architecture supporting system measuring capabilities in different operating conditions of local Internet domains where MWING agents are to be deployed. Agents can also include own specific functionality. The system measuring functionalities supported by MWING include local and central database, measurement scheduler, heartbeat service, and Internet-wide host time synchronization. Data measured locally is loaded to local databases which are automatically synchronized with a central database to gather all measurements in a unified way to be able to analyze them together in a unified way. Agents may force the synchronization of clocks of local hosts to synchronize their runs in time (by means of NTP-based MWING service).

In this Internet knowledge discovery research project we used MWING system in the specific distributed measurement experiment which was designed to achieve performance knowledge about Web.

*MWING* operated from four client nodes, located in 3 universities in Poland (Wroclaw, Gdansk, Gliwice) and one in USA (Las Vegas). They are denoted in the texts as WRO, GDA, GLI and LAS, respectively. Each client probed periodically a fixed set of 60 web servers, providing (approximately) exact resource to download.

A methodology considers a model of an individual Web (HTTP) transaction featuring the time diagram as shown in Figure 1. Each single measurement resulted in a detailed information on the subsequent Web transaction stages: time to resolve the IP address from name via DNS query, time to establish a TCP con-

nection, time to start a HTTP transaction, time to complete the whole transaction, and the in-between delays. Such measurements, similar in the idea to the standard network ping, give the instantaneous information about the state of the web server performance.

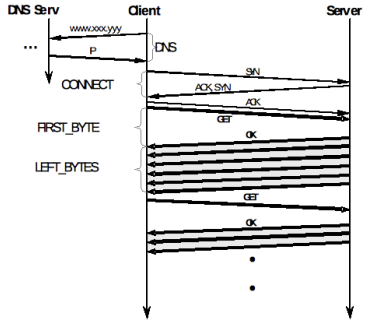



Fig. 1. Time diagram of Web transaction.

The most important information from the measurements is the goodput estimate on a probed path: the value obtained by dividing the downloaded resource size $S$ (in bytes) by total measured time $T$ (in seconds), i.e.: `Gpt` $= \frac{S}{T}$. This value represents the quality of transmission as perceived by user, as opposed to the throughput (`Thr`), which measures the raw data transfer, regardless of its contents. The goodput takes into account all the data transferred (both packet information and resource data, possibly including errors, or retransmitted information) in the measured time. On some links, especially highly lossy, the measured throughput characteristics can be much higher than the actual goodput. As we are interested in the performance observed on the user level, we do not consider throughput in our analysis.

| | |
|---|---|
| **Time** | date and time of measurement |
| **Dns** | time taken by DNS query |
| **D2S** | DSN to SYN delay (SYN starts TCP connection establishment) |
| **Con** | time to establish TCP connection |
| **A2G** | ACK to GET delay |
| **First** | time elapsed from sending GET and receiving first response packet |
| **Last** | remaining time of resource fetching |
| **Meas (T)** | $Dns + D2S + Con + A2G + First + Last$ |
| **Size (S)** | resource size |

Table 1: Summary of web performance data measured by *MWING*. Each such measurement is performed between client-agent and one of known web servers.

All the datasets used in our analysis come from the measurements performed

in the time period 2008.4.24 — 2009.7.5. They were collected by the uniform *MWING* probing with 30 minutes interval. The actual number of collected cases in final datasets is lower due to temporary servers' failures or network malfunctions.

After over a year of operation, the data collected by *MWING* show fairly complete picture of the performance characteristics of the observed web servers.

## 3. Analysis of the periodical phenomena

The use of uniform sampling scheme brings its own inductive bias (see [9]), and whenever high frequency components detection is needed, Poisson sampling should be used. The considered network traffic data is suspected to contain many periodic components in its spectrum. In the uniform sampling, due to the aliasing, the sample cannot contain components of higher frequency than $f_s/2$ Hz, where $f_s$ is a sampling frequency (we used $f_s = 0.56$ mHz, $\omega = 1800$ s). Thus, the shortest effects we consider are of order of 1 hour.

We have proposed and tested a number of general hypotheses concerning long term behavior of measured performance characteristics.

### 3.1. Variability on days of the week

It is suspected that the level of goodput depends on the day of the week. This is due to the different levels of usage of common network resources by specific user classes (for example, some industry users produce considerably less Internet traffic during weekends). We assume these effects are permanent, i.e. for different days of the week the average level of goodput should differ, but for each of these days these levels should be similar throughout the year.

Using analysis of variance we have proven that these effects are statistically significant. We hypothesize that the measurement samples from different days of the week give the same distribution of goodput (and seek to reject this hypothesis). We consider two test cases: all servers' data for each client (covering 2 months of observations each), and all single end-to-end measurement datasets (covering whole observation time, over one year). The first case allows to determine if most of the servers behave similarly for a given client, as observed during the week. The second case gives detailed information on which of the servers give significant differentiation.

Instead of using `Gpt` value directly, we transform it via logarithm function: $\texttt{LogGpt} = \log_{10} \texttt{Gpt}$. The cumulative histogram of all servers gives the shape of `LogGpt` very close to the normal distribution. However, the distributions of `LogGpt` in single end-to-end datasets are rarely close to normal. Aware of this fact,

in ANOVA tests we use both standard Snedecor's F-statistic, as well as Kruskal-Wallis ranks method. The standard one-way method assumes normal distribution among samples, but is known to perform well even with deviations from normality (see [5]). The Kruskal-Wallis test is a nonparametric method, which compares medians instead of means.

After calculating chi-squared statistic with 6 degrees of freedom (amount of days of the week minus 1) for all servers datasets (test case 1.), in three cases we can reject the hypothesis with significance level $\alpha = 0.05$[1].

Only goodput from GLI dataset is not evidently changing on different days (however smaller variations can be still observed). This suggests an interesting feature: the larger the client network is, the more sensitive it is to such periodic factors.

| | **LAS** | **WRO** | **GDA** | **GLI** |
|---|---|---|---|---|
| **p-value** | $\approx 0.0$ | 0.00052 | 0.036 | 0.10833 |

Table 2: ANOVA for cumulative goodput—time of the week dependency. For LAS, WRO and GDA datasets we have p-value $< 0.05$. Insignificant influence in GLI may be explained by lower overall traffic.

Because full datasets give only overall impression of the average goodput shaping, we also compare day of the week influence on the single end-to-end measurement datasets (test case 2.). This is illustrated in Figure 2. The Table 3 shows the number of client-server pairs for which the day of the week was significant for measured goodput (with level $\alpha = 0.05$). The second value is a number of all servers, for which the tests were made. This number is different among analyzed clients, because for some of them there were not enough measurements collected to have equal statistical power. Nevertheless, we have enough data to conclude, that this factor is generally very significant.

| | **LAS** | **WRO** | **GDA** | **GLI** |
|---|---|---|---|---|
| **significant cases,** $\alpha = 0.05$ | | | | |
| **F-statistic test** | 38/44 | 46/52 | 29/45 | 25/54 |
| **Kruskal-Wallis test** | 29/53 | 53/61 | 38/59 | 28/61 |

Table 3. ANOVA for end-to-end goodput—time of the week dependency.

### 3.2. Variability on time of the day

Similarly, we expect the strong variability in goodput depending on the time of the day. This is another factor coming from the users' habits (for example, there

[1]The tests presented in this paper were performed in R statistical computing environment (see [8]), and reproduced, when possible, in Statistica 8 (see [7]).

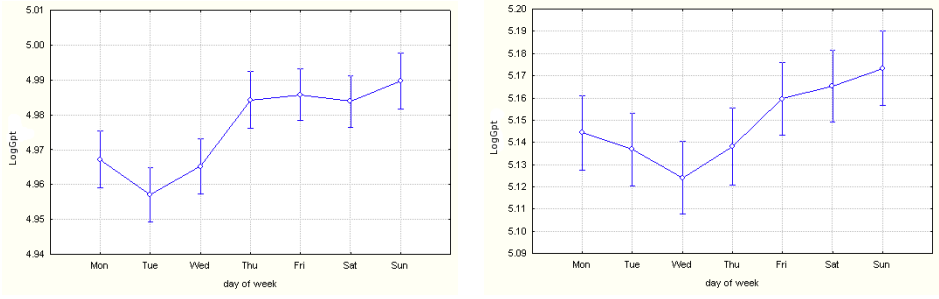



Fig. 2: Hypotheses decomposition on an example *MWING* datasets for goodput—day of week dependency. Vertical bars denote 0.95 confidence intervals. On the left: all servers requested by WRO client (test case 1.). On the right: one example server requested by WRO client (test case 2.). The average goodput increases at the end of week, and very quickly drops at Monday. The lowest values are on Tuesday and Wednesday.

should be lower traffic produced by some industry clients at the night hours). This is not immediately obvious, because the active server machine could be located in a completely different time zone than client.

For client-server pairs, using one-way ANOVA, we reject the null hypothesis, that the long-term goodput distribution is time of the day independent. As previously, we use `LogGpt`, to neglect the influence of irregularities in distributions of `Gpt`. This time we use chi-squared distribution with 23 degrees of freedom (number of hours minus 1) for F-statistic tests. Again, in most cases, the `LogGpt` was not distributed normally, thus we also use Kruskal-Wallis tests.

The Table 4 shows the summary of significant tests with level $\alpha = 0.05$. The Figure 3 shows typical decomposition of goodput variability under analysis.

| | **LAS** | **WRO** | **GDA** | **GLI** |
|---|---|---|---|---|
| **significant cases,** $\alpha = 0.05$ | | | | |
| **F-statistic test** | 27/48 | 47/52 | 30/48 | 27/54 |
| **Kruskal-Wallis test** | 35/53 | 55/61 | 33/59 | 31/61 |

Table 4: ANOVA for end-to-end goodput—time of the day dependency. Except from LAS client, the influence is even slightly more noticeable than the one from the day of the week.

This time only the LAS client distinguished itself from others, by revealing balanced behavior on different hours throughout the day more often than the rest. Even though, a very significant differentiation was observed for LAS in more than a half of the probed servers. This allows for a conclusion, that this factor is very influential.

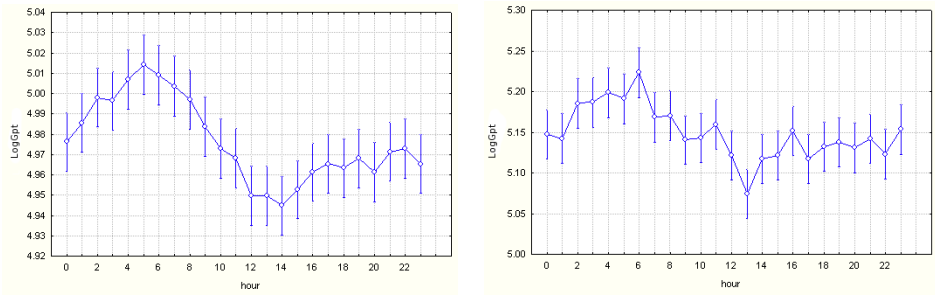



Fig. 3: Hypotheses decomposition on an example *MWING* datasets for goodput—time of day dependency. Vertical bars denote 0.95 confidence intervals. On the left: all servers requested by WRO client (test case 1.). On the right: one example server requested by WRO client (test case 2.). The average goodput is higher at the night hours and early morning (maximal value between 5—6 a.m.), as the usage is lower. During the day goodput drops down, and the lowest values are observed about 1—2 p.m., raising in the evening hours.

## 4. Analysis of the cross-path similarity

Very often the requests from different clients travel through a common section of the network path to a target server. Therefore, we are interested if the inference concerning one client could be extended to another one. In other words, we would like to determine, how much information about goodput is contained in the server's surroundings, and how much in the first part of the path from the client.

In order to test this, we have compared the goodput histograms of the common servers, servicing 4 different clients. For each test we take two datasets: measurements of common server for two different clients. Using this approach, we consider the estimates of probability distributions of goodput per each end-to-end observation, performed for a long time. Such method of comparison allows for a small time-local differences, focusing on the long term behavior. Otherwise, there is a risk of making wrong conclusions for the prediction purposes; if we consider the data only as a time series, and try to evaluate the goodness of fit (after normalization), we would always get very poor results.

We have used `LogGpt` normalized to the unit interval $[0, 1]$, via formula $z_i = (x_i - \min x_i)/(\max x_i - \min x_i)$. Due to this transformation, only the shape of the estimated goodput distribution is considered, and the differences in magnitudes are neglected.

We consider the goodness of fit of the two estimated distributions as a measure of datasets' origin consistency. If two such estimates are similar, we conclude that the underlying data have been produced by the same true probability distribution.

In our evaluation we make use of the standard coefficient of determination

$R^2 = \frac{SS_E}{SS_R}$, where $SS_E$ is a residual sum of squares, computed from a linear model fit, and $SS_R$ is total sum of squares (differences from the data points to their mean). This coefficient shows how linearly correlated are the both curves estimating two distributions. If this value is considerably high (e.g. $R^2 > 0.3$), we can suspect that both estimates are similar.

If the similarity is suspected, we can perform more restrictive test. We hypothesize that both estimated distribution are equal. The standard chi-squared goodness of fit test (see [4]) can be used for this purpose. The rejection of null hypothesis in this tests means that the two datasets were produced by significantly different distributions. The positive result gives a fairly good evidence that the goodput shaping on both end-to-end paths is the same. This test however always failed, as the considered distributions were very irregular (usually long-tailed and multimodal).

Another form of chi-squared test was more helpful. The test of independence, asserting the null hypothesis that corresponding samples from two clients' datasets are statistically independent, was performed on each client-server pair. Here, rejection of the hypothesis with $\alpha = 0.05$ means, that the two random processes governing the goodput levels, are dependent.

For performing such tests, it is a matter of accuracy requirements and the assumed sensitivity of observation level, to determine the test parameters, such as minimal $R^2$ value for acceptance and the discretization level (number of bins) in the chi-squared test. As we operate on histograms in order to estimate probability distributions, we need to select appropriate number of cells. This describes the level of detail we wish to consider. However, taking too many of them make the chi-squared tests fail too often, while we want to tolerate minor differences. For the chi-squared tests we use 30 bins.

| | WRO-GDA | WRO-GLI | WRO-LAS | GDA-GLI | GDA-LAS | GLI-LAS |
|---|---|---|---|---|---|---|
| all pairs | 55 | 59 | 50 | 55 | 46 | 49 |
| $\chi^2, p < 0.05$ | 29 | 23 | 13 | 34 | 21 | 10 |
| $R^2 > 0.3$ | 11 | 7 | 6 | 22 | 19 | 11 |
| $R^2 > 0.5$ | 4 | 5 | 2 | 16 | 12 | 7 |

Table 5: This table summarizes the cross-server measurement similarities. For the chi-squared tests of independence, the table contains the numbers of cases for which the hypothesis was rejected, thus the measurements for two clients were mutually dependent. Also the numbers of cases, for which the $R^2$ goodness of fit values were high are included.

It turns out that for a small number of client pairs the similarity in goodput distributions is apparent. This suggests that either the routing paths cover significantly, or that the goodput is mainly determined by the server's conditions (and possibly by the few of the hops located close to the server). This is especially visible for GDA and GLI clients, which both share a major section of the core network, and their

behavior is the most similar. In the majority of cases, the estimated distributions were alike; compare second and third histogram in Figure 4.

For the rest of the HTTP transactions, the majority of the average time spent on a packet delivery depends on the client's surroundings. After reaching the high throughput switches of the core network, the fluctuations in measured time are much lower. Also the server performance efficiency does not bring so much influence (note however, that the servers probed by *MWING* are mostly low traffic ones). The most important part in the packet delivery lies in the last mile and in the network infrastructure located intermediately between the client and the core.

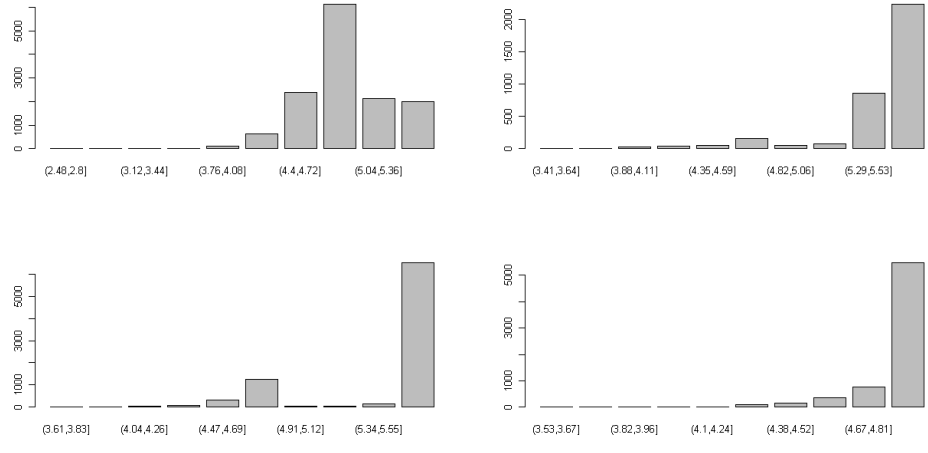



Fig. 4: Comparison of goodput shaping for the same server, measured for 4 different clients. From the top left: WRO, GDA, GLI, LAS.

The Figure 4 shows that goodput distributions for the same server can differ significantly for different clients.

## 5. Goodput regression analysis

From the long term observations it is evident that the goodput changes are not only of the periodic nature. In fact, the web traffic is known to be long-tail distributed and self-similar (see [6]). Estimated goodput distributions are generally multimodal, and the observed time series are characterized by high burstiness and irregularities. The measured goodput values oscillate around the periodic components, with many different kinds of deviations, produced by temporary factors. This makes the Internet traffic parameters hard to model and forecast.

As it was mentioned earlier, in our datasets, we can only observe effects lasting for at least one hour. Thus in our analysis we are able to detect aperiodic deviations from steady goodput shape curve, only if the effect is sufficiently important (observable by the user).

The *MWING* tool collected detailed timings of the subsequent client-server transaction stages. The value of goodput in each measurement is proportional to the sum of all the measured time periods. In practice we are unable to wait until the end of transaction to estimate goodput. Instead, we wish to use the server response time estimate, based on the measurement of a small part of the total transaction time. Such information can be thought of as a round trip time type of measurement.

We have analyzed the correlation matrices of the *MWING* measurements. For a typical client-server dataset the correlation coefficients between `LogGpt` and `Dns`, `Con` and `First` (see Table 1) measurements were 0.4—0.5. Therefore, we can consider the value $r = \texttt{Dns} + \texttt{Con} + \texttt{First}$ as a value linearly correlated with goodput.

This makes the value of $r$ a natural candidate for the goodput predictor. Our experiments show that the predictions based on the measurements of $r$ are effective in practice. Linear regression of $\log r$ versus `LogGpt` gives the coefficient of determination $R^2$ around 0.4—0.6 (see Table 6).

In more detailed prediction settings, we can consider separate predictors `Dns`, `Con`, `First`, or even nonlinear functions of them, e.g. $r^2$, $\lambda r^2 + (1 - \lambda)r$, etc. In addition, we can similarily build nonlinear regression or classification models, using also periodic component predictors (time of the day, day of the week), which increase predictive power. This approach is a subject of our further research, see [3].

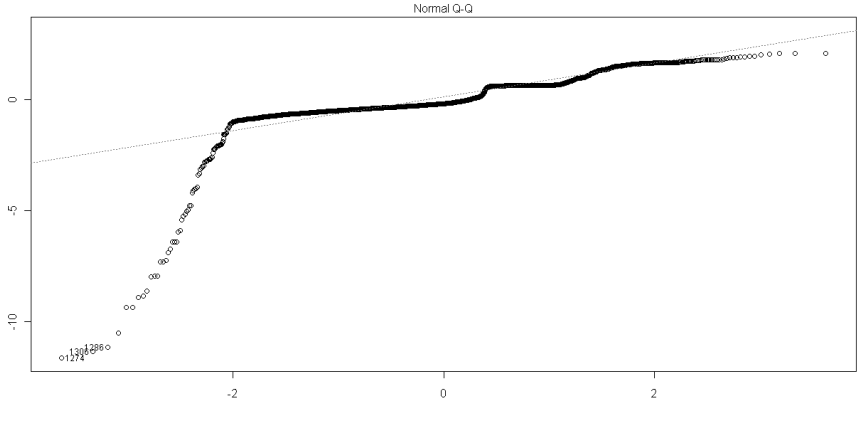



Fig. 5: Example QQ plot, obtained by fitting linear regression model of `LogFirst` to `LogGpt`, $R^2 = 0.8$. We observe major deviations mainly in the extreme regions.

| LAS | WRO | GDA | GLI |
|---|---|---|---|
| 0.66 | 0.37 | 0.52 | 0.49 |

Table 6: The mean value of $R^2$ from fitting linear regression model of `LogFirst` to `LogGpt`, computed for all servers.

## 6. Conclusions

The considerations presented in this paper provide useful guidelines for performing the feature selection in any measurement-based web performance prediction experiment. We have shown that the goodput level in HTTP traffic discloses changing regularities in time. Our results prove the time of the day dependency (based on client's local time) in all four datasets with significance level $\alpha = 0.001$ (considering all servers). Such high significance level suggests that the major impact on the perceived goodput comes from the requester's network surroundings. Both time of the day, and day of the week dependencies are valid, with significance level $\alpha = 0.05$, for the majority of observations. We have also observed that the variability is coupled with the size of client network.

These factors constitute periodic components in the user level traffic shaping. Apart from that, the aperiodic contributions to the goodput level can be determined from the measurements of subsequent transaction timings.

Our examinations also prove that the clients sharing the backbone network could expect similar goodput level significantly more often. A small collection of probing machines could provide reliable measurements for the purpose of goodput prediction covering a large network, e.g. for a small number of autonomous systems.

This conceptual framework provides a foundation for designing traffic engineering systems, operating on the high level in the Internet.

## 7. Acknowledgements

This work was supported by the Polish Ministry of Science and Higher Education under Grant No. N516 032 31/3359 (2006—2009).

## References

[1] Borzemski L. et al., MWING: A Multiagent System for Web Site Measurements, Lecture Notes in Computer Science: Agent and Multi-Agent Systems: Technologies and Applications, Springer Berlin, July 2007, p. 278-287

[2] Borzemski L, Cichocki Ł, Kliber M. Architecture of multiagent Internet system MWING release 2. Lecture Notes in Computer Science, vol. 5559, 2009

[3] Drwal M., Borzemski L. Prediction of Web goodput using nonlinear autoregressive models. The 23rd Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems, Córdoba, Spain 2010 (submitted for publication)

[4] Huber-Carol C. et al., Goodness-of-Fit Tests and Model Validity, Statistics for Industry and Technology, Birkhauser Boston 2002

[5] Lindman, H. R., Analysis of variance in complex experimental designs. San Francisco (CA): W. H. Freeman & Co. 1974

[6] Park K., Willinger W., Self-similar Network Traffic and Performance Evaluation, 1st Edition, John Wiley & Sons, Inc. 2000

[7] StatSoft, Inc. (2008). Statistica (data analysis software system), version 8.0, `http://www.statsoft.com`

[8] The R Project for Statistical Computing, `http://www.r-project.org`

[9] Wolf T., Cai Y., Kelly P., Gong W., Stochastic Sampling for Internet Traffic Measurement, IEEE Global Internet Symposium, May 2007

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 259–272

# A Multi-Dimensional CAN Approach to CRN Routing

ALEXANDRU POPESCU [a,b] DAVID ERMAN[a] MARKUS FIEDLER[a]
DEMETRES KOUVATSOS [b]

[a]Dept. of Communications and Computer Systems
School of Computing
Blekinge Institute of Technology
371 79 Karlskrona, Sweden

[b]Dept. of Computing
School of Informatics
University of Bradford
Bradford, West Yorkshire BD7 1DP, United Kingdom

**Abstract:**
The paper reports on an innovative approach to the management of Cognitive Radio Networks (CRNs). As part of achieving this functionality, a new architecture implemented at the application layer is presented. The research challenges for developing a CRN management solution are on addressing, sensing and prediction, routing, decision making and middleware. Cognitive radio networks are expected to address and to solve several important challenges like, e.g., opportunistic spectrum access, spectrum heterogeneity, network heterogeneity and the provision of diverse Quality of Service (QoS) to different applications. Consequently, a number of management functions are necessary to address the associated challenges with the management of CRNs. The paper provides a brief introduction to these components while focusing on the CR routing, addressing and content representation.
**Keywords:** cognitive radio networks, content addressable networks, addressing, routing

## 1. Introduction

Given that an overcrowding of the available frequency bands is present today, there is a need to do research on and develop Cognitive Radio Devices (CRD) able to sense and identify vacant domains in the spectrum. A CRD is simply put a device that is able to learn from experience and can adapt its operational parameters

to function under any given conditions, thus being able to make use of the under-utilized parts of the available spectrum. In order to successfully employ a Software Defined Radio (SDR), intelligent management and control facilities need to be conceptualized. By harvesting the benefits offered by P2P systems we aim at advancing a new architecture implemented at the application layer for the management of cognitive radio networks (CRNs). This can be achieved by invoking a specific adapted distributed P2P addressing system, together with additional functionality disseminated throughout several overlays connected through a middleware. The paper reports on the framework of a new architecture while focusing on the Cognitive Routing (CR), addressing and content representation.

The remainder of the paper is as follows: In Section II we present an overview of general Content Addressable Network (CAN) operation as well as a short introduction to the suggested Multi-Dimensional CAN (MD-CAN) approach to CRN management. Section III presents the architectural solution for management and the main elements are described. In Section IV we comment on modeling aspects of the MD-CAN. Section V is dedicated to describing the solution adopted for CRN routing. Finally, Section VI concludes the paper.

## 2. Multi-Dimensional CAN

As part of the CR management solution, we suggest an architecture where a modified MD-CAN (a multi-dimensional extension to the standard CAN implementation [5]) overlay is used for the information representation of a multihop CR-network. In regular CAM implementation member nodes have $\Theta(2d)$ neighbors and the average path lengths are of $\Theta(d/4)(\,n^{1/d})$ hops [5]. By increasing the number of dimensions we gain the benefit of shorter path lengths, though at the price of higher per-node-states. Every node now has to keep track of its neighbors in all neighboring dimensions not solely its own dimension. However, by utilizing a MD-CAN approach we have a solution that can be modified to suit the particular needs for the management of CRNs. Different CAN dimensions can be used to represent different CR-dimensions like, e.g., space, frequency, power, code. All of them are functions of time and with independent coordinate spaces for every node. Rendezvous points are selected within a maximum geographical radius, for bootstrapping purposes to populate the available operational zones [5]. Joining of unlicensed users is done through a scheduler in order to fairly partition the available spectrum holes given as these are a limited resource. Figure 1 shows an example of Multi-Dimensional CAN (MD-CAN) representing $N$ nodes in different CR-dimensions during the time period $\Delta t$.

In our approach, the frequency domain is represented as a CAN dimension

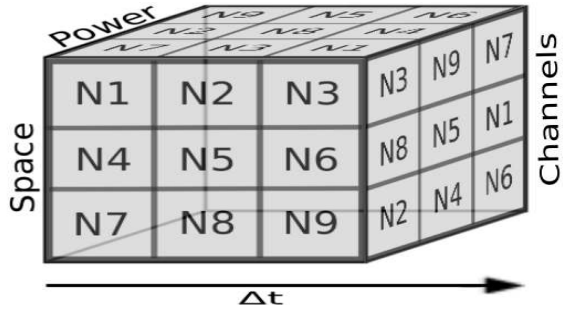



Fig. 1. Overview of a Multi-Dimensional CAN.

named Channels. This lets us depict all available operation opportunities (channels) in the available frequency spectrum. Depending on the employed technology different types of multiplexing in the available frequencies are used, enabling for instance the CR users to exploit the same frequency though at different time periods. Through the Channels dimension in an MD-CAN we can gather and depict the maximum amount of users that can inhabit the available spectrum simultaneously. This allows CR users to fully use the available resources when making adaptations to receive service.

The stored operational parameters for a node in different CR-dimensions are identified with the Cartesian coordinates $n(t) = (x(t), y(t), f(t), p(t))$, where $x(t)$ and $y(t)$ denote the space dimension (we assume 2-dimensional space for now, though a third space dimension might be required later for depicting height), $f(t)$ the frequency dimension (partitioned into channels) and $p(t)$ the power dimension, all of them as functions of time. This means that $n(t)$ can, at any given time, characterize a CRD occupying the operational zone at coordinates $x, y$ in the space dimension, channel $c$ in the frequency dimension and using transmission output power $p$ in the power dimension. Naturally, some parts of the MD-CAN functionality are rendered inactive, since different CR-dimensions are represented with every dimension instead of space alone, which otherwise is the case in regular CAN implementation. The defining characteristics of each dimension can also be changed independently of each other. A vacant position (zone) in a CAN dimension represents a hole in a particular CR-dimension as a function of time. Optimization can hence be achieved from the perspective of every single CR user, enabling proactive adaptation to network changes. Users might need to adapt in one or more CR-dimensions to keep the location in the space dimension. For instance, a slight change of the operating frequency or the transmit power output might be required to maintain a position in the space dimension and thus retain service.

The MD-CAN information is updated per dimension for all registered changes (e.g., new vacant positions in different dimensions, arriving licensed users pushing out unlicensed users) through the standard CAN control and takeover mechanisms. In regular CAN implementation only the affected neighbors are informed of the changes in the topology through the immediate updates. On the other hand, in our solution, all member nodes (not just the affected neighbors) are informed of the changes, with updates that spread outwards from the affected neighbors. Having all CAN members storing global network topology information is contrary to regular CAN implementation. However, since CR users depend on fast adaptations in one or more CR-dimensions, to minimize lookup delays and retain service, every user is responsible to store relevant information related to possible future adaptations. To retain the operational zone $< x_2, y_1 >$ in the space dimension an adaptation in the frequency and power dimensions may be necessary, for instance from: $< x_2, y_1, f_4, p_5 >$ to $< x_2, y_1, f_9, p_3 >$. These adaptations are bounded by $t$ and represented as MD-CAN dimensions. In other words, this is the process of spectrum mobility represented as multi-dimensional routing with the goal of providing smooth and fast transition, with a minimum of performance degradation [2].

## 3. Management Architecture

Cognitive Radio Devices (CRDs) are able to change their operational parameters on the fly, depending on the surrounding environment and consequently emerging as a viable solution to spectrum shortage problems [2]. A CRD collects information about the characteristics of the surrounding environment like, e.g., frequency, modulation and transmission power and is able to adapt and learn from experience. The challenging research task is to develop CRDs able to sense and identify vacant domains in the spectrum. The so-called spectrum holes (unused spectrum available for unlicensed users) are present at different time periods and places, thus new facilities have to be devised to utilize them for communication. Cognitive radio networks provide these facilities by sharing the temporally unused resources in an n-dimensional space with unlicensed mobile users. Examples of such dimensions are geographical space, power, frequency, code, time and angle [1, 2, 9]. Four fundamental operations are needed for the management of a CRN [1, 2, 17]: Spectrum sensing, spectrum decision, spectrum sharing and spectrum mobility.

These operations are necessary in order for the two CRN user classes (licensed and unlicensed) to function seamlessly. The licensed users exploit the licensed part of the spectrum band and the unlicensed users use the available spectrum holes (free spaces) in the spectrum band. Whereas the control for the licensed band is focused

on the detection of primary users, the control for unlicensed bands is trying to minimize possible conflicts and interference to primary users when accessing the spectrum holes. Given the extreme complexity present in the process of managing CRNs, we propose a new architecture to achieve this functionality. To control the four fundamental operations in the management of a CRN, we base our solution on the use of a middleware with a common set of Application Programming Interfaces (APIs), a number of overlay entities and a multi-dimensional routing. The proposed architectural solution is shown in figure 2.

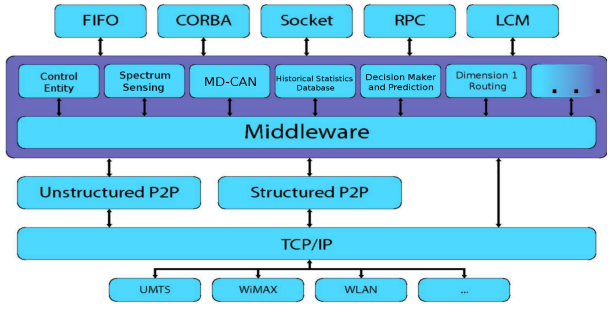



Fig. 2. CRN Management Architecture.

The specific middleware presented in figure 2 was originally developed by the BTH research group and it is a software that bridges and abstracts underlying components of similar functionality, exposing it through a common API [3]. Through the deployment of our middleware (and using different overlays) we can ensure a flexibility of adding new services, through future overlays.

An overlay is defined to be any network that implements its own routing or other control mechanisms over another already existing substrate, e.g., TCP/IP, Gnutella. With reference to the addressing type, we can partition the overlays into two categories, namely structured and unstructured. Structured overlays implement DHT to decide the type of routing geometry employed by the particular network. Unstructured overlays use IP addresses or other forms of addressing like, e.g., in the case of Gnutella, which uses Universal Unique IDs (UUIDs) [8]. Underlays are defined by us as being forwarding or transport substrates, which are abstracted using a common API and can also in their turn be either structured or unstructured. The topology of structured underlays is imposed meaning decided beforehand like in the case of Chord [4], whereas unstructured overlays topology can instead be viewed as emergent. Furthermore, to move away from the current incompatible overlay implementations and facilitate the integration of different overlays in our CR management solution, a middleware based on the Key-Based Routing (KBR)

layer of the common API has been implemented [7]. This way an independent development of overlay protocols, services and applications is viable. We employ two different sets of APIs, one for application writers and one for interfacing various overlays and underlays. An exported API by one overlay can be used by other overlays, like in the case of "Sensing", "Prediction", "Decision Making" and "Routing", where all have to work together to provide the needed functionality.

The following overlays are proposed: Control Entity, Spectrum Sensing, Multi-Dimensional CAN (MD-CAN), Historical Statistics Database, Decision Maker and Prediction and Dimension 1-4 Routing. The management solution used in our system is an intelligent wireless communication system, defined to represent a set of algorithms. Its purpose is to perform sensing, learning, optimization and adaptation control of the CRN [11].

A CRN is a dynamic distributed network that changes constantly. Given the difficulty of accurately predicting future changes in advance (e.g., an unlicensed user that occupies a spectrum hole at a certain time period might be pushed out by an arriving licensed user taking over that particular spectrum band), the routing decisions have to be flexible, adaptable and performed on the fly. These decisions are depending on both current environmental constraints, (e.g., spectrum availability, power output, coded signal type, angle of arrivals) and the considered statistics. The statistics are uniquely collected for every CR-dimension. To make accurate decision, several mechanisms are used depending on the scenario at hand, e.g., Multiple Criteria/Attribute Decision Making, Fuzzy Logic/Neural Network and Context-Aware mechanisms. Since CRN are enabled to learn from experience, this entails compiling statistics from previous network changes like, e.g., cost, throughput, delay, primary user activity. These statistics are stored on the nodes (CRDs) participating in the MD-CAN overlay (a geometric multi-dimensional representation of the CR dimensions) and are gathered for each node through spectrum sensing. In other words, occupancy in an n-dimensional CR space is identified and in our case stored and represented through an MD-CAN. Simply put, this is the ability of a CRN to measure, sense, learn and be aware of environmental changes and restrictions i.e., find opportunities in the different CR domains [1, 9]. These parameters can be related to things like network infrastructure, radio channel characteristics, spectrum and power availability, local policies and other operating restrictions.

For a functional architectural solution the complexity is disseminated to different overlays. Further, even though we employ a modified MD-CAN for the suggested CR management architecture, typical CAN mechanisms for topology construction, maintenance and addressing are assumed [5]. We also assume a bounded number of users per dimension. Optimization is performed from the perspective of all CRN users and every CR dimension. The maximum number of users that

a CAN dimension can accommodate is determined by the limiting CR dimension placing an upper limit $N_{\max}$ on the network size. If the spectrum band becomes the limiting dimension, all other CAN dimensions inherit the particular upper limit of users that can be accommodated, i.e., all users are represented in every dimension.

To achieve this, a lightweight control mechanism is required (minimizing the signaling overhead) together with a low out-of-band frequency utilization in the unlicensed spectrum. This is a so-called Common Control Channel (CCC). The CCC is conceived to cover long distances, though operate at low rates [1, 10]. Furthermore, carrying all control information on a single channel eliminates synchronization problems, which otherwise can arise from having users tuned to different channels. We suggest to place the CCC in the unused parts of the UHF band, known as UHF white spaces which operates over long distances, perfectly matching our requirements [12]. Replacing the need for large-scale broadcasts over multiple channels, the CCC has to be available to all nodes in the network at all times, enabling so the MD-CAN overlay to maintain its topology and stored information up to date. However, the time needed until information (regarding a node's profile) updated on the CCC is available for every node has to be considered. This time is bounded to the number of CR users present in the network [10]. It takes longer for the information to reach the CR users farthest away from the source in case of a large CRN. Changes are furthermore registered through the collaboration with the Spectrum Sensing overlay and the Control Entity overlay. The Spectrum Sensing overlay collects connectivity parameters for each CR user and for a group of CR users as well. The collected connectivity parameters per user are communicated to the MD-CAN overlay for statistics computation. The computed statistics per user and per CR-dimension are saved on the participating nodes (CRDs) represented in each corresponding virtual CAN dimension.

Naturally, the MD-CAN stored statistics per CR user are subject to constant changes, meaning users come and go. The Decision Maker and Prediction overlay takes therefore decisions based not only on current available user statistics, but also on statistics computed over a longer time period. Statistics of the CRN changes gathered over time for a group of CR users are stored in the Historical Statistics Database overlay. This enables the Decision Maker and Prediction overlay to learn from experience and to predict future changes more accurately. This also means that the available spectrum holes are identified, creating so opportunities for unlicensed users to receive service. This identification provides relevant information for both unlicensed users already present in the CRN (to adapt their operational parameters and retain service) and for unlicensed users wishing to join the CRN and to receive service.

An important aspect for the MD-CAN spectrum hole representation to work

properly is the need to consider real geographical distances between users present in the CAN space dimension. This distance radius is bound by the members ability to operate in the place of any other member present in this domain. In short, a CR user should be able to adapt its operating parameters to function in every opportunity that may be presented in the MD-CAN. If the geographical locations are too distant, a user in one corner of the space domain might not be able to exploit a spectrum hole present in the other part of the space domain. This is simply due to the limited transmit power output and the ability to access the particular available spectrum that might be too far away. Accurate information representation in the MD-CAN overlay is therefore vital for the overall CR management functionality. The e2e source based routing is locally computed by the Decision Maker and Prediction overlay at the request of a communicating CRD considering factors like, e.g., QoS/QoE, cost, service availability, security, privacy levels and user defined preferences. This is possible through the use of the connectivity statistics per individual user stored in the MD-CAN overlay together with the group operational statistics stored in the Historical Statistics Database overlay.

The routing path is computed by predicting the CRN changes that may occur due to CRD adaptations. However, the accuracy of the computed path also depends on the quality of the available statistics. Furthermore, the Control Entity overlay is responsible for the actions related to user control, context-aware and security facilities. This particular overlay informs the MD-CAN and hence the Decision Maker and Prediction about user, context and service defined preferences as well as other relevant information, offering so the end user facilities for Always Best Connected (ABC) decisions. Since the e2e path is computed locally from the information stored in the MD-CAN and Historical Statistics Database, a simple lookup query in the locally stored virtual CAN space dimension suffices to find the desired destination. Retaining service in a CRN is however not only a matter of reacting to traffic load changes. A number of factors have to be considered to model and to predict user activity and mobility under the QoE framing. Users are also expected to perform their own measurements, contributing so to the MD-CAN overlay connectivity statistics buildup. The considered factors can furthermore affect one or more CR-dimensions simultaneously.

Finally, the Dimension Routing overlays are employed in collaboration with a global time dimension common to all CR-dimensions. The Dimension Routing overlays are responsible for performing the actual physical routing to reach the destination. Packets are forwarded in the space domain through the multi-objective optimized path compiled by the Decision Maker and Prediction overlay. The Dimension Routing overlays also need to maintain an updated MD-CAN structure to offer the needed functionality, meaning their topology updates are synchronized

with the MD-CAN overlay. Furthermore, the actual MD-CAN overlay topology might change during the time spent for compiling the source based e2e path in the Decision Maker and Prediction overlay. Therefore, when the actual physical routing begins, the Dimension Routing overlays might have more up-to-date information. An important responsibility for reaching the destination lies thus with these overlays. Own decisions have to be taken by them, adapting the precompiled e2e path according to the particular network conditions.

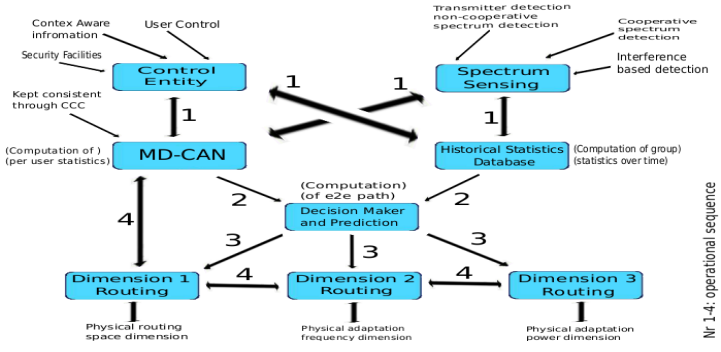



Fig. 3. Overlay functionality and operational sequence.

Figure 3 shows an overview of the overlay functionality and operational sequence described above.

1 Information gathering and statistics computation operations.

2 Source based e2e path computation from the statistics compiled in operational sequence 1.

3 Physical routing and adaptations to intermediate CRD along the precompiled e2e path from sequence 2. Enables communication packets to reach their destinations.

4 Allows for the destination to be reached even though the precompiled e2e path has changed. Own decisions and adaptations are made on-the-fly by the Dimension Routing overlays according to current environmental constraints received by synchronizing with the MD-CAN overlay.

To reach the destination a node forwards a received packet to neighbors along the path through the information written in the packet header. Due to the approach used for an e2e source-based route compilation in a multi-hop scenario, the risk does exist that a destination becomes unavailable before it is reached. In such case

packets are either discarded according to a Time to Live (TTL) counter added to all packets send on the network or by an intermediate node having more up-to-date knowledge of the destination availability. Finally, a scheme to prioritize between CR unlicensed users competing for free-spaces has to be devised. Such a solution can be based on cost (users who can afford the available spectrum hole to receive service) or power (users who can meet the required power outputs for the available spectrum hole to receive service) or a balanced approach (several unlicensed users can share the cost and the available spectrum).

## 4. Modeling Aspects

The CR space-dimension depicted as a MD-CAN dimension needs to be able to visualize three different categories: licensed users, unlicensed users and free space. These categories are highly important while executing one of the two primary activities performed by the CR management system, i.e., to compute an e2e routing path to the destination or to adapt the operational parameters of a CR user to retain service. For instance, an e2e routing path has to be computed for reaching the destination without interfering with the present licensed users. For doing this, the following pieces of information are needed to characterize the attributes of spectrum holes in the space dimension: Geographical location, First and second order statistics of the busy time. Another example is given by the power spectra of incoming RF stimuli. The following three categories can be distinguish when passively sensing the radio domain [13]:

- Black spaces, which are occupied by high-power local interferers and should be avoided in most cases due to the need for sophisticated spectral detection, analysis and decision. However, when the interferers are inactive these spaces can be used in an opportunistic manner.
- Gray spaces, which are partially occupied by low-power interferers and are potential candidates for use by the unlicensed users when the interferers are closed.
- White spaces, which are free of RF interferers and influenced by ambient noise only. These are the best candidates for use by the unlicensed users.

We regard a CRN frequency spectrum as consisting of a number of $N$ channels (synchronous slot structure), where each channel has the bandwidth $B_i$, for $i = 1, 2, ..., N$. These channels can be used either by licensed users or by unlicensed users if they are free. The unlicensed users use these channels in an opportunistic

way. Furthermore, we use the discrete-time Markov models to model different statistics related to these channels like, e.g., occupancy of the $N$th channel.

Available spectrum bands at a given node may be different from the spectrum bands at other nodes. They may even be flexible and not fixed, with unpredictable channel variations, meaning the spectrum bands could even differ drastically. The consequence is that such aspects should be captured in the process of modeling. Our employed model offers the possibility to do this, given the rather general data structure used for decision making and prediction. Other elements that can be captured and used in the process of CRD routing are, e.g., geographical distance between nodes, number of nodes in the network, number of available bands at a particular node, number of available bands in a network, maximum transmission power, spectral density at a transmitter node, minimum threshold of power spectral density to decode a transmission at a receiver node, link layer delay and others [14]. The diversity of the data collected, together with the hard requirements for fast routing decisions in the case of opportunistic routing leads to demands for efficient multidimensional data representation based, e.g., on multiple correspondence analysis (MCA) [15]. Tools like MCA, in combination with On Line Analytical Processing (OLAP), may help towards a better data summarizing, exploration and navigation in data cubes as well as detection of relevant information to help performing an efficient and fast opportunistic routing.

## 5. Routing Aspects

Routing in cognitive radio networks is challenging due to the large diversity of network configurations, diversity of network conditions as well as user activities [10]. The consequence is in form of large variations in the topology and connectivity map of the CRN as well as the available frequency bands of licensed users and their variations (e.g., power spectra). Furthermore, it is also important to consider possible spectrum handover when changing the frequency of operation. This in turn creates difficulties in finding an appropriate path between a particular source node and a particular destination node. Considering the activity and holding times of licensed users as a reference framing, we can partition the CRN routing as follows [1, 2, 10]:

- Static routing, used given that the holding times of the licensed bands are relatively static processes, i.e., with variations in the order of hours or days. Although this shows resemblance to multi-radio multi-channel mesh networks, there is the need to deal with transmissions over parallel channels as well as handling interference among licensed users and unlicensed users.

- Dynamic routing, licensed bands are intermittently available. Smaller temporal framings in the order of hours down to minutes, demands for more dynamic routing decisions. The main challenges are to find stable routing paths able to provide the expected e2e performance and to define appropriate routing metrics able to capture spectrum fluctuations. The routing algorithms used are those for wireless ad-hoc networks, combinations of static mesh routing and per-packet dynamic routing.
- Opportunistic (or highly dynamic) routing, the licensed bands show very high variations with temporal framings in the order of minutes down to seconds. Demands for routing algorithms able to take quick, local decisions. Solutions encompasses, per-packet dynamic routing over opportunistically available channels. Adapts the routing to the particular conditions existent at a specific time moment. This type of self-aware routing is also called cognitive routing protocols [16].

Our solution for CRN routing is a combined routing solution. The Decision Maker and Prediction overlay uses a static routing model to compile the e2e routing path locally. The computed path takes into account all available resources like, e.g., gathered network parameters from the MD-CAN overlay and the Historical Statistics database overlay, user preferences, environmental constraints and predictions of future network adaptations. Learning from previous experience, the predictions are performed with increased accuracy over time leading to further improved e2e path compilations that hold for the expected time period.

However, the "Dimension Routing" overlays responsible for the actual physical routing and adaptations can easily update and adapt the precompiled path if needed, according to the present network conditions and through the employment of dynamic and opportunistic routing. For every hop along the way the validity of the e2e path and destination is examined and reevaluated if necessary through a dynamic routing approach. All CRN users are responsible to keep their MD-CAN topology representation as accurate as possible. Obviously, peers located closer to the source of occurring changes are likely to be updated faster than those farther away. In case of peer failures a new path is computed by the intermediate peer, which currently holds the packet. The recompiled path stretches from the current location to the destination. In the case of destination node failure the packet is discarded. It is also necessary to consider the scenario where all original CR users in the CAN space dimension (from the original e2e path compiled by the "Decision Maker and Prediction" overlay) are still valid, though the destination has become unreachable due to adaptations in the other domains. In such a case an opportunistic adaptation is used for the remaining CR dimensions to adapt some parameters (e.g.,

in the frequency or power domains) and thus still be able to reach the destination compiled in the original path.

## 6. Conclusions

The paper has advanced a novel architectural solution for the management of cognitive radio networks. The architecture is based on the use of a middleware with a common set of Application Programming Interfaces (APIs), a number of overlay entities and a multi-dimensional routing. A short description of this architecture has been presented, with a particular focus on the research challenges associated with this architecture. Furthermore, two of the most important elements of the above-mentioned architecture are the Multi-Dimensional CAN (MD-CAN) and the CRN routing. The paper has particularly developed on the solutions adopted for these elements. Future work is three-fold. First of all, we intend to develop a solution for the Decision Maker and Prediction overlay. Further, we want to develop analytical and simulation models of this architecture and to do performance evaluation and validation. Finally, we want to implement this architecture.

## References

[1] Akyildiz I.A., Lee W-Y. and Chowdhury K.R., *Spectrum Management in Cognitive Radio Ad Hoc Networks* IEEE Network, pp. 6-12, July/August 2009

[2] Akyildiz I.A., Lee W-Y. and Chowdhury K.R., *CRAHNs: Cognitive Radio Ad Hoc Networks* Ad Hoc Networks Journal, Elsevier, 7(2009), 810-836, 2009

[3] Ilie D. and Erman D., *ROVER Middleware API Software Development Document*, BTH internal document, Karlskrona, Sweden, February 2007

[4] Stoica I., Morris R., Karger D., Kaashoek F. and Balakrishnan H., *Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications*, ACM SIGCOMM 2001, San Diego, USA, August 2001

[5] Ratnasamy S., Francis P., Handley M., Karp R. and Shenker S., *A Scalable Content-Addressable Network*, ACM SIGCOMM, San Diego, CA, USA, August 2001

[6] Loguinov D., Casas J. and Wang X., *Graph-Theoretic Analysis of Structured Peer-to-Peer Systems: Routing Distances and Fault Resilience*, IEEE/ACMTransactions on Networking, Vol. 13, No. 5, October 2005

[7] Popescu Adrian, Erman D., de Vogeleer K., Popescu Alex and Fiedler M., *ROMA: A Middleware Framework for Seamless Handover*, 2nd Euro-NF

Workshop on "Future Internet Architectures: New Trends in Service Architectures", Santander, Spain, June 2009

[8] ITU-T Rec. X.667 | ISO/IEC 9834-8, Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier components

[9] Yucek T. and Arslan H., *A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications*, IEEE Communications Surveys and Tutorials, Vol. 11, No. 1, First Quarter 2009

[10] Khalife H., Malouch N. and Fdida S., *Multihop Cognitive Radio Networks: To Route or Not to Route*, IEEE Network, pp. 20-25, July/August 2009

[11] Le B., Rondeau T.W. and Bostian C.W., *Cognitive Radio Realities*, Wireless Communications and Mobile Computing, Wiley InterScience, May 2007

[12] Bahl P., Chandra R., Moscibroda T., Murty R., and Welsh M., *White Space Networking with Wi-Fi like Connectivity*, ACM SIGCOMM, August 17-21 2009, Barcelona, Spain

[13] Haykin S., *Fundamental Issues in Cognitive Radio*, Cognitive Wireless Communication Networks, Hossain E. and Bhargava V.K., editors, Springer, 2007

[14] Shi Y. and Hou T., *Analytical Models for Multihop Cognitive Radio Networks*, Cognitive Radio Networks, Xiao Y. and Fei H., editors, CRC Press, 2009

[15] Messaoud R.B., Boussaid O. and Rabaseda S.L., *Efficient Multidimensional Data Representations Based on Multiple Correspondence Analysis*, 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, Pennsylvania, USA, August 2006

[16] Gelenbe E., *Steps Toward Self-Aware Networks*, Communications of the ACM, Vo. 52, No. 7, July 2009

[17] Engelbrecht A.P., *Fundamentals of Computational Swarm Intelligence*, Wiley, 2005

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 273–284

# Predictive Models for Seamless Mobility

SAID RUTABAYIRO NGOGA DAVID ERMAN ADRIAN POPESCU [a]

[a]Dept. of Communications and Computer Systems
School of Computing
Blekinge Institute of Technology
371 79 Karlskrona, Sweden

**Abstract:** User's location modeling and prediction is complex and a challenge for seamless mobility in heterogeneous networks. Location prediction is fundamental for pro-active actions to be able to take place. Markov models and information-theoretic techniques are appropriate to perform location prediction. The paper characterizes user's location as a discrete sequence. We survey and describe Markovian methods and information-theoretic techniques for location prediction.
**Keywords:** : Location prediction, Discrete Markov models

## 1. Introduction

Today, various wireless technologies and networks exist and capture different user's preferences with different services these technologies provide. The range of wireless access network technologies includes GSM, GPRS, UMTS, WiMax, WLAN (802.11 a/b/g/h) and Bluetooth. On the other hand, a great part of today's mobile terminals are already capable of having more than one interface active at the same time. Since the available wireless networks are complementary to each other, the vision of the Next Generation Networks (NGN) is to combine them over a common IP-based core network to support high usability (any system, any time, any where). This will empower mobile users to choose from their terminals which network interface to use in order to connect to the best available access network that fits their preferences. Key features are user friendliness and personalization as well as terminal device and network heterogeneity [1].

Existing solutions attempt to handle mobility at the link layer (L2) and the network layer (L3) with a particular focus on the reduction of handoff latency. The basic idea is to rely on the capability of L2 to monitor and to trigger changes

with regard to L2 or L1 conditions. This further assists IP in handover preparation and execution. However, mobility in heterogeneous networks is mainly user centric [1]. This calls for dynamic and adaptive mechanisms adapted to the current situation of users, which makes control and management difficult for operators. The IEEE 802.21 working group is working on the Media Independent Handover (MIH) standard to enhance the collaborative use of the information available at the mobile terminal and the network infrastructure. The primary focus is on the decision phase of handoffs.

In addition, QoS is an important part of distributed multimedia applications [2]. This calls for applications to specify their QoS requirements before any attempt for resource reservation. Guaranteed QoS requirements for the whole duration of the service delivery makes resource provisioning a complex task in traditional fixed Internet. Thus, the high heterogeneity supported in NGNs introduces additional complexities such as bandwidth fluctuations, temporary loss of connectivity when clients disconnect from one AP and connect to a new one (handoffs) [3].

Seamless mobility therefore requires continuous resource reservation and efficient context transfer for handover management as the mobile terminal moves. The work in [4] suggests to act pro-actively against handoffs as one way towards seamless mobility. Furthermore, due to the complexity of handoffs there is a need to consider solutions based on full context awareness implemented at L5 in the TCP/IP protocol stack [3]. In other words, the high heterogeneity supported in NGNs, along with their respective specific handoff related procedures, requires a complete visibility of all aspects related to the handover process before any attempt for handover management.

Similarly, Blekinge Institute of Technology (BTH) suggests an architectural solution based on middleware and overlays [1]. This is an architectural solution implemented at L5, with the objective to offer less dependence on physical parameters and more flexibility in the design of architectural solutions. By this, the convergence of different technologies is simplified. A number of research challenges have been identified by the authors and are being worked on. These regard SIP and delay measurements, security, quality of Experience (QoE) management, overlay routing, node positioning, mobility modeling and prediction, middleware and handover.

As a functional block within the BTH architecture, the “Mobility Modeling and Prediction” overlay has the main function to perform mobility modeling and prediction. Specifically, the “Node Positioning” overlay collects the mobile terminal location informations, and transforms them into GPS coordinates. Further, the “Mobility Modeling and Prediction” overlay relies on this location information to determine and to learn the user mobility behavior for the appropriate handoff pro-

cess to take place pro-actively. The user behavior includes user's location as well as user's preferences (applications or terminal devices), which can be observed in terms of requests.

Mobility prediction is fundamental for pro-active actions to be able to take place. We envision to develop mobility predictors that exploit the past user's behavior and deduce how the user will behave in the future. These will include on-line models as well as off-line models. Furthermore, mobility prediction will be dynamic with the prediction agent running on the network as well as on the mobile terminal. Previous work [4, 5, 6] has shown that Markovian models and information-theoretic techniques are appropriate for on-line location prediction in wireless and cellular networks. According to [7], we should envisage other prediction techniques that have been developed in other disciplines, such as computational biology, machine learning, and Wold Wide Web. In this paper we present an overview of the state-of-the-art techniques for location prediction in wireless and cellular networks. This paper focuses on the mobile terminal location, as an example of the mobile's context. We survey different models and algorithms to track a user's mobility by determining and predicting their location. We consider only infrastructure-based networks such that the user's location can be described in terms of the topology of the corresponding access infrastructure and not the actual location of mobile.

The rest of the paper is as follows. In section two the BTH architecture for seamless mobility is described. In section three mobility prediction techniques are presented. In section four we present the model for user mobility. In section five models for movement history are presented. Finally, section six concludes the paper.

## 2. Mobility prediction techniques

Mobility prediction is needed to estimate where and/or when the next handoff will occur. This can be done at the link layer (L2), network layer (L3), and application layer (L5) in a TCP/IP protocol stack [1]. This is achieved by monitoring the mobile, gathering related location information and inferring the associated model of the mobile motion. Mobility prediction techniques can be classified into:

- **User-Centric:** the mobile stores its most frequent path locally or gets them from a home repository and builds a model of its motion behavior [4].
- **AP-Centric:** prediction is performed locally at an access point (AP), which builds the model using the motion behavior of mobiles encountered in the past [4].

The concept behind all these techniques is that users' mobility present repetitive patterns, which can be observed, gathered and learned over time. These patterns are merely the favorite route of travel and the habitual residence time. Therefore, the underlying mobility model can be characterized as a stationary stochastic process whose output is a sequence of observable location information.

## 3. Mobility modeling

The mobility models reported in [8, 9, 10, 11, 12, 13] are based on the assumption that a user's location is a priori information, which allows for simplification and analysis. The problem however is that mobility is a stochastic process, which has to be learned. Thus we need a scenario to describe user's mobility.

Let us consider a mobile user walking down a street while connected to a wireless network. The mobile communicates with a given access point and can regularly measure some information directly related to the path it follows (e.g signal strength or GPS coordinates). By reporting these measurements at discrete time steps $t = 1, 2, 3, ...$ we get a sequence of observable $\mathcal{V} = V_1, V_2, V_3, ...$

Since we are interested in gathering location information, let us divide the zone near an AP in small areas $\vartheta = v_1, v_2, v_3, ...$ where $v_i$ is an index identifying a zone-Id. Therefore, at any time a mobile can be described as being in one of the area zones $v_i$, thus $\vartheta$ is a state space. In addition, the road layout in the area near the AP represents the way area zones are interconnected. While a mobile is moving, it collects $v_i$ at discrete time steps, thus any sequence $v_1, v_2, v_3, v_4, ...$ represents a movement history.

Here we are interested in matching the sequence of observable $\mathcal{V}$ and the movement history $\vartheta$ in order to build a model of the mobile motion. This is referred to as state-to-observation problem. It is the base of the learning process for Markovian models.

## 4. Models for movement history

### 4.1. Discrete Markov models

Discrete Markov models relies on the simple mapping for the state-to-observation problem. Thus, given a state space $\vartheta = v_1, v_2, ..., v_N$ of size $N$, a mobile undergoes a change of state according to a set of probabilities associated with the state. If the time associated with state changes is $t = 1, 2, 3, ..., N$, $V_t$ represent the actual state at time $t$. This mean, the movement history collected (L2 signals) correspond to the observable location information. In general, a full probabilistic description requires specification of the current state and all previous past

states. However the first order Markov chain specifies only the current state and its previous state [14]:

$$\begin{aligned} P[V_n = v_n \mid V_1 = v_1, ..., V_{n-1} = v_{n-1}] &= P[V_n = v_n \mid V_{n-1} = v_{n-1})] \\ &= P[V_t = v_j \mid V_{t-1} = v_i] \end{aligned} \quad (1)$$

For a finite, irreducible ergodic Markov chain, the limiting probabily that the process will be in state $j$ at time $n$, $\pi_j$, exists and is a unique non-negative solution of

$$\begin{aligned} \pi_j &= \sum_i \pi_i P_{ij}, \quad j \geq 0, \\ \sum_j \pi_j &= 1 \end{aligned} \quad (2)$$

Therefore, a complete specification of a first order Markov model requires specification of the model parameter $N$, the one-step transition propobility matrix $\mathbf{P}$ and the limiting probability vector $\mathbf{\Pi}$.

For the usage of Markov models, let us refer to the location tracking problem in cellular networks as mentioned in [5], where the movement history is recorded using time-movement based update scheme. The service area is composed of eight zones $a, b, c, d, e, f, g, h$ interconnected with highway roads. Thus all location information (i.e., the sequence of observable information) related to the path a mobile follows corresponds to the movement history.

Given a typical sample path, e.g., $\mathcal{V} = aaababbbbbaabccddcbaaaa$, we are required to specify $N$, $\mathbf{P}$ and $\mathbf{\Pi}$ in order to build a first order Markov model, which can be used either as a model of this particular motion process or a generator of location information (sequence of observable). According to [5], a simple count approach is used to get values for these parameters.

The first order Markov model is appropriate for memoryless movement modeling, but in general the mobile user travels with a destination in mind. This requires considering the favorite route profiles of a user for the design of the motion model.

### 4.2. LeZi-update scheme

The mere purpose of this algorithm is to manage the uncertainty associated with the current location of a mobile based on specification of all previous past locations [5]. By doing this, we can only predict the current location of a mobile with a high degree of accuracy given its route prolife.

In other words, given a motion process $\mathcal{V} = v_1 v_2 v_3 ...$, we need to build candidate models based on previous last visited locations. The *null-order* model gathers

| 0-context | 1-context | | 2-contexts | | |
|---|---|---|---|---|---|
| $a(10)$ | $a\|a(6)$ | $b\|c(1)$ | $a\|aa(3)$ | $a\|ba(2)$ | $a\|cb(1)$ |
| $b(8)$ | $b\|a(3)$ | $c\|c(1)$ | $b\|aa(2)$ | $b\|ba(1)$ | $d\|cc(1)$ |
| $c(3)$ | $a\|b(3)$ | $d\|c(1)$ | $a\|ab(1)$ | $a\|bb(1)$ | $a\|cd(1)$ |
| $d(2)$ | $b\|b(4)$ | $c\|d(1)$ | $b\|ab(1)$ | $b\|bb(3)$ | $b\|dc(1)$ |
| | $c\|b(1)$ | $d\|d(1)$ | $c\|ab(1)$ | $c\|bc(1)$ | $c\|dd(1)$ |

Table 1. Context locations and their frequencies for a sequence $aaababbbbbaabccddcbaaaa$ [14]

all the *0-context* (no previous location) with their respective frequencies. Then the *first-order model* gathers all the *1-context* (one previous location) with their respective frequencies, and so on until we reach the *k-order* model, which collects the highest meaningful context. Further, the *entropy rate* of the motion process along each candidate model is calculated until we reach the limiting value $H(\mathcal{V})$ of the *entropy rate*, if it exists. Results in [5] show that for a stationary stochastic process the limit $H(\mathcal{V})$ exists and corresponds to the *universal model*.

For example, let us consider a movement history at a given time, $\mathcal{V} = aaababbbbbaabccddcbaaaa$, captured using L2 signals (see Fig.2). Under the concept of Markov model, candidate motion models of the movement history might be constructed by specifying, for the current location, the $k$-context locations $(k = 0, 1, 2, ...)$. These are all the past locations of a user conditioned on the current location. Thus, for a sequence $aaababbbbbaabccddcbaaaa$, all the 0-context, 1-context, and 2-context along with their respective frequencies of occurrence are enumerated in Table.1. With reference to the illustration in [5], the entropy rate of the movement history, $\mathcal{V} = aaababbbbbaabccddcbaaaa$, are respectively: $H(\mathcal{V}|\text{ 0-order model}) = 1.742$, $H(\mathcal{V}|\text{ 1-order model}) = 1.182$ and $H(\mathcal{V})|\text{ 2-order model}) = 1.21$. Intuitively, the highest order model is directly linked to the highest meaningful context. This corresponds to the universal model.

Again, under a *first order Markov Chain* the mobile terminal maintains a cache scheme for a user's location until the next update. By reporting this, the system uploads the new location and recomputes location statistics. The *universal model* relies on specification of the highest meaningful context. Therefore, the mobile is needed to maintain a *dictionary-like* scheme of route profile. At each location update, it reports a *phrase-like* message as a new route explored. The system, on its turn retrieves the *phrase-like* message (new route), uploads the mobile's route *dictionary*, and estimates all predictable routes untill the next location update.

Thus, the degree of location uncertainty lies between two consecutive message updates. A good solution to reduce the number of updates is that upon detection of

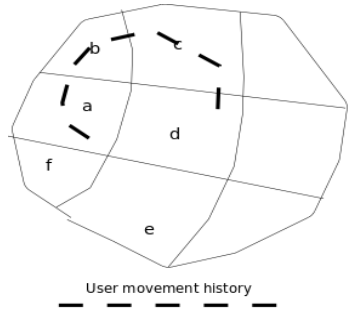



Fig. 1. A user movement history at a given time

the same message, to delay message update until a new message arrives. According to [7], this is what achieves the Lempel-Ziv-78 scheme, a parsing technique for data compression. It is referred to as the *prefix-matching technique*.

The concept behind this formulation is that the movement history is compressible. The LeZi-update scheme, as reported in [5], is an interlaced learning and coding process that:

1. Uses the *prefix-matching technique*, during the learning process, to generate *phrase-like messages* of variable length.
2. Applies, during the encoding process, the *variable-to-fixed length coding scheme* and generates *fixed length dictionary indices* representing context statistics.
3. Stores these *fixed length dictionary indices* in a *symbol-wise model* that can be represented in a tree form.

With reference to illustrations presented in [5, 6], let us consider a typical movement history, e.g., $\mathcal{V} = aaababbbbbaabccddcbaaaa$, parsed as distinct *phrase-like* messages $a, aa, b, ab, bb, bba, abc, c, d, dc, baa, aaa$, by using a *prefix-matching technique*. Then, by applying the *variable-to-fixed length coding scheme*, each message is explored and symbol contexts together with their respective frequency of occurrence are stored in a *symbolwise context model*, which can be implemented by a tree (see Fig.2). Furthermore, context statistics are updated for each message entry in the tree using the following technique: *increment context statistics for every prefix of every suffix of the phrase*.

Therefore, as larger and larger messages arrives, the symbolwise context model, maintained in every mobile's location database, will contain sufficient information regarding user mobility profile. Further, the system uses this information to estimate the location of a mobile terminal until the next update message arrives, and

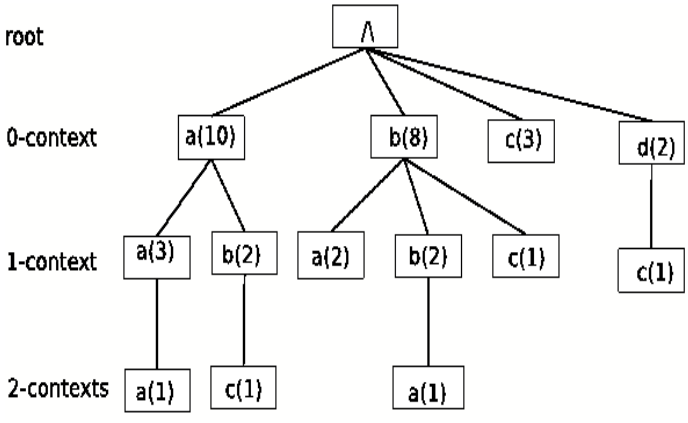



Fig. 2: **Symbolwise context model:** a contex represents a path from root to any node; a sub-tree rooted at a node represents a corresponding conditioned context; a path from root to leaves represents the largest context [14]

this time period corresponds to the location uncertainty period. This gives insight to the predictability of higher order Markov models.

Another point to consider is with respect to specification of probability measures estimated based on context statistics stored in the tree form model. Since the system holds the mobile's route profile until the last update message (current context) in the form of all previous past locations, we are interested in estimating the next update message to be sent by the mobile. This is the condition probability distribution given the current context. This is what the *prediction by partial match* (PPM) techniques achieves for text compression [7]. However, due to space limitation, the work in [5] suggests the *blending-PPM* that work with total probability distributions instead of relying on conditional probability distributions.

### 4.3. Hidden Markov models

So far, we have mentioned the state-to-observation matching problem, which is the base of the learning process for the model of the mobile motion. By using the concept of Markov models, the movement history collected using L2 signals corresponds to the observable location information. In the following we extend this concept to include the case where the movement history information is absent, but can be generated through a set of stochastic processes. Thus, the observable location information is a probabilistic function of the physical location of a mobile. The physical locations of the mobile are hidden to the system, therefore the resulting model of the mobile motion is a doubly embedded stochastic processes called Hidden Markov model [15].

Typically for a L2/L3 handover case, characterization of user's mobility as a stochastic process relies on the ability of a mobile to monitor the user's motion behavior within the topology of the corresponding access infrastructure. Thus, the mobile collects some piece of information associated to its location along the route it follows. These informations can be very precise (GPS positions), less informative (signal strength), or very fragmentary (past pass by AP, time of day, or cell residence time) [4]. By reporting these informations to a handover management functional module such as L3 Mobile IP (MIP), the type of handover (horizontal or vertical) as well as the time to perform it can be determined. This is done based on knowledge of the access points surroundings. By using the concept of Hidden Markov models, on one side a prediction scheme can easily adapt any network, and on the other side it can accommodate any type of information emitted by the mobile terminal.

Therefore, a Hidden Markov model is characterized by:

- A hidden state space of size $N$, $\vartheta = \{v_1, v_2, ..., v_N\}$, where the state at time $t$ is denoted $q_t$. The $v_i$(s) are the more likely physical locations of a user, a mobile terminal can collect.
- A set of observable location information per state of size $M$, $\mathcal{V} = \{V_1, V_2, ..., V_M\}$. The $V_i$(s) are any type of location information associated with a user location, a mobile terminal can send to the system.
- The single step state transition distribution, $\mathbf{P} = \{P_{ij}\}$

$$P_{ij} = P[V_t = v_j \mid V_{t-1} = v_i]. \tag{3}$$

- The observable location information distribution in state $j$, $\mathbf{B} = \{b_j(V_k)\}$

$$b_j(V_k) = P[V_k \text{ at } t | q_t = v_j] \qquad \begin{matrix} 1 \leq j \leq N \\ 1 \leq k \leq M \end{matrix} \tag{4}$$

- The initial state probability vector, $\mathbf{\Pi} = \{\pi_i\}$

$$\pi_i = P[q_i = v_i] \tag{5}$$

Thus, given values for the parameters $\mathbf{N}, \mathbf{M}, \mathbf{P}, \mathbf{B}$ and $\mathbf{\Pi}$, a Hidden Markov model can be used as a generator or a model of a sequence of observable location information, $\mathcal{V} = V_1 V_2 V_3 ...$ For the usage of Hidden Markov Model, the work of [4] suggests a framework based on artificial intelligence technique that links directly a user movement to the handover mechanism.

## 5. Conclusion

Seamless mobility in heterogeneous networks requires continuous resource reservation and efficient context transfer for handover management as the mobile terminal moves. The BTH architectural solution for seamless mobility, as reported in [1], offers less dependence on physical parameters and more flexibility in the design of architectural solutions. Based on this, we envision to develop predictive models that exploit a user's previous behavior and deduce the user's future behavior. This paper focuses on user's location prediction as a fundamental for pro-active actions to be able to take place. We describe the user's locations as a discrete sequence. We present an overview of Markov models and information-theoretic techniques for location prediction.

Future work includes using simulation to perform comparative studies in order to evaluate various mobility prediction techniques in different mobility scenarios.

## References

[1] A. Popescu, D. Ilie, D. Erman, M. Fiedler, A. Popescu, K. De Vogeleer: An Application Layer Architecture for Seamless Roaming, *Sixth International Conference on Wireless On-Demand Network Systems and Services (WONS 2009)*, Snowbird, Utah, USA, 2009.

[2] J. Jin, K. Nahrstedt: QoS Specification Languages for Distributed Multimedia Applications: A survey and Taxonomy, *IEEE MultiMedia*, 2004, vol. 11, no 3, pp. 74-87.

[3] P. Bellavista , A. Corradi, L. Foschini: Context-aware Handoff Middleware for Transparent Service Continuity in Wireless Networks, *Pervasive and Mobile Computing*, Elsevier, 2007, vol. 3, pp. 439-466.

[4] J.M. Francois, G. Leduc: Performing and Making Use of Mobility Prediction, *PhD Thesis*, Univesite de Liege, 2007.

[5] A. Bhattacharya, S.K. Das: LeZi-Update An Information-theoretic Framework for Personal Mobility Tracking in PCS Networks, *Springer Wireless Networks*, 2002, vol. 8, no. 2-3, pp. 121–135.

[6] F. Yu, V.C.M. Leung: Mobility-Based Predictive Call Admission Control and Bandwidth Reservation in Wireless Cellular Networks, *IEEE INFOCOM 2001*, 2001.

[7] D. Katsaros, Y. Manolopoulos: Prediction in wireless networks by Markov chains, *IEEE Wireless Communications*, 2009, vol. 16, no. 2, pp. 56-64.

[8] R. Thomas, H. Gilbert, G. Mazziotto: Infuence of the Moving of Mobile Stations on the Performance of a Radio Mobile Cellular Network, *Third Nordic Seminar on Digital Land Mobile Radio Communication*, 1988, pp. 1-9.

[9] I. Seskar, S.V. Maric, J. Holtzam, J. Wasserman: Rate of Location Area Updates in Cellular Systems, *IEEE Vehicular Technology Conference*, 1992, vol. 2, pp. 694–697.

[10] T. Camp, J. Boleng, V. Davies: A Survey of Mobility Models for Ad Hoc Network Research Wireless Communications and Mobile Computing (WCMC), *Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2002, vol. 2, pp.483-502.

[11] R. A. Guerin: Channel Occupancy Time Distribution in a Cellular Radio System, *IEEE Transanctions on Vehicular Technology*, 1987, vol. VT-35,no. 3, pp. 89-99.

[12] D. Hong, S. Rappaport: Traffic Model and Performance Analysis for Cellular Mobile Radio Telephone Systems with Prioritized and Nonprioritized Handoff Procedures, *IEEE Transanctions on Vehicular Technology*, 1986, vol. 35, no. 3, pp. 77-92.

[13] M.M. Zonoozi, P. Dassanayake: User Mobility Modelling and Characterization of Mobility Patterns, *IEEE Journal on Selected area in Communications*, 1997, vol. 35, no. 7, pp. 1239-1252.

[14] A. Bar-Noy, I. Kessler, M. Sidi: Mobile users: to update or not to upadate? *ACM Wireless Networks*, 1995, vol. 1, pp. 175-185.

[15] L. Rabiner: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Morgan Kaufmann Readings in Speech Recognition*, 1990, vol. 53, no. 3, pp. 267-296.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 285 – 296

# Evaluation of possible applications of dynamic routing protocols for load balancing in computer networks

KRZYSZTOF ZAJDA

Institute of Computer Science
Silesian Technical University
*krzysztof.zajda@student.polsl.pl*

**Abstract**: The load balancing issue in computer networks has been known for many years. It can be done in many ways: duplicated routes can be rigidly configured via the use of costly hardware load balancers or use dynamic routing. The subject of this research was made using possibilities of computer networks with dynamic routing for load balancing. Three dynamic routing protocols were considered: RIP, OSPF, EIGRP. The research was conducted for two simultaneous links with the same or different bandwidths. The research was carried out to confirm, which protocols to use, and which are the most efficient for load balancing for links with equal and unequal costs. It was also a trial to answer the question, what kind of link parameters (bandwidth, delay) have an impact on load balancing.

**Keywords**: Load balancing, dynamic routing protocols.

## 1. Introduction

Load balancing is a router ability to send packets to the destination IP address, using more than one route, Fig. 1. Routes can be set statically or dynamically through protocols such as RIP, OSPF, EIGRP.

If the router supports more than one dynamic routing protocol, then the concept of administrative distance is introduced. It is a number representing a level of trust in relation to information source about the route. The principle is quite simple, the smaller administrative distance is (smaller number), the more trustworthy the data source about the route is. If the router recognizes some routes to a destination network (the routes detailed in the routing table), it choose the one which has the smallest administrative distance. Sometimes there is a situation, that the router must choose one route from among several which have the same administrative distance. In this case the router chooses the route with the smallest cost which is calculated based on the routing protocol metric. If the router knows some routes with the same administrative distance and the same

cost than it is possible a transmission with load balancing. Mostly routing protocols default install maximum four equivalent routes.

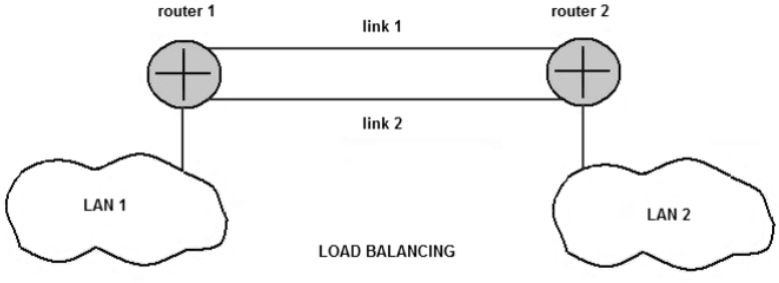



Fig. 1. A general scheme of load balancing

## 2. RIP protocol

A type of dynamic routing protocol distance vector. The routing algorithm of the distance vector periodically copies a routing table from one router to another. These regular updates between routers communicate about topology changes. Distance vector algorithms are also known as Bellman-Ford algorithms. Each router gets the routing table from the directly connected neighbor. Algorithm can store the value of the distance, creating and maintaining a database of network topology. However, these algorithms do not allow the router to became familiar with the exact network topology, each router "sees" only directly connected neighbors.

### 2.1 RIP metric.

The measure used by the RIP protocol to measure the distance between the source and destination is hop counting. Each hop on the way from source to its destination is usually assigned one value. When the router receives the routing table update that contains a new or changed an entry for the destination in the network, it adds 1 to the measurement values indicated in the update and enters a change in a routing table. Next hop address is the IP address of sender. RIP protocol reduces the number of hops that may occur between the source and destination, preventing a transmission data stream without the end in a the loop. The maximum number of hops in the path is 15. If the router accept routing update that include new or changes entry, and if the measure will be increased in one, and will be exceeded 15 hops, such destination on the network is considered as unavailable.

### 2.2 RIP load balancing.

This type of load balancing allows the router to select more than one route to the destination IP address. RIP protocol can set a maximum of six simultaneous routes for the packages, but the default is set four paths. The RIP protocol as a metric uses number of hops and based on that, calculates the best routes for packets. Therefore it is possible to route using symmetrical and asymmetrical links that have different bandwidths, however the total bandwidth is shared, in proportion to the number of links, Fig. 2 (if there are 2 links, the distribution is 50%/50%). This is not an effective solution, because it does not take into consideration the fact that one link may have better parameters than second link.

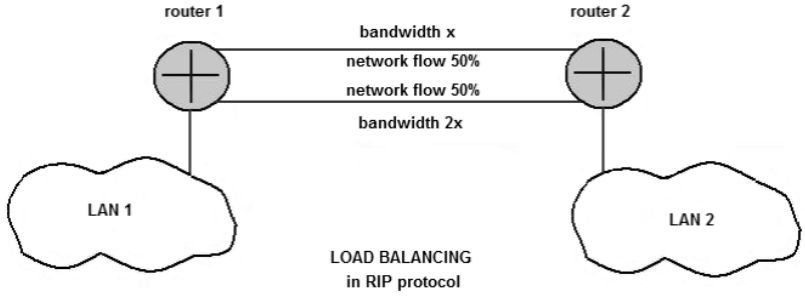



Fig. 2. Load balancing in RIP protocol.

## 3. OSPF protocol.

OSPF (Open Shortest Path First) dynamic routing protocol is designed to work with IP based networks. It is an internal protocol within a single Autonomous System (AS). Its rise has been forced due to underperformance of the RIP protocol, which could not satisfactorily handle the ever growing IP supporting networks. Action protocol is based on the algorithm SFP (Shortest Path First), sometimes known as Dijkstra's algorithm. OSPF is a link state protocol type, it means that every short time is sent notification (LSA Link State Advertisement) to all routers located in the autonomous zone. LSA contains information about available interfaces, metrics and other network parameters. Based on this information, routers calculate efficient traffic routes using the SPF algorithm. Each router periodically sends out LSA packets containing information about availability or change of router status. Therefore, fast changes are detected in the network topology which enables efficient adaptation of

routing to the existing conditions. Sending the LSA gives the possibility to build a topology of network. Based on this, using the SPF algorithm, routes tables are created.

### 3.1 OSPF metric.

OSPF supports a system of penalty tables. Networks systems assign a metric depending on the setting of OSPF-enabled applications and tools for configuration options, related to network interfaces and connections. Most often these are: the link bandwidth, the delay on the link and the individual preferences of the network system administrator. In practice however, the value metric depends only on the bandwidth, according to the formula:

$$metric=10^8/bandwidth \tag{1}$$

### 3.2 OSPF load balancing.

If to the network lead two or more routes with equal metric value, the protocol shared traffic between 4 routes (maximum 6 routes). Unfortunately, when to the network lead two or more routes with different metric values, then all traffic goes through the route with the lowest metric value. Therefore, the OSPF load balancing is possible, but only for routes with identical parameters, Fig. 3.

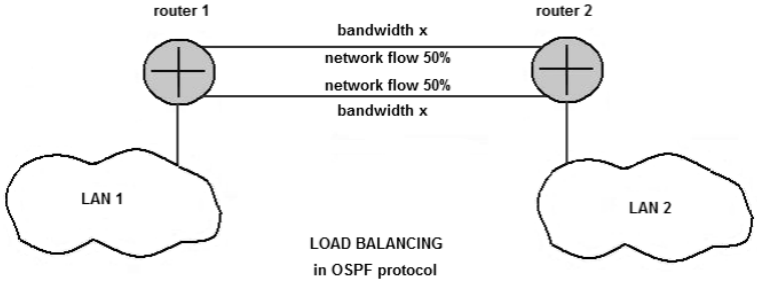



Fig. 3. Load balancing in OSPF protocol.

In specific cases, where we want to force a transmission on routes with lower bandwidth, we can manually modify the OSPF metric.

## 4. EIGRP protocol.

EIGRP (Enhanced Interior Gateway Routing Protocol) was introduced by Cisco as a scalable and improved version of its own routing protocol running

based on distance vector – IGRP. EIGRP is often described as a hybrid protocol, because it combines the best features of routing algorithms using distance vector and link state. EIGRP protocol uses typical functions for link state routing protocols and some key features of OSPF, such as partial updates and discovering of neighboring devices. One important advantage of EIGRP protocol is DUAL (Diffusing Update Algorithm) algorithm, which enables the identification and rejection of looped routes, and allows to find alternative routes without waiting for update from other routers. EIGRP does not send periodic updates. Instead, it refreshes the relationship with nearby neighboring routers through sending small packets and sending partial updates, when it detects changes in network topology. Therefore it consumes much less bandwidth than distance vector protocol (RIP).

### 4.1 EIGRP metric.

Its value depends on bandwidth and delay, although it is possible to extend: the average load, reliability and MTU (Maximum Transmission Unit). Link metric value is calculated by the formula:

$$metric = 256*[K1*Bandwith + (K2*Bandwith)/(256 - Load) + K3*Delay] + [K5/\mathrm{Re}liability + K4)] \quad (2)$$

Where the parameters K1 … K5 take the value 0 or 1 depending on whether the characterization is used and they correspond to the characteristics of: *bandwidth, load, delay, reliability*, MTU. The value *bandwidth* means metric of the lowest bandwidth link, and is calculated from the formula:

$$bandwidth = [10000000/(lowestbandwidthinkbps)]* 256 \quad (3)$$

$$delay = (interfacedelay/10) *256 \quad (4)$$

Very often we assumed K1=K3=1 and K2=K4=K5=0 to simplify some calculation, then the metric is calculated:

$$metric = 256*( bandwidth + delay) \quad (5)$$

### 4.2 EIGRP load balancing.

If the EIGRP protocol finds some routes with the same cost it will automatically start to distribute the traffic between them. Using the *variance* command it can also distribute traffic to link with a higher cost. *Multiplier*, in this case must be between 1 to 128 and the default is 1. This means only the

distribution of traffic between links has the same cost, the cost is directly connected with the metric. If we have links with different bandwidth, with different costs, then a route metric with a lower cost must be multiplied by the *multiplier* value and we obtain a cost, which is the same as the cost of link with a higher cost. Therefore we obtain (though extortion) two links with the same cost and a possible load balancing mechanism, Fig. 4.

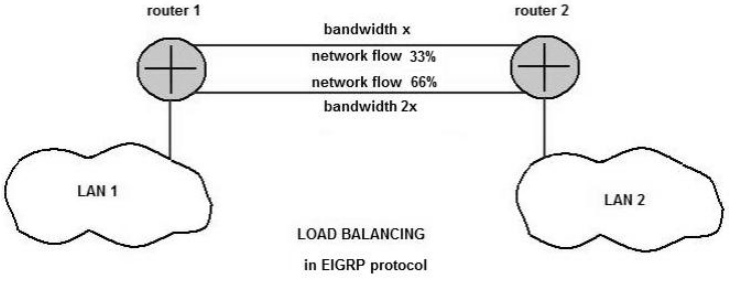



Fig. 4. Load balancing in EIGRP protocol.

This function seems to be the best, if we consider load balancing with different costs routes, however it has one major defect, EIGRP protocol is a commercial solution used only in Cisco routers, which obviously limits its application.

## 5. Tabular assess possibilities of load balancing in dynamic routing protocols.

Tab. 2 consists a theoretical comparison of load balancing possibilities in dynamic routing protocols.

| | **Load balancing** | | |
|---|---|---|---|
| | **RIP** | **OSPF** | **EIGRP** |
| The same cost links | Yes | Yes | Yes |
| Different cost links, traffic shared 50%/50% | Yes | No | No |
| Different cost links, traffic proportionally shared | No | No | Yes |
| Limitation of OS platform | No | No | Only Cisco |
| Network scalability | Small network | Medium network | Medium network |

Tab. 2. Comparison of load balancing possibilities

## 6. The research bench.

The research was shared into two parts. In the first part of research where RIP and OSPF protocols were studied, the research bench was built on the linux operating system in which the ZEBRA system was installed. Zebra allows dynamic routing protocols RIP and OSPF to run, Fig. 5. The research used two different 4 Mbs links, in which, during the tests it reduced bandwidth, if it was necessary, using network limiters (CBQ packet in linux). In the second part of research, load balancing was studied in EIGRP protocol. Because EIGRP protocol can run only on Cisco routers, tests were carried out using Cisco 2801 routers, Fig. 8. In both cases, the measurement of bandwidth and load were done by MRTG software.

## 7. Load balancing research in RIP protocol.

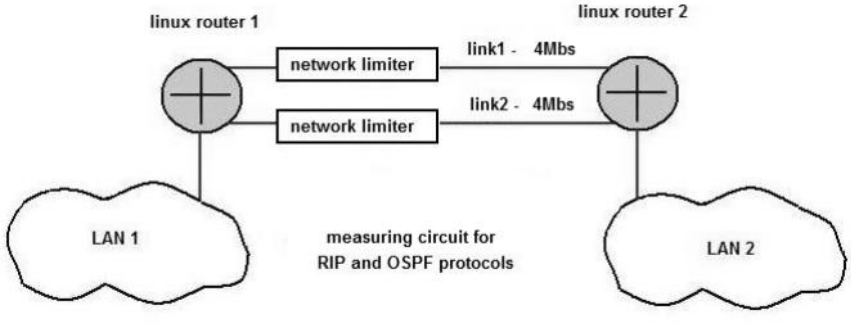



Fig. 5. Measuring circuit for testing load balancing in RIP and OSPF protocols

The routers had the software in which the RIP dynamic routing was working. Then begins transmission of a large number of files, about 1000 between LAN1 and LAN2. Using network limiters a different network bandwidth was set on link1 and link2. It was observed that load balancing appeared and traffic is always distibuted 50%/50%, regardless of bandwidth, limited to the link with lower bandwidth. It was also noticed that if one of the link has very small bandwidth, compared to the second link bandwidth:

$$\textit{bandwidth link1/bandwidth link2} = \textit{about 25} \tag{6}$$

Load balancing does not occur, and the transmission goes through larger bandwidth link.

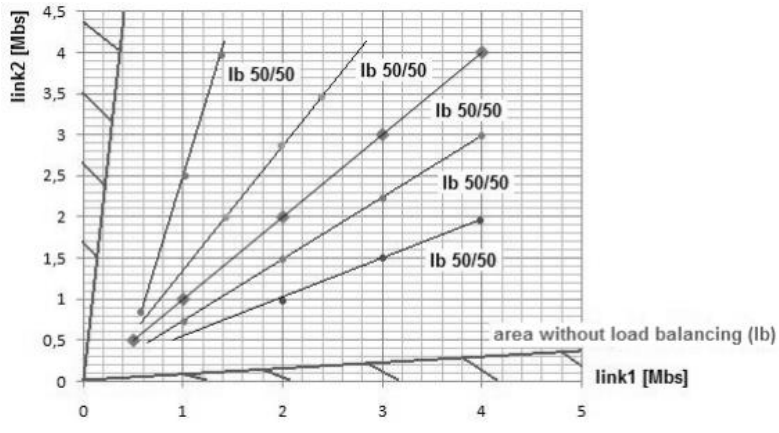



Fig. 6. Distribution of transmission between link1 and link2 with different bandwidth for RIP protocol

The research shows that load balancing in dynamic routing protocol RIP is possible and transmission is always divided in half, regardless of whether links have different or the same bandwidth as detailed in Fig. 6. It also appears that the large difference in links bandwidth causes that load balancing does not occur. In small networks, where the RIP protocol is used, you can use the alternative link (backup) and start the load balancing in a very easy way.

## 8. Load balancing research in OSPF protocol.

The routers had the software in which the OSPF dynamic routing was working. Then began transmission large number of files, about 1000 between LAN1 and LAN2. Using network limiters set a different network bandwidth on link1 and link2. It turned out, that load balancing is possible only for routes with the same bandwidth that are equal 50%/50%.

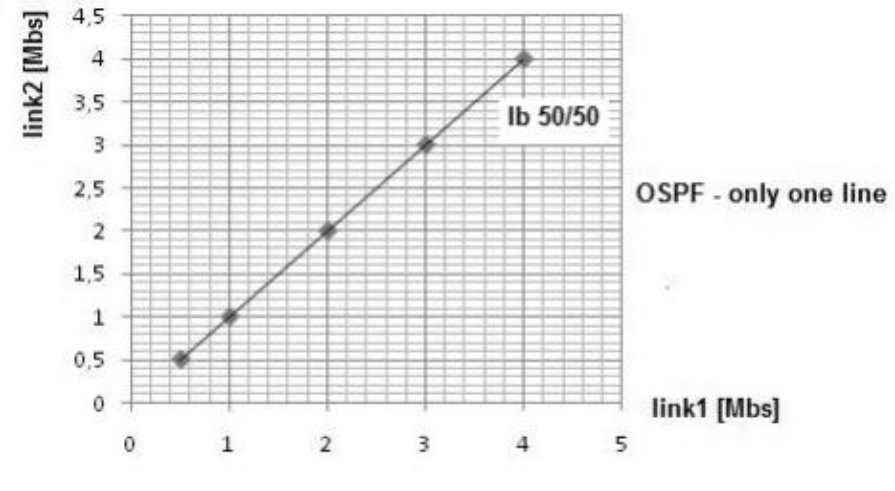



Fig. 7. Distribution of transmission between link1 and link2, for OSPF protocol

OSPF is very flexible, if we take into account, the hardware platform and operating system, but shows small possibilities in load balancing, Fig. 7. In principle, it is possible only for routes with identical parameters, then the transmission is divided in half.

## 9. Load balancing research in EIGRP protocol.

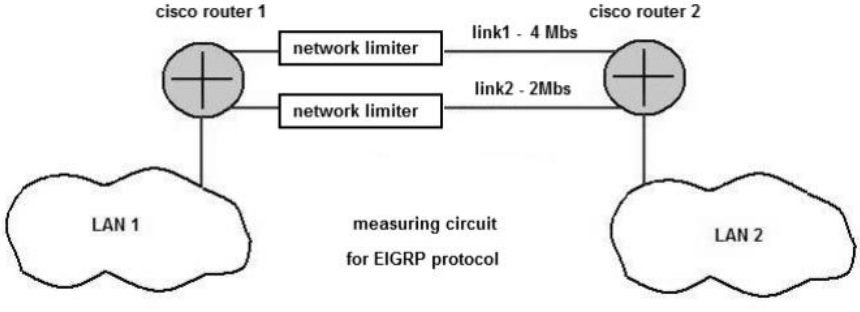



Fig. 8. Measuring circuit for testing load balancing in EIGRP protocol

On Cisco 2801 routers run the EIGRP dynamic routing, which enables running load balancing, Fig. 8. Bandwidth link1 was 4 Mbs, but bandwidth link2 was limited to 2 Mbs. For link2 *variance* value was set to 2. This configuration

allowes load balancing on 4Mbs and 2Mbs links. Then began the transmission of a large number of files, about 1000 between LAN1 and LAN2, using 2Mbs and 4Mbs links. In such settings, load balancing was not present, MRTG measurement showed transmission only on 4 Mbs link. By increasing a variance to 3 transmission occurred on both links (Fig. 10) and distribution was in proportion: 35% 2Mbs link and 65% 4Mbs link. You can guess that, when the variance was set to 2, a problem with various dalays on link1 and link2 appeared, which resulted in another metric in the calculation by the router. A few tests were performed by limitings the bandwidth on link1 and link2. By empirical selection of value *variance*, load balancing occurred and a traffic was proportionally distributed on two links, Fig. 9.

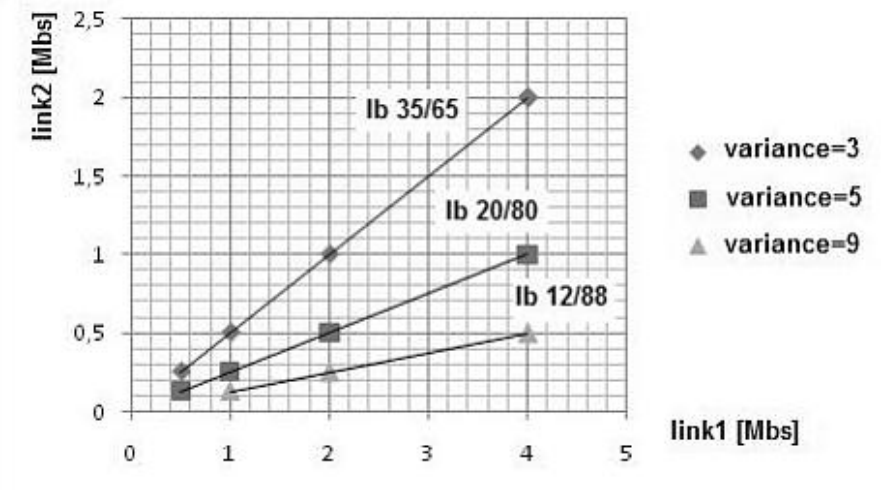



Fig. 9. Distribution of transmission between link1 and link2 with different bandwidth for EIGRP protocol

The research shows that load balancing in dynamic routing protocol EIGRP is possible and that traffic is distributed proportionally to each links bandwidth. The downside, which limits this solution is that the EIGRP protocol can only run on one hardware platform (Cisco), which is very expensive.

## 10. Conclusions.

The research have confirmed the theoretical function (chapter 5) regarding the possibilities of using the load balancing in dynamic routing protocols. After each experiment are summarized, in detail, the practical possibilities (chapter 7,8,9). Generally speaking, studies confirmed the theoretical possibilities of dynamic routing protocols with regard to load balancing using. The EIGRP protocol has the greatest possibilities because, load ballancing can be run on various cost links, but works only on one platform, Cisco. RIP and OSPF protocols slightly worse deal with this subject but the load balancing is possible. In studies tried to also answer the question, whether the link parameters (bandwidth, delay) affect the load balancing. The bandwidth parameter has a significant impact on the load balancing in all protocols, while the delay parameter was important in the EIGRP protocol. Although OSPF has the smallest possibilities when it comes to load balancing, it has tremendous flexibility when it comes to platform (different type of operating systems) and is widely used. You could focus on the topic of load balancing in OSPF protocol for different bandwidth links (different cost links), which could be the subject of futher research.

# References

[1] Malhotra R.: IP routing, O'Reilly Media, 2002

[2] Slattery T.: Advanced IP Routing in Cisco Networks, Paperback, 2000

[3] Wright R.: IP routing primer, Cisco Press, 1998

[4] Moy T.: *OSPF Anatomy of an Internet Routing Protocol*, Addison-Wesley, 1998

[5] Breyer R., Rileyi S.: Switched, Fast i Gigabit Ethernet, Helion, 2001.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 297–310

# Effective-availability methods for point-to-point blocking probability in switching networks with BPP traffic and bandwidth reservation

MARIUSZ GŁĄBOWSKI[a] MACIEJ SOBIERAJ[a]

[a]Chair of Communication and Computer Networks
Poznan University of Technology
ul. Polanka 3, 60-965 Poznan, Poland
mariusz.glabowski@et.put.poznan.pl

**Abstract:** The paper presents the analytical methods for determination of traffic characteristics in multistage switching networks servicing multirate BPP (Binomial-Poisson-Pascal) traffic. The proposed methods allow us to determine point-to-point blocking probability in the switching networks in which reservation mechanisms are implemented. The basis of the proposed methods are the effective availability concept and the models of interstage and outgoing links. The results of analytical calculations of blocking probability in the switching networks with BPP traffic and bandwidth reservation are compared with the simulation results.

**Keywords:** traffic characteristics, switching networks, BPP traffic.

## 1. Introduction

Together with increasing popularity of voice, audio, video, TV and gaming applications the infrastructure supporting these applications is being deployed very fast. Simultaneously we can observe increasing effort in elaboration of analytical methods of modeling and dimensioning the multiservice systems, and, in particular multiservice multistage switching networks [1].

Modern networks should ensure high service quality for a wide range of service classes with different Quality of Service (QoS) requirements as well as high usage of network resources. One of possible ways of fulfilling these requirements is a satisfactory choice of admission control function which is responsible for the most favorable access control strategy for different calls. One of the possible strategies of admission control function is bandwidth reservation. The reservation mech-

anisms allow us to obtain unrestricted relations between the blocking probabilities of individual traffic classes.

The multiservice switching networks with bandwidth reservation and Poisson call streams were the subject of many analysis [2–4]. However, recently we can notice increasing interest in elaborating effective methods of analysis of multi-service systems in which traffic streams of each class are generated by a finite number of sources. This is due to the fact that in modern networks the ratio of source population and capacity of a system is often limited and the value of traffic load offered by calls of particular classes is dependent on the number of occupied bandwidth units in the group, i.e. on the number of in-service/idle traffic sources. In such systems the arrival process is modeled by Binomial process or Pascal process.

The first method devoted to modeling the switching networks with BPP traffic and bandwidth reservation, the so-called recurrent method, was published in [5]. The method is based on the simplified algorithm of calculation of the effective-availability parameter. The results presented in [5] were limited to a single reservation algorithm. In this paper two methods of blocking probability calculation in the switching networks which are offered multi-service traffic streams generated by Binomial (Engset)–Poisson (Erlang)–Pascal traffic sources has been proposed. The proposed methods are directly based on the effective-availability methods [6] and allow us to model the switching networks with different reservation algorithms. The calculations involve determination of occupancy distributions in interstage links as well as in the outgoing links. These distributions are calculated by means of the full-availability group model and the limited-availability group model, respectively.

The further part of this paper is organized as follows. In Section 2 the model of multistage switching network is presented. In Section 3 we describe the models of interstage links and outgoing links of switching networks, i.e. the full-availability group model and the limited-availability group model. Section 4 presents the concept of effective availability. In Section 5 the reservation algorithms are presented. In Section 6 two analytical methods for determination of the blocking probability in switching networks with BPP traffic and bandwidth reservation are proposed. The analytical results of blocking probability are compared with the simulation results in Section 7. Section 8 concludes the paper.

## 2. Model of switching network

Let us consider a switching network with multirate traffic (Fig. 1). Let us assume that each of the inter-stage links has the capacity equal to $f$ BBUs[1] and that outgoing transmission links create link groups called directions. One of typical methods for realization outgoing directions was presented in Fig. 1. This figure shows that each direction $s$ has one outgoing link $s$ from each the last-stage switch. We assume that interstage links can be modeled by the full-availability group and that the outgoing directions can be modeled by the limited-availability group.

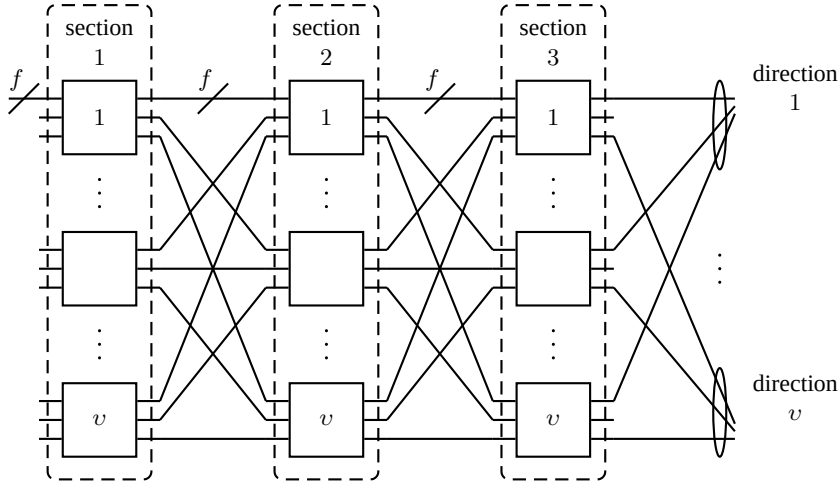



Fig. 1. A three-stage switching network

Let us consider a switching network with point-to-point section. Process of setting up a new connection in switching network is as follows. First, the control device of the switching network determines the first stage switch, on the incoming link of which a class $c$ call appears. Next, the control system finds the last-stage switch having a free outgoing link, which has at least $t_c$ free BBUs in required direction. Next, the control device tries to find a connection path between the determined first-stage and the last-stage switch. The existence of such connection path causes realization of the connection. In opposite case, when the connection path cannot be found the call is lost due to the internal blocking. If each last-stage switch does not have $t_c$ free BBUs in the required direction, the call is lost as a result of the external blocking.

[1]BBU is the greatest common divisor of resources required by call streams offered to the system [7–9].

## 3. Modeling links in switching networks

### 3.1. Limited-availability group with bandwidth reservation

Let us consider the limited-availability group (LAG) with a reservation mechanism, to which three types of traffic streams are offered: $M_1$ Erlang traffic streams (Poisson), $M_2$ Engset traffic streams (Binomial) and $M_3$ Pascal traffic streams [5, 10]. The limited-availability group is a model of the system composed of $v$ separated transmission links. Each link has capacity equals to $f$ BBUs. Thus, the total capacity of the system is equal to $V = vf$. The system services a call only when this call can be entirely carried by the resources of an arbitrary single link and when the number of free BBUs is equal or greater than the reservation space $R = V - Q$. The reservation space $R$ is equal to substract of total capacity of system $V$ and reservation threshold $Q$. It is, therefore, the example of the system with state-dependent admission process, in which the state-dependence results from the structure of the group and from the reservation mechanism applied.

The mean value of traffic offered by class $i$ Erlang stream can be described by the following formula:

$$A_i = \lambda_i/\mu_i, \tag{1}$$

where $\lambda_i$ is the arrival rate of new class $i$ calls and $\mu_i^{-1}$ is the mean holding (service) time of class $i$ calls. Taking into account the dependence of Engset and Pascal traffic streams on the state of the system (defined by the number of busy BBUs) we can express the mean traffic $A_j(n)$ offered by class $j$ Engset stream and the mean traffic $A_k(n)$ offered by class $k$ Pascal stream with the help of the transition coefficients $\sigma_{j,T}(n)$ and $\sigma_{k,T}(n)$:

$$A_j(n) = N_j\alpha_j\sigma_{j,T}(n), \tag{2}$$

$$A_k(n) = S_k\beta_k\sigma_{k,T}(n), \tag{3}$$

$$\sigma_{j,T}(n) = (N_j - y_j(n))/N_j, \tag{4}$$

$$\sigma_{k,T}(n) = (S_k + y_k(n))/S_k. \tag{5}$$

where:

- $N_j$ – the number of class $j$ Engset sources,
- $\alpha_j$ – the mean traffic offered by a single idle Engset source of class $j$,
- $y_j(n)$ – the mean number of class $j$ Engset calls serviced in state $n$,
- $S_k$ – the number of class $k$ Pascal sources,
- $\beta_k$ – the mean traffic offered by a single idle Pascal source of class $k$,

- $y_k(n)$ – the mean number of class $k$ Pascal calls serviced in state $n$.

The mean traffic offered by a single idle Engset class $j$ source and Pascal class $k$ source can be expressed by the following formulas:

$$\alpha_j = \gamma_j/\mu_j, \tag{6}$$

$$\beta_k = \gamma_k/\mu_k, \tag{7}$$

where $\gamma_j$ and $\gamma_k$ are the arrival rates of calls generated by a single free source of class $j$ and $k$, respectively. In the model considered we assume that the holding time for calls of particular BPP traffic classes have an exponential distribution with intensity $\mu_j$ and $\mu_k$, respectively.

In order to determine the occupancy distribution in the considered system, it is first necessary to define the state transition coefficients $\sigma_{c,S_L}(n)$[2]. These coefficients take into account the dependence between call streams and the state of the system and allow us to determine the part of the incoming call stream $\lambda_c$ to be transferred between the states $n$ and $n+t_c$ due to the specific structure of LAG. The parameter $\sigma_{c,S_L}(n)$ does not depend on the arrival process and can be calculated as follows [6]:

$$\sigma_{c,S_L}(n) = \frac{F(V-n,\upsilon,f,0) - F(V-n,\upsilon,t_c-1,0)}{F(V-n,\upsilon,f,0)}, \tag{8}$$

where $F(x,\upsilon,f,t)$ is the number of arrangements of $x$ free BBUs in $\upsilon$ links under the assumption that the capacity of each link is equal to $f$ BBUs and that in each link there is at least $t$ free BBUs:

$$F(x,\upsilon,f,t) = \sum_{r=0}^{\left\lfloor \frac{x-\upsilon t}{f-t+1} \right\rfloor} (-1)^r \binom{\upsilon}{r} \binom{x-\upsilon(t-1)-1-r(f-t+1)}{\upsilon-1}. \tag{9}$$

Let us observe that the total value of the transition coefficient in LAG with bandwidth reservation can be expressed in the form of the product of the coefficients $\sigma_{c,S_L}$, related to the structure of the group, and $\sigma_{c,S_R}$, related to the introduced reservation mechanism:

$$\sigma_{c,S}(n) = \sigma_{c,S_L}(n) \cdot \sigma_{c,S_R}(n), \tag{10}$$

where $\sigma_{c,S_R}(n)$:

$$\sigma_{c,S_R}(n) = \begin{cases} 1 & \text{for} \quad n \le Q, \\ 0 & \text{for} \quad n > Q. \end{cases} \tag{11}$$

[2]In the present paper, the letter "$i$" denotes an Erlang traffic class, the letter "$j$" - an Engset traffic class, the letter "$k$" - a Pascal traffic class, and the letter "$c$" - an arbitrary traffic class.

The possibility of a product-form presentation of the dependence between the admission process and the occupancy distribution stems from the fact that the introduced reservation mechanism does not depend on the structure of LAG.

Taking into account the presented dependencies, the iterative method of blocking probability calculation in LAG with bandwidth reservation can be presented in the form of the following Multiple Iteration Algorithm [10]:

1. Setting of the iteration number $l = 0$.
2. Determination of the initial values $y_j^{(0)}(n)$, $y_k^{(0)}(n)$:
   $\bigvee_{1\le j\le M_2} \bigvee_{0\le n\le V} y_j^{(0)}(n) = 0, \bigvee_{1\le k\le M_3} \bigvee_{0\le n\le V} y_k^{(0)}(n) = 0.$
3. Increase of the iteration number: $l = l + 1$.
4. Calculation of state-passage coefficients $\sigma_{j,T}^{(l)}(n)$ i $\sigma_{k,T}^{(l)}(n)$ (Eq. (4) and (5)).
5. Determination of state probabilities $\left[P_n^{(l)}\right]_V$ [10]:

$$\begin{aligned} n\left[P_n^{(l)}\right]_V = &\sum\nolimits_{i=1}^{M_1} A_i\sigma_{i,S}(n-t_i)t_i\left[P_{n-t_i}^{(l)}\right]_V + \\ &+\sum\nolimits_{j=1}^{M_2} N_j\alpha_j\sigma_{j,T}^{(l-1)}(n-t_j)\sigma_{j,S}(n-t_j)t_j\left[P_{n-t_j}^{(l)}\right]_V + \\ &+\sum\nolimits_{k=1}^{M_3} S_k\beta_k\sigma_{k,T}^{(l-1)}(n-t_k)\sigma_{k,S}(n-t_k)t_k\left[P_{n-t_k}^{(l)}\right]_V. \end{aligned} \quad (12)$$

6. Calculation of reverse transition rates $y_j^{(l)}(n)$ and $y_k^{(l)}(n)$:

$$y_c^{(l)}(n) = \begin{cases} A_c^{(l-1)}(n-t_c)\sigma_{c,S}(n-t_c)\left[P_{n-t_c}^{(l-1)}\right]_V \Big/ \left[P_n^{(l-1)}\right]_V & \text{for } n \le V, \\ 0 & \text{for } n > V. \end{cases} \quad (13)$$

7. Repetition of Steps 3, 4, 5 and 6 until the assumed accuracy $\xi$ of the iterative process is obtained:

$$\bigvee_{0\le n\le V} \left| \frac{y_c^{(l-1)}(n) - y_c^{(l)}(n)}{y_c^{(l)}(n)} \right| \le \xi. \quad (14)$$

8. Determination of blocking probabilities $e(c)$ for class $c$ calls:

$$e(c) = \sum\nolimits_{n=0}^{V-t_c} [P_n]_V \left(1 - \sigma_{c,S}(n)\right) + \sum\nolimits_{n=V-t_c+1}^{V} [P_n]_V \, \sigma_{c,S}(n). \quad (15)$$

### 3.2. Full-availability group

Subsequently, let us consider the full-availability group (FAG) with capacity equal to $V$ BBUs. The group is offered traffic streams of three types: $M_1$ Erlang traffic streams, $M_2$ Engset traffic streams and $M_3$ Pascal traffic streams. The full-availability group is the model of a system with state-independent service process, i.e. the system with complete sharing policy. Therefore, the conditional state-passage probability $\sigma_{c,S}(n)$ in FAG is equal to 1 for all states and for each traffic class. Consequently, the occupancy distribution and blocking probabilities in the groups with infinite and finite source population can be calculated by the equations (12) and (15), taking into consideration the fact that:

$$\bigvee_{1\leq c\leq M_1+M_2+M_3} \bigvee_{1\leq n\leq V} \sigma_{c,S}(n) = 1 \tag{16}$$

Bandwidth reservation in the full-availability group consists in designating the reservation threshold $Q$ for each traffic class. The reservation thresholds for calls of particular traffic classes are determined in a manner of ensuring blocking probability equalization for all traffic streams offered to the system:

$$Q = V - t_{\max}. \tag{17}$$

The occupancy distribution and blocking probabilities in the FAG with bandwidth reservation and BPP traffic can be calculated by Eq. (12), where $\sigma_{c,S}(n)$ is determined by Eq. (11).

## 4. Effective availability

The analytical models of switching networks with bandwidth reservation, proposed in Section 6., are based on the effective availability concept. The basis of the determination of the effective availability for class $c$ stream is the concept of the so-called equivalent switching network [6], carrying single-rate traffic. Each link of the equivalent network has a fictitious load $e_l(c)$ equal to blocking probability for class $c$ stream in a link of a real switching network between section $l$ and $l+1$. This probability can be calculated on the basis of the occupancy distribution in the full-availability group with or without bandwidth reservation. The effective availability $d_{c,z}$ for class $c$ stream in $z$-stage switching network can be calculated by using following formula [6]:

$$d_{c,z} = [1-\pi_z(c)]\upsilon + \pi_z(c)\eta\upsilon e_1(c) + \pi_z(c)[\upsilon - \eta\upsilon e_1(c)]e_z(c)\sigma_z(c), \tag{18}$$

where: $\pi_z(c)$ – the probability of non availability of a given last stage switch for the class $c$ connection; $\upsilon$ – the number of outgoing links from the first stage switch,

$\eta$ – a portion of the average fictitious traffic from the switch of the first stage which is carried by the direction in question, $\sigma_z(c)$ – the so-called secondary availability coefficient, determined by equation [11]:

$$\sigma_z(c) = 1 - \prod_{r=2}^{z-1} \pi_r(c). \tag{19}$$

## 5. Reservation algorithms in switching networks

In Algorithm 1, the common reservation threshold is determined for the outgoing direction (modeled by LAG) independently of the distribution of busy bandwidth units in particular links of the direction. The reservation mechanism is not introduced in interstage links. In this algorithm the reservation threshold $Q$ is designated for all call classes with the exception of the oldest class (i.e. the one which requires the greatest number of bandwidth units to set up a connection). This means that only calls of the oldest class can be serviced by the system in states belonging to the reservation space $R$.

In Algorithm 2 the reservation threshold $Q = f - t_{\max}$ is introduced for all traffic links of the switching network, both interstage and outgoing links. The presented algorithm allows us to obtain the total blocking equalisation for all traffic classes.

## 6. Point-to-point blocking probability in switching network with bandwidth reservation and BPP traffic

In this section two approximate methods of point-to-point blocking probability calculation in multi-stage switching networks with multi-rate BPP traffic and bandwidth reservation are presented, i.e. PPBPPR (Point to Point Blocking for BPP Traffic with Reservation) method and PPBPPRD (Point-to-Point Blocking for BPP Traffic with Reservation – Direct Method) method. The presented considerations are based on PPBMF and PPFD methods, worked out in [12, 13] for multiservice switching networks with BPP traffic and without bandwidth reservation.

The presented switching networks calculations are based on the reduction of calculations of internal blocking probability in a multi-stage switching network with BPP traffic to the calculation of the probability in an equivalent switching network model servicing single channel traffic. Such an approach allows us to analyse multi-stage switching networks with multi-rate traffic with the use of the effective availability method.

### 6.1. PPBPPR Method

Let us consider now the PPBPPR method for blocking probability calculation in switching networks with bandwidth reservation and point-to-point selection, servicing multi-rate BPP traffic streams. The basis for the proposed method is the PPBMF method worked out in [12] for switching networks without bandwidth reservation. Modifications to the PPBPPR method consists in the introduction of the appropriate group models with bandwidth reservation to calculations.

The internal point-to-point blocking probability can be calculated on the basis of the following equation:

$$E_c^{in} = \sum_{s=0}^{\upsilon - d_e(c)} P(c, s \wedge 1) \left[ \binom{\upsilon - s}{d_e(c)} / \binom{\upsilon}{d_e(c)} \right], \tag{20}$$

where $d(c)$ is the effective-availability parameter for class $c$ calls in the switching network with bandwidth reservation and $P(c, s \wedge 1)$ is the so-called *combinatorial distribution of available links in a switch*. This distribution can be calculated on the basis of the following formula:

$$P(c, s \wedge 1) = \frac{\sum_{x=0}^{V} P(c, s|x)[1 - P(c, 0|x)][P_{V-x}]_V}{1 - \sum_{n=0}^{k} \left[ \sum_{x=0}^{V} P(c, n|x) P(c, 0|x)[P_{V-x}]_V \right]}, \tag{21}$$

where $[P_n]_V$ is the occupancy distribution in LAG with BPP traffic and bandwidth reservation, and $P(c, s|x)$ is the so-called conditional distribution of available links in LAG with BPP traffic and bandwidth reservation. The distribution $P(c, s|x)$ determines the probability of an arrangement of $x$ $(x = V - n)$ free BBUs, in which each of $s$ arbitrarily chosen links has at least $t_c$ free BBUs, while in each of the remaining $(\upsilon - s)$ links the number of free BBUs is lower than $t_c$. Following the combinatorial consideration [6]:

$$P(c, s|x) = \binom{k}{s} \sum_{w=st_c}^{\Psi} F(w, s, f, t_c) F(x - w, \upsilon - s, t_c - 1, 0) \Big/ F(k, x, f, 0), \tag{22}$$

where: $\Psi = sf$, if $x \geq sf$, $\Psi = x$, if $x < sf$. In the case of the switching network with Algorithm 2, in Equation (22) we assume: $t_c = t_{\max}$.

The phenomenon of the external blocking occurs when none of outgoing links of the demanded direction of the switching network can service the class $c$ call. The blocking probability in the outgoing direction can be approximated by the blocking probability in LAG with BPP traffic and bandwidth reservation:

$$E_c^{ex} = e(c), \tag{23}$$

where $P(c,s)$ is the distribution of available links for class $c$ calls in LAG with BPP traffic. This distribution determines the probability $P(c,s)$ of an event in which each of arbitrarily chosen $s$ links can carry the class $c$ call:

$$P(c,s) = \sum_{n=0}^{V} [P_n]_V P(c,s|V-n), \tag{24}$$

The total blocking probability $E_c$ for the class $c$ call is a sum of external and internal blocking probabilities. Assuming the independence of internal and external blocking events:

$$E_c = E_c^{ex} + E_c^{in}[1 - E_c^{ex}]. \tag{25}$$

### 6.2. PPBPPRD

In the PPBPPRD method the evaluation of the internal point-to-point blocking probability in switching network with bandwidth reservation is made on the basis of the effective availability quotient and the capacity of an outgoing group:

$$E_c^{in} = v - d_e(c)/\,v. \tag{26}$$

The proposed method is based on the the PPFD method, elaborated in [12] for switching networks without bandwidth reservation and BPP traffic.

The phenomenon of external blocking occurs when none of outgoing links of the demanded direction in a switching network can service a class $c$ call. The occupancy distribution of the outgoing direction can be approximated by the occupancy distribution in LAG with bandwidth reservation and BPP traffic. Consequently, the external blocking probability $E_c^{ex}$ and the total blocking probability $E_c$ for class $c$ calls, can be calculated by (23) and (25).

## 7. Numerical results

The presented methods for determining point-to-point blocking probability in switching networks with BPP traffic and bandwidth reservation are approximate ones. In order to confirm the adopted assumption, the results of analytical calculations were compared with the simulation data. The research was carried for 3-stage switching network consisting of the switches $v \times v$ links, each with capacity of $f$ BBUs. The results of research are presented in Fig. 2-5, depending on the value of traffic offered to single BBU or the value of reservation space $R$. The simulation results are shown in the form of marks with 95% confidence intervals

that have been calculated according to $t$-Student distribution for the five series with 1,000,000 calls of each class.

In considered switching networks two reservation algorithms were implemented. The research was carried out for two structures of offered traffic. The values of holding times, the numbers of BBUs demanded for calls of particular traffic classes and the numbers of traffic sources of particular traffic classes are as follows:

- structure 1: class 1 (Erlang): $t_1 = 1$ BBU, $\mu_1^{-1} = 1$; class 2 (Pascal): $t_2 = 3$ BBUs, $\mu_2^{-1} = 1$, $N_2 = 362$; class 3 (Engset): $t_3 = 5$ BBUs, $\mu_3^{-1}$, $N_3 = 362$; class 4 (Engset): $t_4 = 8$ BBUs, $\mu_4^{-1}$, $N_4 = 362$; (Fig. 2 and 3),
- structure 2: class 1 (Erlang): $t_1 = 1$ BBU, $\mu_1^{-1} = 1$; class 2 (Pascal): $t_2 = 3$ BBUs, $\mu_2^{-1} = 1$, $N_2 = 544$; class 3 (Engset): $t_3 = 10$ BBUs, $\mu_3^{-1}$, $N_3 = 544$; (Fig. 4 and 5).

## 8. Conclusions

In the paper the analytical methods for determining blocking probability in switching networks with BPP traffic and different reservation mechanisms were presented. The reservation algorithms ensure a substantial decrease in blocking probabilities of certain traffic classes in the access to switching network resources. In the particular case, the algorithms can be applied to equalize the level of call service of a particular stream or all traffic streams. The results of analytical calculations of the considered switching networks were compared with the simulation data which confirmed high accuracy of the proposed methods. It should be emphasized that the proposed methods can be used for blocking probability calculation in the state-dependent systems with infinite and finite source population.

## References

[1] Y. Zhou and G.-S. Poo, "Optical multicast over wavelength-routed WDM networks: A survey," *Optical Switching and Networking*, vol. 2, no. 3, pp. 176–197, 2005.

[2] M. Głąbowski and M. Stasiak, "Point-to-point blocking probability in switching networks with reservation," *Annales des Télécommunications*, vol. 57, no. 7–8, pp. 798–831, 2002.

[3] M. Stasiak and M. Głąbowski, "PPBMR method of blocking probability calculation in switching networks with reservation," in *Proc. GLOBECOM 1999*, vol. 1a, Dec. 1999, pp. 32–36.

[4] M. Stasiak and M. Głąbowski, "Point-to-point blocking probability in switching networks with reservation," in *Proc. 16th International Teletraffic Congress*, vol. 3A. Edinburgh: Elsevier, Jun. 1999, pp. 519–528.

[5] M. Głąbowski and M. Sobieraj, "Recurrent method for determining blocking probability in multi-service switching networks with BPP traffic and bandwidth reservation," in *Information Systems Architecture and Technology*, Oficyna Wydawnicza Politechniki Wrocşawskiej, 2009, vol. Service Oriented Distributed Systems: Concepcts and Infrastructure, pp. 205–218.

[6] M. Stasiak, "Combinatorial considerations for switching systems carrying multichannel traffic streams," *Annales des Télécommunications*, vol. 51, no. 11–12, pp. 611–625, 1996.

[7] V. G. Vassilakis, I. D. Moscholios, and M. D. Logothetis, "Call-level performance modelling of elastic and adaptive service-classes with finite population," *IEICE Transactions on Communications*, vol. E91-B, no. 1, pp. 151–163, 2008.

[8] K. Ross, *Multiservice Loss Models for Broadband Telecommunication Network*. London: Springer, 1995.

[9] M. Pióro, J. Lubacz, and U. Korner, "Traffic engineering problems in multiservice circuit switched networks," *Computer Networks and ISDN Systems*, vol. 20, pp. 127–136, 1990.

[10] M. Głąbowski, "Modelling of state-dependent multi-rate systems carrying BPP traffic," *Annals of Telecommunications*, vol. 63, no. 7-8, pp. 393–407, Aug. 2008.

[11] M. Stasiak, "An approximate model of a switching network carrying mixture of different multichannel traffic streams," *IEEE Trans. on Communications*, vol. 41, no. 6, pp. 836–840, 1993.

[12] M. Głąbowski, "Point-to-point blocking probability calculation in multi-service switching networks with BPP traffic," in *Proceedings of 14th Polish Teletraffic Symposium*, T. Czachórski, Ed., Zakopane, Sep. 2007, pp. 65–76.

[13] M. Głąbowski, "Point-to-point and point-to-group blocking probability in multiservice switching networks with BPP traffic," *Electronics and Telecommunications Quarterly*, vol. 53, no. 4, pp. 339–360, 2007.

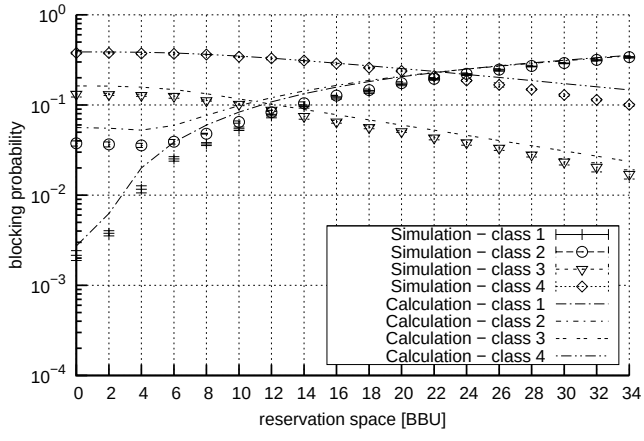



Fig. 2: Point-to-point blocking probability in the switching network with reservation algorithm 1, for $a = 0.9$. PPBPPR method. First structure of offered traffic. Structure of switching network: $v = 4$, $f = 34$ BBUs.

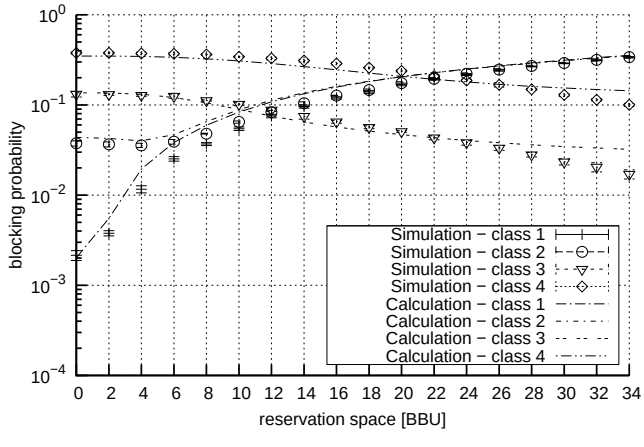



Fig. 3: Point-to-point blocking probability in the switching network with reservation algorithm 1, for $a = 0.9$. PPBPPRD method. First structure of offered traffic. Structure of switching network: $v = 4$, $f = 34$ BBUs.

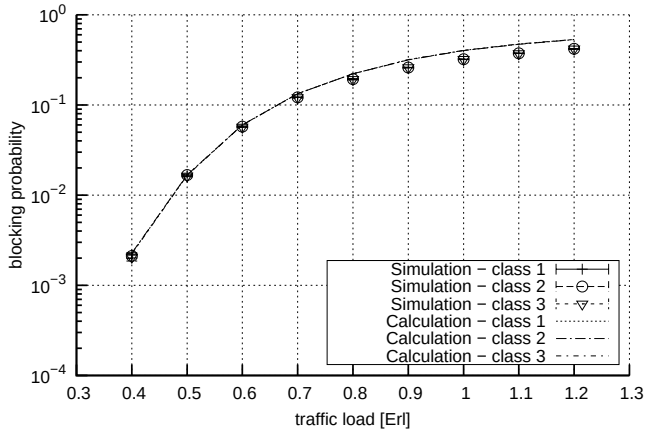



Fig. 4: Point-to-point blocking probability in the switching network with reservation algorithm 2. PPBPPR method. Second structure of offered traffic. Structure of switching network: $v = 4$, $f = 34$ BBUs.

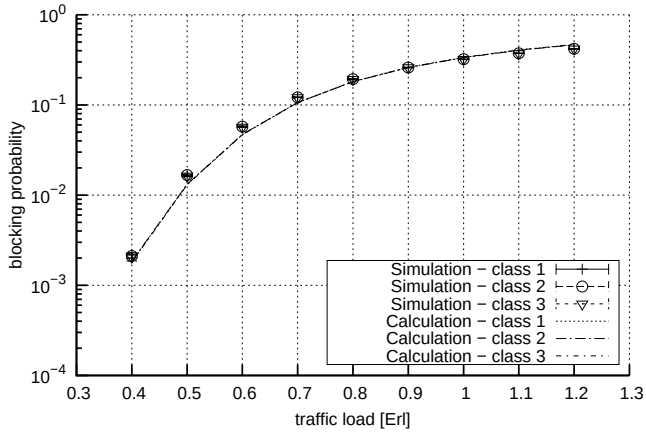



Fig. 5: Point-to-point blocking probability in the switching network with reservation algorithm 2. PPBPPRD method. Second structure of offered traffic. Structure of switching network: $v = 4$, $f = 34$ BBUs.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 311–324

# A tool for the numerical solution of cooperating Markov chains in product-form

SIMONETTA BALSAMO GIAN-LUCA DEI ROSSI
ANDREA MARIN [a]

[a]Università Ca' Foscari di Venezia
Dipartimento di Informatica
via Torino, 155 Mestre
Italy
{balsamo,deirossi,marin}@dsi.unive.it

**Abstract:**
Performance modelling of complex and heterogeneous systems based on analytical models are often solved by the analysis of underlying Markovian models. We consider performance models based on Continuous Time Markov Chains (CTMCs) and their solution, that is the analysis of the steady-state distribution, to efficiently derive a set of performance indices. This paper presents a tool that is able to decide whether a set of cooperating CTMCs yields a product-form stationary distribution. In this case, the tool computes the unnormalised steady-state distribution. The algorithm underlying the tool has been presented in [10] by exploiting the recent advances in the theory of product-form models such as the Reversed Compound Agent Theorem (RCAT) [5]. In this paper, we focus on the peculiarities of the formalism adopted to describe the interacting CTMCs and on the software design that may have interesting consequences for the performance community.
**Keywords:** : Product-form, queueing networks, numerical solution of Markov chains

## 1. Introduction

Markovian models have proved to be a robust and versatile support for the performance analysis community. Performance modeling of complex heterogeneous systems and networks based on analytical model usually describes a system using a high-level formalism, such as Stochastic Petri Nets (SPNs), Performance Evaluation Process Algebra (PEPA), queueing systems or networks, from which its underlying Continuous Time Markov Chain (CTMC) is derived. The desired performance indices, at steady-state, are computed by the analysis of the model

CTMC. This computation is usually a hard task, when not unfeasible, because the solution of the CTMC usually requires to solve the system of Global Balance Equations (GBEs) (with a computational complexity of $O(Z^3)$, where $Z$ is the number of states of the model) to derive the stationary probability of each state. Some algorithms that numerically solve the GBEs more efficiently for special cases or using approximations have been defined.

Product-form models take a different approach. They apply the *divide et impera* paradigm to efficiently solve complex models. Informally, a model $S$ is seen as consisting of several interacting sub-models $S_1, \ldots, S_N$ so that $\mathbf{m} = (m_1, \ldots, m_N)$ is a state of $S$ and $m_i$ is a state of $S_i$. $S$ is in product-form with respect to $S_1$, ..., $S_N$ if its stationary distribution $\pi(\mathbf{m})$ satisfies the following property:

$$\pi(\mathbf{m}) \propto \prod_{i=1}^{N} g_i(m_i),$$

where $g_i$ is the stationary distribution of sub-model $i$ considered in isolation and opportunely parametrised. Roughly speaking, from the point of view of a single sub-model $S_i$, the parametrisation abstracts out the interactions with all the other sub-models $S_j$, $j \neq i$. It should be clear that, since the state space of a sub-model $S_i$ is much smaller than that of $S$ the solution of its GBEs may be computed efficiently. Note that modularity becomes a key-point both for the analysis and the description of the model, since it is a good engineering practice to provide modular models of systems.

Exploiting the product-form solutions requires to address two problems: 1) Deciding if model $S$ is in product-form with respect to the given sub-models $S_1$, ..., $S_N$; 2) Computing the parametrisation of the sub-models $S_1$, ..., $S_N$ in order to study them in isolation. Note that we have not listed the solution of the sub-model CMTCs as a problem because we suppose that the cardinalities of their state spaces are small enough to directly solve the GBEs. If this is not the case, a product-form analysis of the sub-models may be hierarchically applied. In literature, the first problem has been addressed in two ways. The first consists in proving that a composition of models that yield some high-level characteristics is in product-form. For instance the BCMP theorem [2] is based on this idea because the authors specify four type of queueing disciplines with some service properties and prove that a network of such models has a product-form solution. The second way is more general, i.e., the properties for the product-form are defined at the CTMC level. Although this can lead to product-form conditions that are difficult to interpret, this approach really enhances the compositionality of the models. In this paper, we often refer to a recent result about product-forms: the Reversed Compound Agent Theorem (RCAT) [5]. This theorem has been extensively used to prove a large set

of product-form results previously known in literature (BCMP product-form [4], G-networks with various types of triggers [6], just to mention a few). Problem 2 is usually strictly related to Problem 1. In general, the parametrisation of the sub-models requires the solution of a system of equations that is called system of traffic equations. For several years the fact that product-form solutions must be derived from linear systems of traffic equations has been considered true, but the introduction of G-networks has shown that this is not necessary.

*Contribution.* This paper illustrates a tool that given the description of a set of cooperating CTMCs (i.e., when some transitions in one chain force transitions in other chains) it decides whether the model is in product-form and, in this case, computes its stationary distribution. The tool is based on the algorithm presented in [10] which is briefly resumed in Section 2.. Since the analysis of the product-form is performed at the CTMC level, it is able to study product-form models that are originated form different formalisms, such as exponential queueing networks, G-networks or queueing networks with blocking. To this aim, we observe that it is important to decouple the analyser and the model specification interface (MSI). We propose both a Java implementation of the analyser and of a general MSI (note that multiple specification interfaces may be implemented according to the modeller needs). With this tool, a modeller has a library of product-form models that, even if they were created using some (possibly high-level) formalism, are stored as stochastic automata (basically a CTMC with labelled transitions allowing self-loops or multiple arcs between states). Using the MSI (which acts as a client with respect to the analyser), the various sub-models can be instantiated and their interactions be specified. The operations that the modeller performs in the MSI are translated into commands for the server side, i.e., the analyser. The analysis is requested from the MSI, computed by the analyser and displayed by the MSI. We have also developed a textual interface that will not be presented in this paper to allow the usage of the analyser from non-graphical clients.

*Paper structure.* The paper is structured as follows. Section 2. briefly illustrates the formalism used to describe the interactive CTMCs, the idea underlying RCAT and the algorithm presented in [10]. Section 3. gives the details of the software implementation. In particular, Section 3.1. presents the naming conventions which are an important matter to enhance the modularity of the tool, Section 3.2. the client server architecture, and finally Section 3.3. gives a brief idea of some use-cases. Section 4. shows an instance of implemented MSI. Some final remarks conclude the paper in Section 5..

## 2. Theoretical background

In this section we briefly recall some basic definitions and models to keep the paper self-contained. We present the topics in a way that allows us to simplify the description of the tool algorithm, features and architecture in what follows.

Let us suppose to have $N$ model $S_1, \ldots, S_N$ that cooperate, i.e., some transitions in a model $S_i$ force other transitions in a model $S_j$, $i \neq j$. At a low-level we can describe each model by a set of labelled matrices: $\mathbf{M}_i^a$ is the matrix with label $a$ associated with model $S_i$. Labels may be chosen arbitrarily when a model is defined. However, we always assume that every model has at least one label called $\epsilon$. We consider, at first, models with a finite number of states, $Z_i$. $\mathbf{M}_i^a$ is a $Z_i \times Z_i$ matrix with non-negative elements that represent the transition rates between two states of the model. Note that self-loops, i.e., transitions from a state to itself, are allowed. The infinitesimal generator $\mathbf{Q}_i$ can be easily computed as the sum of all the matrices associated with a model, where the diagonal elements are replaced with the opposite of the sum of the extra-diagonal row elements. If the stationary distribution $\boldsymbol{\pi}$ exists (and hereafter we will work under this hypothesis) then it can be computed as the unique solution of $\boldsymbol{\pi}\mathbf{Q} = \mathbf{0}$ subject to $\boldsymbol{\pi}\mathbf{1} = 1$. From $\boldsymbol{\pi}$ we can compute the rates in the reversed process associated with each label [9, 5] in a straightforward way. Suppose that $\mathbf{M}_i^a[\alpha, \beta] > 0$, with $1 \leq \alpha, \beta \leq Z_i$ and $1 \leq i \leq N$, then the reversed rate of this transition, denoted by $\overline{\mathbf{M}_i^a[\alpha, \beta]}$ is defined as follows:

$$\overline{M_i^a[\alpha, \beta]} = \frac{\pi(\alpha)}{\pi(\beta)} M_i^a[\alpha, \beta]. \tag{1}$$

Let us show how we specify the interaction of two models. According to RCAT restrictions, we just deal with pairwise interactions, i.e., a transition in a model may cause a transition just for another model. The cooperation semantics used in this paper (but also in [5]) is very similar to that specified by PEPA, i.e., a Markovian stochastic process algebra introduced by Hillston in [8]. Consider sub-models $S_i$ and $S_j$ and suppose that we desire to express the fact that a transition labelled with $a$ in $S_i$ can occur only if $S_j$ performs a transition labelled with $b$, and vice-versa. Specifically, if $S_i$ and $S_j$ are in states $s_i$, $s_j$ such that they are able to perform a transition labelled with $a$ and $b$, respectively, that take the sub-models to state $s_i'$ and $s_j'$, then they can move simultaneously to state $s_i'$ and $s_j'$. The rate at which this joint transition occurs is decided by the active sub-model that can be $S_i$ or $S_j$. We express such a cooperation between $S_i$ and $S_j$, with $S_i$ active, as follows:

$$S_i \underset{(a^+, b^-)}{\overset{y}{\times}} S_j,$$

which means that transitions labelled by $a$ is $S_i$ are active with respect to the cooperation with transitions labelled by $b$ of $S_j$ and originate a models where the joint transitions are labelled with $y$. The fact that the resulting model is still Markovian should be obvious because the synchronisation inherits the properties derived for that of PEPA. Note that the major difference is that we can synchronise different labels and assign a different name to the resulting transitions. This happens because we would like a modeller to be able to use a library of models whose labels have a local scope. In this way the library items can be created independently and instantiated several times in the same model.

**Example 1 (Example of cooperation)** *Suppose we would like to model within the presented framework the trivial queueing network depicted in Figure 1 where two identical exponential queues with finite capacities $B$ are composed in tandem. When the first queue if saturated, arrivals are lost. When the second queue is saturated at a job completion of the first queue, the customer is served again (repetitive service blocking). Customers arrive to the first queue according to a Poisson process with rate $\lambda$. A queue can be described by three matrices with dimension*

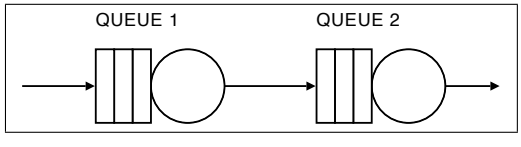



Fig. 1. Tandem of two exponential finite capacity queues.

*$B \times B$:*

- $\mathbf{M}^{\epsilon} = \mathbf{0}$ *that describes the transitions that cannot synchronise (something like the private part of the model).*
- $\mathbf{M}^{a}$ *where* $\mathbf{M}^{a}[\alpha, \beta] = \lambda$ *if* $\beta = \alpha + 1$ *or* $\mathbf{M}^{a}[\alpha, \beta] = 0$*, otherwise. This matrix describes the transitions corresponding to arrival events.*
- $\mathbf{M}^{d}$*, where* $\mathbf{M}^{d}[\alpha, \beta] = \mu$ *if* $\beta = \alpha - 1$ *or* $\mathbf{M}^{d}[\alpha, \beta] = 0$*, otherwise. This matrix describes the transitions corresponding to job completion events.*

*Consider two instances of this model, $S_1$ and $S_2$. The tandem network of Figure 1 can be described by the model* $S_1 \times^{y}_{(d^+, a^-)} S_2$.

A pairwise cooperation may involve more than one label. In this case we may write:

$$S_1 \underset{(a_1^+, b_1^-)}{\overset{y_1}{\times}} \underset{(a_2^-, b_2^+)}{\overset{y_2}{\times}} S_2$$

to specify that $S_1$ ($S_2$) is active on $y_1$ ($y_2$) and passive on $y_2$ ($y_1$) with transitions labelled $a_1$ ($b_1$) and $a_2$ ($b_2$), respectively.

The following operator allows us to change all the rates of a matrix labelled by $a$: $S_1\{a \leftarrow \lambda\}$ is the sub-model $S_1$ with only matrix $\mathbf{M}^a$ modified so that all its non-zero elements are set to $\lambda$.

*RCAT.* Theorem RCAT [5] gives sufficient conditions to decide and derive the product-form solution of pairwise interactions (possibly involving more than one label) between two sub-models. Le us consider the following synchronisation:

$$S = S_1 \underset{(a_1^*, b_1^*)}{\overset{y_1}{\times}} \dots \underset{(a_T^*, b_T^*)}{\overset{y_T}{\times}} S_2,$$

where symbol $*$ stands either for a $+$ or a $-$. The conditions to apply RCAT are:

1. If $a_t$ is active (passive) in $S_1$ and $b_t$ is passive (active) in $S_2$ then $M_1^{a_t}[\cdot, z]$ ($M_1^{a_t}[z, \cdot]$) has exactly one non-zero element for every $1 \leq z \leq Z_1$, and $M_2^{b_t}[z, \cdot]$ ($M_2^{b_t}[\cdot, z]$) has exactly one non-zero element for every $1 \leq z \leq Z_2$[1].
2. Suppose that for every pair $(a_t^+, b_t^-)$ ($(a_t^-, b_t^+)$) we know a value $\beta_t$ ($\alpha_t$) such that:

$$\begin{aligned} S_1' &= S_1\{a_t \leftarrow \alpha_t\} && \text{for all } a_t \text{ passive in the cooperation} \\ S_2' &= S_2\{b_t \leftarrow \beta_t\} && \text{for all } b_t \text{ passive in the cooperation} \end{aligned}$$

   and given an active label $a_t$ ($b_t$) in $S_1$ ($S_2$) all the transitions with that label have the same reversed rate $\beta_t$ ($\alpha_t$).

If these conditions are satisfied, then the stationary distribution $\boldsymbol{\pi}$ of $S$ is $\boldsymbol{\pi} \propto \boldsymbol{\pi}_1 \boldsymbol{\pi}_2$ (for each positive recurrent state ).

Basically, the first condition says that every state of a model which is passive (active) with respect to a label must have one outgoing (incoming) transition with that label. To understand the second condition, suppose that $(a_t^+, b_t^-)$ is a pair in the synchronisation between $S_1$ and $S_2$. Then, we must determine a rate $\beta_t$ to assign to all the transitions labelled by $b_t$ that is also the constant reversed rate of the active transitions $a_t$ in $S_1$. Note that, in general, this task is not easy, and is shown to be equivalent to the solution of the traffic equations in Jackson networks and G-networks. The algorithm proposed in [10] aims to give an iterative, numerical and efficient way to perform this computation.

[1] $M_1^{a_t}[z, \cdot]$ represents the $z$-th row vector of the matrix, and analogously $M_2^{a_t}[\cdot, z]$ represents the $z$-th column vector.

Although it is out of the scope of this paper discussing the modelling implications of RCAT conditions, it is worth pointing out that several works in literature have proved that this result has not only a theoretical relevance but can be actually used to characterise the product-form solutions for models that may be used for practical case-studies.

*The underlying algorithm.* The algorithm that underlies our tool has been presented in [10]. It takes the matrices that describe models $S_1, \ldots, S_N$ and the synchronisations as input, and computes as output a boolean value which is true if a product-form has been identified, false otherwise. In case of product-form, the unnormalised stationary distribution is output. In its simplest formulation (two submodels and without optimisations) it can be summarised in the following steps:

1. Generate randomly $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$
2. Compute the reversed rates of the active transitions using Equation (1)
3. Use the mean of the reversed rates for each label to set the rates of the corresponding passive transitions. For instance let $a$ be active for $S_1$ and $b$ passive for $S_2$. Then let $x$ be the mean of the reversed rates of the non-zero elements in $\mathbf{M}_1^a$. $\mathbf{M}_2^b$ is updated by setting all the non-zero elements to $x$
4. Compute $\boldsymbol{\pi}_1$ and $\boldsymbol{\pi}_2$ as solution of the GBEs of the underlying CTMCs of $S_1$ and $S_2$
5. Are the reversed rates of the transitions constant for each active label?
   - **true** $\Rightarrow$ product-form found and the stationary distribution is $\boldsymbol{\pi} \propto \boldsymbol{\pi}_1\boldsymbol{\pi}_2$ and terminate.
   - **false** and the maximum number of iterations has been reached $\Rightarrow$ product-form not found and terminate.
   - **false** and the maximum number of iterations has not been reached $\Rightarrow$ go to step 3

The algorithm is extended in order to include the possibility of multiple pairwise synchronisations (as proposed in [7]) and several optimisations: an efficient way to define an order in the solution of the sub-models (based on Tarjan's algorithm [12]), a parallel implementation, and a special technique to deal with self-loops.

## 3. Tool

In this section we describe some salient characteristics of the proposed tool. First, we explain our approach in the specification of the interactions between

the sub-models. Then, we describe the client-server architecture and illustrate its strengths.

### 3.1. Specifying the interactions

In order to better understand the motivations of this section, let us consider again the model depicted by Figure 1 with a variation, i.e., after a job completion at the first station the customer may exit the system with probability $p$ or go to the second station with probability $1 - p$, as depicted by Figure 2. We note that the

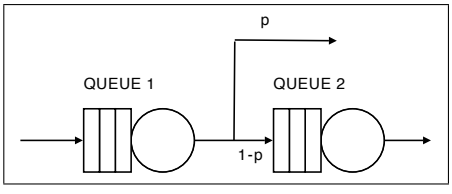



Fig. 2: Probabilistic routing in the model of Figure 1.

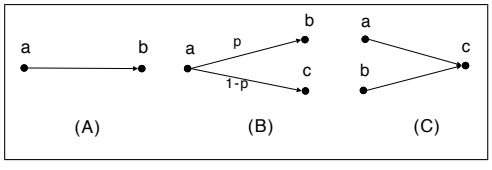



Fig. 3. Types of connections between labels.

processes underlying the first and second queue are different, and we could not use two instances of the same model anymore. Indeed, in the first queue the transition corresponding to a job completion from state $j$ to state $j - 1$ must be split in two: one synchronising with the arrivals in the second queue with rate $(1-p)\mu_1$ and one without synchronisation with rate $p\mu_1$. We decided that this splitting of transitions should be done automatically by our tool, so that the library of sub-models can be defined without any knowledge about the future usage and connections.

From the modeller point of view, a sub-model is seen just as a black box where the labels are exported, i.e., a model specification consists of a set of connections about instances of modules. The simplest possible connection between two labels is that depicted by Figure 3-(A). Note that in this Figure we use a graphical representation of the connections which is coherent with the MSI that we developed, however different approaches are possible (such as a PEPA-like syntax). Figure 3-(A) illustrates a label $a$ of a sub-model that interacts with a label $b$ of another sub-model. The arrow is oriented, meaning that $a$ is active and $b$ is passive. This specification of synchronisation does not require any modification to the structure of the active or passive sub-models. Let us now consider Figure 3-(B). In this case the active action $a$ of one sub-model synchronises with passive actions $b$ (with probability $p$) or $c$ (with probability $1 - p$) of other sub-models. In this case, we need to alter the structure of the active model. Recall that matrix $\mathbf{M}^a$ represents the transitions labelled by $a$. Then we define $\mathbf{M}^{a'} = p\mathbf{M}^a$ and $\mathbf{M}^{a''} = (1 - p)\mathbf{M}^a$. Hence, in the active sub-model, matrices $\mathbf{M}^{a'}$ and $\mathbf{M}^{a''}$ replace matrix $\mathbf{M}^a$. Note that this tech-

nique can be applied to an arbitrary number of probabilistic synchronisations under the obvious constraint that the synchronisation probabilities must sum to a value which is less or equal to 1. Suppose that the sum of the probabilities $p_1, \ldots, p_K$ is $p_t < 1$ (see Figure 2 for an example). In this case we have $\mathbf{M}^{a^k} = p_k \mathbf{M}^a$ for $k = 1, \ldots K$, and $\mathbf{M}^\epsilon$ (which is always present in a model description an represents the transition that cannot synchronise) is replaced by $\mathbf{M}^\epsilon + \mathbf{M}^a(1 - p_t)$. We use the notation $S_1 \times^{y,p}_{(a^+,b^-)} S_2$ to denote that $a$ in $S_1$ is active in the synchronisation with $b$ in $S_2$, and the synchronisation is called $y$ and occurs with probability $p$. The latter case is depicted by Figure 3-(C) where two active labels $a$ and $b$ (that can belong to the same or different sub-models) synchronise with the same passive label $c$. In this case we simply replace matrix $\mathbf{M}^c$ of the passive model with two matrices $\mathbf{M}^{c'}$ and $\mathbf{M}^{c''}$ identical to the former (we do not need to modify the rates since they are replaced with the rate of the corresponding active transitions).

**Example 2 (Application to the model of Figure 2)** *Let us show how we model the tandem of exponential queues with finite capacities $B$ depicted by Figure 2. We still consider two identical instances of the same sub-model which is described in Example 1. The user specifies in some way the interactions. The model corresponding to the second queue does not change, while that corresponding to the first queue becomes the following:*

- $\mathbf{M}^\epsilon = p\mathbf{M}^d$ *that describes the transitions that cannot synchronise*
- $\mathbf{M}^a$,
- $\mathbf{M}^{d'} = (1 - p)\mathbf{M}^d$,

*where $\mathbf{M}^a$ and $\mathbf{M}^d$ are the matrices defined in Example 1.*

It may be worth pointing out some notes about this approach to the specification of the sub-model interactions: 1) Its scope is to allow the specification of a model despite to the synchronisations it will be involved in. For instance, if we have a model of a simple exponential queue, we can straightforwardly define a Jackson queueing network with probabilistic routing by simply instantiating several copies of the same model. Moreover, connections have a simple and intuitive meaning. 2) When an active label is split the infinitesimal generator of the sub-model does not change, i.e., its stationary distribution does not change. Moreover, if the revered rates of the transitions corresponding to active label $a$ are constant in the original model, then also the transitions corresponding to a split label associated with $a$ have constant reversed rates. 3) The effects of the replication of passive label matrices on the algorithmic analysis of the product-form is that the rate associated with the passive transition is the sum of the (constant) reversed rates of every associated

active transition. 4) Specifying pairwise interactions where the same label is simultaneously active and passive with respect to two or more synchronisations is not allowed. This characteristic is inherited from the semantics of the cooperation given in the theoretical paper which this tool is based on.

### 3.2. Client-server architecture

The tool consists of two parts: the analyser (the server) and the MSI (the client). The idea is that although we propose a graphical client side that exploits the strengths of our modular approach and the specification of the module synchronisation, one could write his own MSI in order to make it compatible with the favourite formalism.

The server opens an independent session for each MSI connected. It provides a character interface which is used by the MSI to: 1) Create/Import a sub-model, 2) Specify a synchronisation between two labels of two sub-models, 3) Require the solution of a model given a precision and a maximum number of iterations. In the first and second case the server just answers the client if the required operation has been executed correctly, while the latter one returns the following data: 1) A flag that specifies if the product-form has been identified, 2) The steady-state probabilities of each sub-model, 3) The reversed rates of all the active transitions. Note that knowing the reversed rates of the active transitions means knowing the solution of the system of traffic equations. In [10] it is proved that when the algorithm analyses a Jackson queueing network, the iterations are equivalent to the Jacobi scheme for the solution of the model linear system of traffic equations. Similarly, when it is applied to a G-network it is equivalent to iterative scheme proposed by Gelenbe et al. for the solution of the non-linear system of traffic equations [3].

### 3.3. Use cases

In this section we illustrate some examples of case studies. We give a description of the model which is independent of the MSI that will be adopted. We just focus our attention on three well-known results about product-form that have been widely used in the communication networks performance evaluation analysis, although several other instances may be easily produced.

*Jackson networks.* Jackson networks are easy to study because they are characterised by a linear system of traffic equations. However, in our framework, they require some attention since each sub-model (i.e., each exponential queue) has an infinite state space. In many cases in which the sub-model is known to have a geometric steady-state distribution and the transitions between states $n$ and $n+1$ are the same for all $n \geq 0$, we can simply represent the sub-model using just a pair of

adjacent states [10]. We apply this technique to reduce the infinite state space of a sub-model we must disable the RCAT structural check (Condition 1) because some transitions that are present in the real model, are omitted in the finite one. Figure 4 shows the truncation of an exponential queue. If the synchronisations it will be involved in impose $a$ to be passive and $d$ to be active, we note that Condition 1 of RCAT is satisfied for the infinite model but is not satisfied for the reduced one (e.g., state $n+1$ does not have any incoming active transition or outgoing passive transition). Nevertheless, the algorithm may still be applied, so the structural check for this model must be disabled.

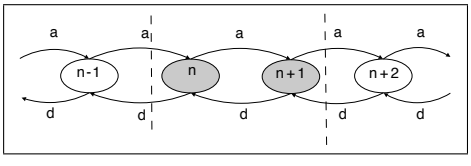



Fig. 4: Truncation of the birth and death process underlying an exponential queue.

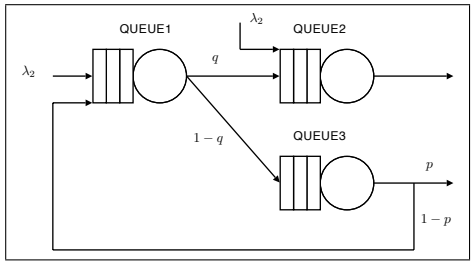



Fig. 5. Jackson network of Example 3.

**Example 3 (Jackson network)** *Consider the Jackson network depicted by Figure 5. A sub-model of an exponential queue consists of three matrices (states are in the order $n$ and $n+1$):*

$$\mathbf{M}^{\epsilon} = \mathbf{0} \quad \mathbf{M}^{a} = \begin{bmatrix} 0 & \lambda \\ 0 & 0 \end{bmatrix} \mathbf{M}^{d} = \begin{bmatrix} 0 & 0 \\ \mu & 0 \end{bmatrix}$$

*We also use a single-state sub-model to represent the external Poisson arrivals with $\mathbf{M}^{\epsilon} = \mathbf{0}$ and $\mathbf{M}^{a} = [\lambda]$. Supposing the service rates for Queue 1, 2 and 3 are $\mu_1$, $\mu_2$ and $\mu_3$, let $S$ be the library model for the queue and $A$ that for the external arrivals, then we have:*

$$S_i = S\{d \leftarrow \mu_i\}\ i = 1,2,3 \qquad A_t = A\{a \leftarrow \lambda_1 + \lambda_2\}$$

*The synchronisations are specified with the following commands to the server:*

$$A_t \underset{(a^+,a^-)}{\overset{y_1,\lambda_1/(\lambda_1+\lambda_2)}{\times}} S_1, \quad A_t \underset{(a^+,a^-)}{\overset{y_2,\lambda_2/(\lambda_1+\lambda_2)}{\times}} S_2, \quad S_1 \underset{(d^+,a^-)}{\overset{y_3,q}{\times}} S_2, \quad S_1 \underset{(d^+,a^-)}{\overset{y_4,1-q}{\times}} \underset{(a^-,d^+)}{\overset{y_5,1-p}{\times}} S_3.$$

*G-networks.* G-networks can be modelled in our frameworks in an analogous way of that presented for Jackson networks. Note that although the models are

different both in the specification and in the analysis, our tool treats them uniformly by exploiting the RCAT theoretical result. The truncation mechanism presented for Jackson queueing centers is applied also for G-queues which consist of three matrices: the epsilon, $A$ representing the transitions for positive customer arrivals, $d$ representing the transitions for the job completion and, finally, $a$ representing the transitions for the negative customer arrivals:

$$\mathbf{M}^{\epsilon} = \mathbf{0}, \quad \mathbf{M}^{A} = \begin{bmatrix} 0 & \lambda_A \\ 0 & 0 \end{bmatrix}, \quad \mathbf{M}^{d} = \begin{bmatrix} 0 & 0 \\ \mu & 0 \end{bmatrix}, \quad \mathbf{M}^{a} = \begin{bmatrix} 0 & 0 \\ \lambda_a & 0 \end{bmatrix}.$$

*Finite capacities queueing networks with blocking.* Akyildiz's product-form queueing networks with blocking [1] can be analysed by this tool. Finite capacity queues have a finite state space so the truncation mechanism is not needed. In order to reproduce Akyldiz's results on the reversible routing it suffices to synchronise a departure label of a queue with an arrival label of another queue considering the former passive and the latter active.

## 4. MSI implementation example

In this Section we illustrate a possible implementation of the MSI. Recall that the tool client-server architecture allows for different MSIs according to the modeller's needs. We show a general-purpose MSI that is independent of the formalism used by the modeller. As an example we model the Jackson network depicted by Figure 5. Each sub-model is represented by a coloured circle and arcs represent the synchronisations. Each object, circle or arc, has a name. In the former case it is the sub-model name, in the latter it has the form $y(a, b)$ that stands for $S_1 \times^{y}_{(a^+,b^-)} S_2$, where $S_1$ is the sub-model from which the arc outgoes from, and $S_2$ is the destination sub-model. A screen-shot is shown in Figure 6. By clicking on a sub-model circle a window appears with its description in matrix-form and the user is allowed to perform changes (add or remove transitions or change rates). When a arc is set between two sub-model the window shown in Figure 7 appears (the required parameters should be clear). Note that, although one could point out that a standard tool for the analysis of Jackson networks may present a more intuitive interface, we would like to remark that this is the same interface we would use for any stochastic model that can be solved by the algorithm presented in [10]. However, one could also decide to extend the MSI in order to be able to associate a specific symbol to some sub-models of the library, but this is out of the scope of this presentation.

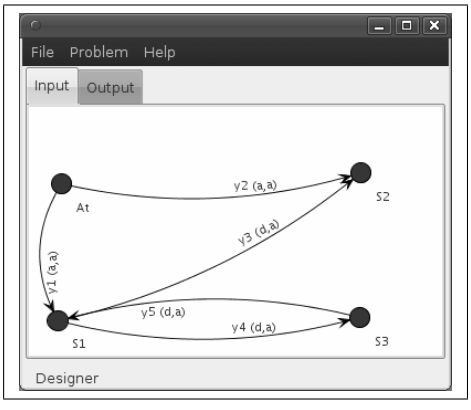



Fig. 6: Screen-shot of the model corresponding to the Jackson network of Figure 5.

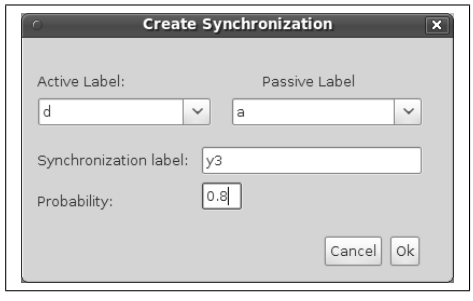



Fig. 7: Screen-shot of the window for the synchronisation details.

## 5. Final remarks

We have presented a tool that we are developing for the analysis of product-form models. It exploits some new results that appeared in product-form model theory and the algorithm presented in [10]. It has proved to be able to identify and compute several product-form results based on pairwise synchronisations, such as Jackson networks, G-networks, Akyildiz's results about product-form networks with blocking and other that have been described in [10]. Current research has three objectives: 1) allow for the specification of models with multiple incoming active transitions, exploiting the result presented in [11], 2) allow for the specification of models with multiple outgoing passive transitions, and 3) allow for the specification of models with regular but infinite structure. The last goal seems to be the hardest one. Indeed, an approximation is needed to truncate the model and we would like it to be decided dynamically in order to produce results which are correct within a specified bound.

## References

[1] I. F. Akyildiz. Exact analysis of queueing networks with rejection blocking. In H. G. Perros and T. Atliok, editors, *Proc. of the 1st Internat. Workshop on Queueing Networks with Blocking*, pages 19–29, North-Holland, Amsterdam, 1989.

[2] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios. Open, closed, and mixed networks of queues with different classes of customers. *J. ACM*, 22(2):248–260, 1975.

[3] E. Gelenbe. Product form networks with negative and positive customers. *Journal of Applied Prob.*, 28(3):656–663, 1991.

[4] P. G. Harrison. Reversed processes, product forms, non-product forms and a new proof of the BCMP theorem. In *Int. Conf. on the Numerical Solution of Markov Chains (NSMC 2003), Urbana IL, USA, September 2-5 2003*, pages 289–304, September 2003.

[5] P. G. Harrison. Turning back time in Markovian process algebra. *Theoretical Computer Science*, 290(3):1947–1986, January 2003.

[6] P. G. Harrison. Compositional reversed Markov processes, with applications to G-networks. *Perform. Eval., Elsevier*, 57(3):379–408, 2004.

[7] P. G. Harrison and T. T. Lee. Separable equilibrium state probabilities via time reversal in markovian process algebra. *Theoretical Computer Science*, 346(1):161–182, 2005.

[8] J. Hillston. *A Compositional Approach to Performance Modelling*. PhD thesis, Department of Computer Science, University of Edimburgh, 1994.

[9] F. Kelly. *Reversibility and stochastic networks*. Wiley, New York, 1979.

[10] A. Marin and S. Rota Bulò. A general algorithm to compute the steady-state solution of product-form cooperating Markov chains. In *Proc. of MASCOTS 2009*, pages 515–524, London, UK, September 2009.

[11] A. Marin and M. G. Vigliotti. A general result for deriving product-form solutions of markovian models. In *Proc. of First Joint WOSP/SIPEW Int. Conf. on Perf. Eng.*, San Josè, CA, USA, To appear.

[12] R. Tarjan. Depth-first search and linear graph algorithms. *SIAM J. on Computing*, 1(2):146–160, 1972.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 325 – 334

# Modeling Nonlinear Oscillatory System under Disturbance by Means of Ateb-functions for the Internet

IVANNA DRONIUK [a] MARIA NAZARKEVYCH [b]

[a] Automated Control Systems Department
Institute of Computer Science
Lviv National Polytechnic University
*nazarkevich@mail.ru*

[b] Automated Control Systems Department
Institute of Computer Science
Lviv National Polytechnic University
*ivanna.droniuk@gmail.com*

**Abstract:** This paper shows the calculation method based on formulas for asymptotic methods of solving differential equations and is widely used for researching the oscillatory systems. Analytical expressions for the solutions of differential equations that describe oscillatory systems are based on the Ateb-functions theory. Application the presented formulas for modeling traffic in computer networks is considered.

**Keywords:** oscillatory systems, network traffic

## 1. Introduction

Corporate computer networks are playing a greater role for management efficiency and success of various organizations. Thus, practically every such network displays the general trend of increasing number of users, volumes of current information and the consequent deterioration of the network traffic and the deteriorating quality of network services. This enables to conduct a research of network properties, not only in the mode of operational monitoring, but also a deeper theoretical study - in particular, to predict their behavior. In addition, it also implies researching and modeling the network traffic.

The number of works [1] is dedicated to problems of mathematical modeling and numerical study of computer networks. Queueing theory apparatus, Markov chain, the theory of tensor analysis [2] are applied in the modelling.

This article offers a different approach to modeling of computer networks, namely, considering the network as a nonlinear oscillatory system. It is known that the number of users in telephone networks, traffic on the Internet is subject

to certain repeated daily, weekly and other variable rates. Such oscillatory changes in network traffic allow to consider computer networks as some oscillatory systems and apply differential equations for their modeling [3], describing oscillatory movements. In the article [4] we considered and modelled free oscillation in the system, this study deals with oscillatory system under disturbance.

## 2. Problem Definition

Let`s consider the nondisturbing differential equation of fluctuations

$$\ddot{x} + f(x) = 0 \tag{1}$$

Let`s assume $x(t)$ is the load node, depending on time, $f(x)$ is defined outside the function.

It is assumed that external force is proportional to load in a node to some degree *v*. Then

$$f(x) = \lambda^2 x^{\nu} \tag{2}$$

where

$$\lambda > 0, \quad \nu = \frac{2\nu_1 + 1}{2\nu_2 + 1}, \quad (\nu_1, \nu_2 = 0, 1, 2, ...) \tag{3}$$

Let`s assume that the system received small disturbance. Then the equation that describes the fluctuations of the disturbing system can be represented as

$$\frac{d^2x}{dt^2} + \lambda^2 x^{\nu} = \varepsilon F\left(x, \frac{dx}{dt}\right) \tag{4}$$

where $\varepsilon > 0$ is small parameter and $F\left(x, \frac{dx}{dt}\right)$ is continuous function.

## 3. Construction of average solution for the disturbing nonlinear oscillatory system

Let`s apply the method of constructing the average solution for the system (4), applying the Ateb-function. Suppose the function $F(x, y)$ is continuous or piecewise continuous on a specified interval and may be represented by a polynomial or an expression with degrees relative to variables $x, \frac{dx}{dt}$ of degree not higher than $N$. Let`s write differential equation (4) in the form of system first order equations

$$\begin{cases} \dfrac{dx}{dt} + y = 0 \\ \dfrac{dy}{dt} - \lambda^2 x^{\nu} = -\varepsilon F(x, y) \end{cases} \tag{5}$$

Let`s make replacement of variables in the system of equations (5), using Ateb-functions by the formulas [5]

$$\begin{cases} x = a\, Ca(\nu, 1, \varphi) \\ y = a^{\mu} h\, Sa(1, \nu, \varphi) \end{cases} \tag{6}$$

where $\mu, h, \nu$ are constants, which are calculated using formula (3) and:

$$\mu = \frac{1+\nu}{2}, \quad h^2 = \frac{2\lambda^2}{1+\nu}.$$

Now follows to the amplitude-phase variables $a$ and $\varphi$. As a result, we get a system of equations:

$$\frac{da}{dt} = \varepsilon \frac{h\, Sa(1,\nu,\varphi)}{\lambda^2 \omega(a)} \cdot F\left(a \cdot Ca(\nu, 1, \varphi); a^{\mu} h\, Sa(1, \nu, \varphi)\right) \equiv \varepsilon\, A(a, \varphi, \lambda),$$

$$\frac{d\varphi}{dt} = \frac{\omega(a)}{L} + \varepsilon \frac{Ca(\nu, 1, \varphi)}{\lambda^2 \cdot L \cdot a \cdot \omega(a)} \cdot F\left(a \cdot Ca(\nu, 1, \varphi); a^{\mu} h\, Sa(1, \nu, \varphi)\right) \equiv \frac{\omega(a)}{L} + \varepsilon\, B(a, \varphi, \lambda), \tag{7}$$

where

$$L = \frac{2B(p,q)}{\pi \cdot h \cdot (1+\nu)}, \quad \omega(a) = a^{\frac{\nu-1}{2}}, \quad p = \frac{1}{2}, q = \frac{1}{1+\nu}.$$

Functions $Ca(\nu,1,\varphi)$, $Sa(1,\nu,\varphi)$ have period $\Pi$, where

$$\Pi(1,\nu) = \Pi(\nu,1) = B\left(\frac{1}{2}, \frac{1}{\nu+1}\right).$$

Amplitude-phase variables $a$ and $\varphi$ are selected by parameter $L$ in a way that functions $A(a,\varphi)$, $B(a,\varphi)$ in the system of equations (6) stay periodic relative to argument $\varphi$ with period $2\pi$. Then the functions $A(a,\varphi), B(a,\varphi)$ can be represented as a finite Fourier series. In determining the expansion coefficients it is necessary to note that the arguments of functions $A(a,\varphi)$ and $B(a,\varphi)$ are Ateb-functions $Ca(\nu,1,\varphi)$ and $Sa(1,\nu,\varphi)$ which satisfy algebraic identities

$$\left(Ca(\nu,1,\varphi)\right)^{1+\nu} + \left(Sa(1,\nu,\varphi)\right)^2 \equiv 1 \tag{8}$$

that is

$$Sa(1,v,\varphi)=\sqrt{1-Ca^{v+1}(v,1,\varphi)} \qquad -1\le Ca(v,1,\varphi)\le 1,\quad 0\le\varphi\le\pi$$

$$Sa(1,v,\varphi)=\sqrt{1-Ca^{v+1}(v,1,\varphi)} \qquad 1\ge Ca(v,1,\varphi)\ge -1,\quad 0\le\varphi\le 2\pi$$

or

$$Sa(1,v,\varphi)=(-1)^{j+1}\sqrt{1-Ca^{v+1}(v,1,\varphi)} \tag{9}$$

where $j=1$ corresponds to the segment $-1\le Ca(v,1,\varphi)\le 1$, and $j=2$ corresponds to the segment $1\ge Ca(v,1,\varphi)\ge -1$.

Using the ratio (9), let`s write down the system of equations (7) as

$$\begin{cases} \dfrac{da}{dt}=\varepsilon\ A(a,\varphi) \\ \dfrac{d\varphi}{dt}=\dfrac{\omega(a)}{L}+\varepsilon\ B(a,\varphi) \end{cases} \tag{10}$$

where

$$A(a,\varphi)=\frac{(-1)^{j+1}h}{\lambda^2\omega(a)}\sqrt{1-Ca^{v+1}(v,1,\varphi)}\cdot f\left(a\cdot Ca(v,1,\varphi);\ (-1)^{j+1}ha^{\mu}\sqrt{1-Ca^{v+1}(v,1,\varphi)}\right),$$

$$B(a,\varphi)=\frac{Ca(v,1,\varphi)}{\lambda^2\cdot L\cdot a\cdot\omega(a)}\cdot f\left(a\cdot Ca(v,1,\varphi);\ (-1)^{j+1}ha^{\mu}\sqrt{1-Ca^{v+1}(v,1,\varphi)}\right) \tag{11}$$

To solve the system of equations (10) let`s use the expansion in Fourier series.

## 4. Construction of average solution for the perturbed nonlinear oscillatory system by expansion in Fourier series

Let`s write down the overall decomposition of functions $A(a,\varphi)$ and $B(a,\varphi)$ from expressions (10) and (11) in a finite Fourier series:

$$A(a,\varphi)=\frac{1}{2}\overline{A}_0(a)+\sum_{\substack{k\neq 0\\ k=-M}}^{M}\left[\overline{A}_k(a)Cosk\varphi)+\overline{\overline{A}}_k(a)Sink\varphi)\right]$$

$$B(a,\varphi)=\frac{1}{2}\overline{B}_0(a)+\sum_{\substack{k\neq 0\\ k=-M}}^{M}\left[\overline{B}_k(a)Cosk\varphi)+\overline{\overline{B}}_k(a)Sink\varphi)\right] \tag{12}$$

Let`s build average solution as the variable $\varphi$ is responsible for the small rapid fluctuations and variable $a$ is responsible for the large fluctuations in

ampoules. Therefore, the variable can be excluded from the right parts of equations (10) and (11), using the decomposition in line by degrees of a small parameter $\varepsilon$ [6]. For this purpose let`s introduce new variables for the formulas

$$\begin{cases} a = b + \varepsilon U_1(b,\theta) + \varepsilon^2 U_2(b,\theta) + ..., \\ \varphi = \theta + \varepsilon V_1(b,\theta) + \varepsilon^2 V_2(b,\theta) + .... \end{cases} \tag{13}$$

New variables $b$ and $\theta$ are solutions to the system of equations

$$\begin{cases} \dot{b} = \varepsilon \lambda_1^{(1)}(b) + \varepsilon^2 \lambda_2(b) + ... \\ \dot{\theta} = \dfrac{\omega(b)}{L} + \varepsilon \beta_1(b) + \varepsilon^2 \beta_2(b) + .. \end{cases} \tag{14}$$

Let`s define coefficients $\alpha_i$ and $\beta_i$ $(i = 1,2,...)$ in the way to make the functions $U_1(b,\theta)$ and $V_1(b,\theta)$ periodic relative to $\theta$ with period $2\pi$. Substitute expressions (13) in the system of equations (7), then write down the right and left parts of the obtained ratios in a series of small parameters and equate the coefficients by equal degrees of this parameter. We receive a system of equations concerning the functions $U_1(b,\theta)$ i $V_1(b,\theta)$.

In the first approximation of the parameter $\varepsilon$ we have

$$\frac{\omega(b)}{L} \cdot \frac{\partial U_1}{\partial \theta} = A(b,\theta) - \alpha_1(b) \tag{15}$$

$$\frac{\omega(b)}{L} \cdot \frac{\partial V_1}{\partial \theta} = B(b,\theta) + \frac{\partial \omega(b)}{\partial b} \cdot U_1(b,\theta) - \beta_1(\theta) \tag{16}$$

The equation (15) will have $2\pi$ periodic solution relative to $\theta$, in case when the right part satisfies the condition

$$\int_0^{2\pi} \left[ A(b,\theta) - \alpha_1(b) \right] d\theta = 0 \tag{17}$$

We receive function $A(b,\theta)$ when substituting new variables (13) into the first expression (12), that is, its decomposition into Fourier series. Then it implies from the expression (17), that in the first approximation we get

$$\alpha_1(b) = \frac{1}{2} A_0(b) \tag{18}$$

The solution to equation (16) according to formula (12) looks like

$$U_1^{(1)}(b,\theta) = \frac{L}{\omega(b)} \cdot \sum_{\substack{k \neq 0 \\ k=-M}}^{M} \frac{1}{k} \left[ \overline{A_k} \sin k\theta - \overline{\overline{A_k}} \cos k\theta \right] + U_1^{\,0}(b). \tag{19}$$

If the right part of the differential equation (16) satisfies the equation

$$\int_0^{2\pi}\left[B(b,\theta)+\frac{\partial\omega(b)}{\partial b}\cdot U_1(b,\theta)-\beta_1(b)\right]d\theta=0 \tag{20}$$

in this case, the solution to the given equation is periodic.

Analogous to the previous, taking into account expression decomposition function $B(b,\theta)$ into trigonometric series (12) and expression (21) we can define from the condition (19) that

$$\beta_1(b)=\bar{B}^0(b)+\frac{\partial\omega(b)}{\partial b}\cdot U_1(b,\theta). \tag{21}$$

The solution to equation (19) looks like

$$V_1^{(1)}(b,\theta)=\frac{L}{\omega(b)}\sum_{\substack{k\neq 0\\ k=-M}}^{M}\left\{\left[\bar{B}(b)-\frac{L}{\omega(b)}\frac{\partial\omega(b)}{\partial b}\sum_{\substack{k\neq 0\\ k=-M}}^{M}\frac{1}{k}\bar{\bar{A}}(b)\right]\cdot\frac{\cos\theta}{k}+.\right.$$

$$\left.\left[\bar{\bar{B}}(b)+\frac{L}{\omega(b)}\frac{\partial\omega(b)}{\partial b}\sum_{\substack{k\neq 0\\ k=-M}}^{M}\frac{1}{k}\bar{A}(b)\right]\cdot\frac{\sin k\theta}{k}\right\}+V_1(b) \tag{22}$$

The functions $U^{(n)}$, $V^{(n)}$, $\alpha^{(n)}$, $\beta^{(n)}$ are similarly defined for $n=2,3,\ldots$.

Integration constants $U^{(n)}(b)$, $V^{(n)}(b)$ are defined in the way to satisfy the initial conditions given for equations (5), that is

$$a\big|_{t=t_0}=b\big|_{t=t_0}=b^{(0)},\quad \varphi\big|_{t=t_0}=\theta\big|_{t=t_0}=\theta^{(0)} \tag{23}$$

Expressions (15) correspond to the first approximation by degrees $\varepsilon$. Indeed, substituting values (15) into the right side of equation (10) and (11), we receive their solutions up to first order values inclusively.

## 5. Simulation of traffic in the network-based solution built for the perturbed oscillatory nonlinear system

In some cases for computer network simulation it is possible to choose value $x(t)$ as the amount of external traffic for a given node at any given time. In the case when modeling the communication networks $x(t)$ is defined as the average number of connections in the switches for a given node connection at any given time. In this paper we carry out simulation of the computer network and will further define $x(t)$ as a total value of the incoming and outgoing traffic in a computer network node. In order to simulate traffic in a computer network with consideration of disturbance it is necessary to use a system of differential

equations (15), (16) with initial conditions (23) and its obtained solutions (19), (22). The defined solutions need to be substituted into the formula (13). Then we can get the values of amplitude-phase variables. The next step is to calculate values $x$ and $y$ according to formulas (6).

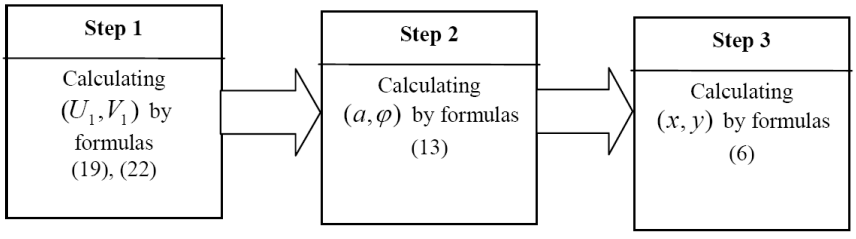



Fig. 1. Stages for calculating the solution to differential equation (4)

Currently we implemented the calculation in Step 1. We will show our algorithm for computing in Step 1. Block diagram of the calculation method is shown in Fig.1. Initially, we input tabulation step and parameters of Ateb-functions $n$ and $m$. After entering the parameters $n$ and $m$ we check the conditions of periodicity (17). Calculation are conducted at the interval $[0, \pi]$ with some given tabulation step. Then the calculations are performed by formulas (19), (22). Calculation of series sum is conducted according to a given precision.

Calculated function values are output into the file or displayed on the screen and then stored in an array to build functions graph. Afterwards, we choose the next point. If this point belongs to the interval, we return to the calculations, if the function is tabulated through the whole interval, the results of calculations are displayed.

Let`s separate the calculation of coefficients $\overline{A}_k$, $\overline{\overline{A}}_k$ and $\overline{B}_k$, $\overline{\overline{B}}_k$ as a separate unit $k = (-M, ..., M; k \neq 0)$. For their calculation we used methods of approximate calculation of double integrals and dichotomous search method of function [7] zeros. Thus, the calculation method in Step 1 was implemented on the basis of decomposition into Fourier series. The results of calculations are presented in Tab.1.

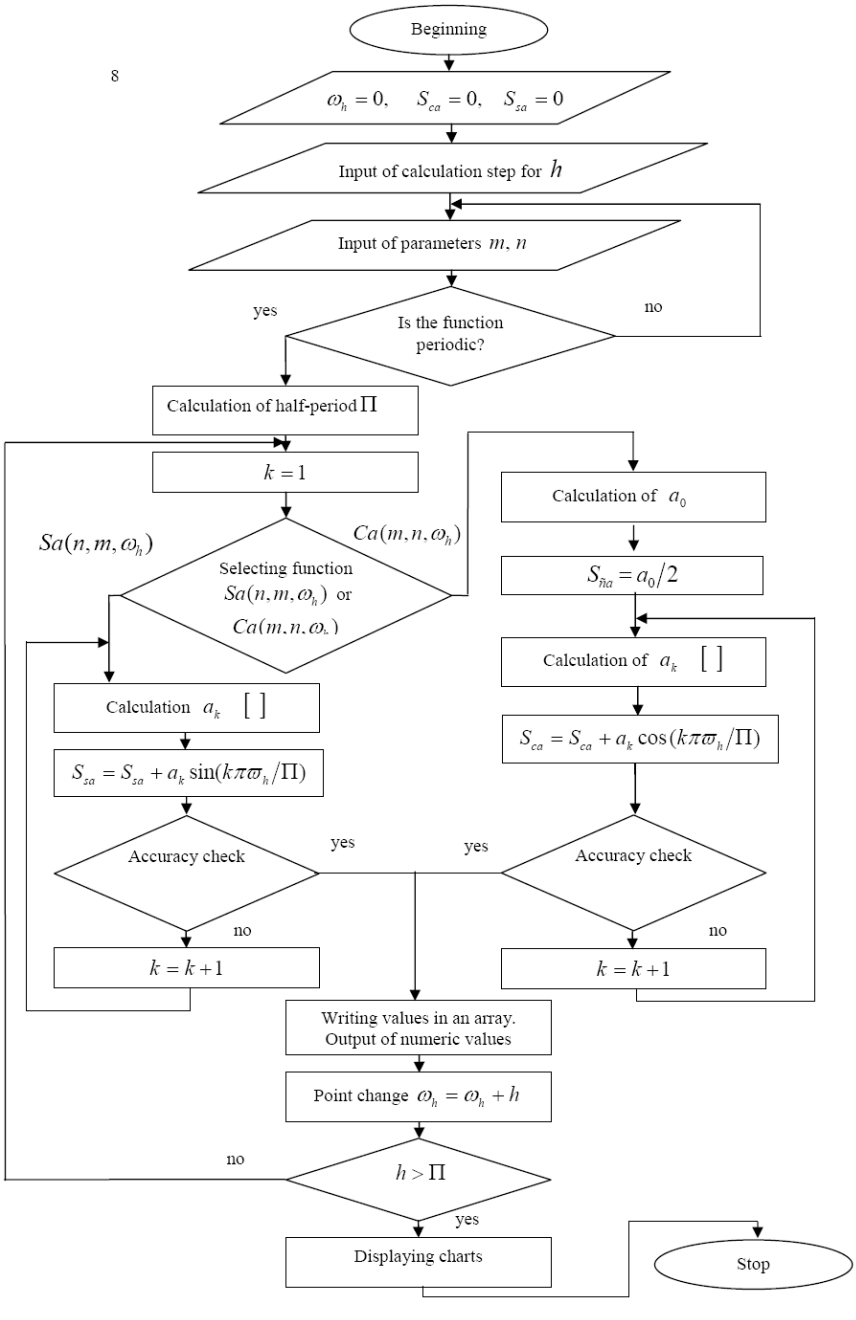



Fig.2. Block diagram of the calculation method of Step 1

**Tab 1.** Results of simulation step 1

| $t$ | $\nu$ =0.166 | $\nu$ =0,2 | $\nu$ =0,6 | $\nu$ =0,833 |
|---|---|---|---|---|
| 0,01 | 1,72E-02 | 1,67E-02 | 1,25E-02 | 1,09E-02 |
| 0,05 | 8,68E-02 | 0,0842945 | 6,26E-02 | 5,46E-02 |
| 0,1 | 0,176491 | 0,171086 | 0,1258645 | 0,109423 |
| 0,15 | 0,269343 | 0,2607955 | 0,1899825 | 0,164684 |
| 0,2 | 0,365641 | 0,353689 | 0,2552525 | 0,220551 |
| 0,25 | 0,4656 | 0,44999 | 0,321898 | 0,277191 |
| 0,3 | 0,569499 | 0,549954 | 0,390126 | 0,3347675 |
| 0,35 | 0,677875 | 0,654013 | 0,4601335 | 0,39344 |
| 0,4 | 0,792004 | 0,7631225 | 0,5321165 | 0,453366 |
| 0,45 | 0,915224 | 0,879679 | 0,6062905 | 0,514707 |
| 0,5 | 1,058303 | 1,0107085 | 0,6829385 | 0,5776285 |
| 0,55 | 1,236163 | 1,1458175 | 0,746344 | 0,6292285 |

## 6. Conclusions

Suggested methods of calculations are based on formulas for asymptotic methods of solving systems of differential equations and are widely used for researching the oscillatory systems. Analytical expressions for the solutions of differential equations systems that describe oscillatory movement are based on the Ateb-functions theory. In the paper we derived analytical formulas for the solution of differential equations with small disturbance. We propose to apply the presented formulas for modeling traffic in computer networks. Previous numerical simulations of traffic on the Internet based on the theory Ateb-functions were conducted without taking into account the disturbances [4]. For incorporation of disturbance we derived analytical expressions, which are planned to be applied for modeling traffic on the Internet. Currently we are developing methods for the numerical representation of derived analytical expressions, followed by application of developed methods for modeling and forecasting the traffic on the Internet.

# References

[1] Steklov V.K. *Construction the telecommunication systems*.- Kiev: "Technics", 2002.- 792p.

[2] Popovsky V.V. *Mathematical bases the theory of the telecommunication systems*.- Kharkov, the CMIT Company, 2006. - 563 p.

[3] Anisimov I.O. *Oscillation and waves*. - Kiev: "Technics", 2003.- 280 p.

[4] Ivanna Droniuk, Maria Nazarkevich, Roman Koshulinsky Modeling of traffic fluctuations in computer networks using Ateb-functions // In Proc. 16th Polish teletraffic

symposium 2009. Technical University of Lodz, Lodz, Poland, September 24-25. P.37-41

[5] Vozny A.M. Application Ateb-functions for constructions of decision one class substantial nonlinear differential equations. – Dopovidi Ukranian SSR, 1970, № 9, P. 971-974.

[6] Bogolubov N. N., Mitropolskiy Y. A. Asymptotic methods of solving differential equations. -IT IS: Science, 1974. – 503 p.

[7] Cegelyk G.G. Numeral methods.- Lvov: Ivan Franko National University, 2004.-408 p.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 335–354

# Performance Evaluation of a Medium Access Control Protocol for Wireless Sensor Networks Using Petri Nets

JALEL BEN-OTHMAN[a] SERIGNE DIAGNE[b] LYNDA MOKDAD[b]
BASHIR YAHYA[a]

[a]PRiSM Laboratory, University of Versailles
45, Avenue des Etas-Unis, 78000 Versailles, France
{bashir.yahya, jalel.ben-othman}@prism.uvsq.fr

[b]LACL Laboratory, Faculty of Science and Technology,
University of Paris XII
61 Avenue de General de Gaulle, 94010 Paris Cedex, France
{serigne.diagne, lynda.mokdad}@univ.paris12.fr

**Abstract:** The general approach to evaluate and analyze networking protocols and algorithms is to use simulation tools, like NS-2, OMNet++, OPNET, ...etc. An alternative method to simulation tools is to use formal modeling and analysis techniques (such as Petri Nets). These techniques allow the user to do both performance evaluation and model checking.

In this paper, we continue our previous work on modeling and evaluating our EQ-MAC protocol [1]. In this work, we introduce a Colored Petri Nets model for modeling and evaluating the EQ-MAC protocol. EQ-MAC protocol is an energy efficient and quality of service aware medium access protocol designed for wireless sensor networks.

To extract some results from the developed model, we used the GreatSPN and WNSIM tools. Results demonstrate the efficiency of our protocol. As well, this work demonstrates the possibility of using Petri Nets in modeling and evaluating of any other MAC protocols for wireless sensor networks.

**Keywords:** Energy Efficient MAC, QoS, Wireless Sensor Networks, Petri Nets.

## 1. Introduction

Recent advances in micro-electro-mechanical systems, low power highly integrated digital electronics, tiny microprocessors and low power radio technologies have created low-cost, low-power, and multi-functional sensor devices, which can observe and react to changes in physical phenomena of their surrounding environments. These sensor devices are equipped with a small battery, a radio transceiver,

a processing unit, and a set of transducers that used to acquire information about the surrounding environment. The emergence of such sensors has led engineers to envision networking of a large set of sensors scattered over a wide area of interest. A typical wireless sensor network consists of a number of sensor devices that collaborate to accomplish a common task such as environment monitoring and report the collected data, using the radio, to a center node (sink node). Wireless Sensor Networks (or WSNs in short) can serve many civil and military applications that include target tracking in battlefields [2], habitat monitoring [3], civil structure monitoring [4], and factory maintenance [5], etc. In these applications, reliable, and real time delivery of gathered data plays a crucial role in the success of the sensor network operation.

Provided that sensor nodes carry limited, generally irreplaceable, power source, then wireless sensor networks must have built-in trade-off mechanisms that enable the sensor network to conserve power and give the end user the ability of prolonging network lifetime at the cost of lower throughput and/or higher latencies. The energy constraints of sensor nodes and the need for energy efficient operation of a wireless sensor network have motivated a lot of research on sensor networks which led to the development of novel communication protocols in all layers of the networking protocol stack. Given that the radio transceiver unit considered as the major consumer of energy resources of the sensor node, specially when the radio transceiver is turned on all time, then a large amount of energy savings can be achieved through energy efficient media access control (MAC) mechanisms. For this reason, energy consideration has dominated most of the research at MAC layer level in wireless sensor networks [6].

However, the increasing interest in real time applications of sensor networks has posed additional challenges on protocol design. For example, handling real time traffic of emergent event triggering in monitoring based sensor network requires that end-to-end delay is within acceptable range and the variation of such delay is acceptable [7]. Such performance metrics are usually referred to as quality of service (QoS) of the communication network. Therefore, collecting sensed real time data requires both energy and QoS aware MAC protocol in order to ensure efficient use of the energy resources of the sensor node and effective delivery of the gathered measurements.

However, achieving QoS guarantees in sensor networks is a challenging task, because of the strict resource constraints (limited battery power, and data memory) of the sensor node, and the hostile environments in which they must operate [8].

This paper introduces a Petri Nets model for our EQ-MAC protocol [1]. EQ-MAC protocol is an energy efficient and quality of service aware MAC protocol for wireless sensor networks. EQ-MAC utilizes a hybrid approach of both sched-

uled (TDMA) and contention based (CSMA) medium access schemes. EQ-MAC differentiates between short and long messages; long data messages are assigned scheduled TDMA slots (only those nodes, which have data to send are assigned slots), whilst short periodic control messages are assigned random access slots. This technique limits message collisions and reduces the total energy consumed by the radio transceiver [6].
Perhaps the greatest advantage of EQ-MAC beside the efficient node's battery usage is its support for quality of service based on the service differentiation concept. The service differentiation is done through employing a queuing model consists of four different priority queues. This model allows sensor nodes to do some type of traffic management and provide extremely highest priority traffic a greater chance of acquiring the channel and hence rapidly served with minimum delay.
The rest of the paper is organized as follows. We present and discuss some related work in section II. Section III describes the EQ-MAC protocol approach. In section IV, we describe the components of the Petri Nets model designed for EQ-MAC protocol. Section V presents the results. Finally, we conclude the paper in section VI.

## 2. Related Work

Power management of the radio transceiver unit of a wireless device has gained significant importance with the emerging of wireless sensor networks since the radio unit is the major consumer of the sensor's energy [1]. It has been shown that the energy consumed in transmitting one bit is several thousand times more than the energy consumed in executing one instruction [9]. Recently, several MAC layer protocols have been proposed to reduce the energy consumption of the sensor's radio unit. Refer to [6] for some examples.
However, the increasing interest in real time applications and multimedia applications of sensor networks requires both energy and Quality of Service (QoS) aware protocols in order to ensure efficient use of the energy resources of the sensor node and effective delivery of the gathered measurements.
Perhaps the most related protocols to our protocol are presented in [10] and [11]. In [10], R. Iyer and L. Kleinrock developed an adaptive scheme for each sensor to determine independently whether to transmit or not so that a fixed total number of transmissions occur in each slot. The protocol accomplishes its task by allowing the base station to communicate QoS information to each sensor node within the network through a broadcasting channel, and by using the Gur Game mathematical paradigm, optimum number of active sensors can be dynamically adjusted.

The protocol makes tradeoffs between the required number of sensors that should be powered-up so that enough data is being collected in order to meet the required QoS and number of sensors that should be turned-off to save a considerable amount energy, and hence maximizing the network's lifetime. The concept of QoS in [10] and our EQ-MAC are completely different; in [10] the QoS is defined as the total number of transmissions that should occur in each slot in order to gather enough data. In other words, QoS in [10] is expressed as the quantity of gathered sensory data should be enough for the command center to make a decision, regardless of the delay requirement. (i.e. maximizing the protocol throughput, while minimizing energy consumption). while in EQ-MAC the QoS is defined as classing network traffic based on its importance into different classes in order to provide better service (in terms of delay and throughput) for certain traffic classes (e.g. real time traffic).

In [11], authors proposed Q-MAC scheme that attempts to minimize the energy consumption in a multi-hop wireless sensor network while providing quality of service by differentiating network services based on priority levels. The priority levels reflect the criticality of data packets originating from different sensor nodes. The Q-MAC accomplishes its task through two steps; intra-node and inter-node scheduling. The intra-node scheduling scheme adopts a multi-queue architecture to classify data packets according to their application and MAC layer abstraction. Inter-node scheduling uses a modified version of MACAW [12] protocol to coordinate and schedule data transmissions among sensor nodes.

Unlike Q-MAC, our EQ-MAC uses a more energy efficient way to coordinate and schedule data transmissions among sensor nodes through a hybrid approach utilizing both scheduled and non-scheduled schemes. A significant amount of energy could be saved through this approach.

In order to investigate the performance of networking protocols, there is a clear need to use simulation tools or formal methods to validate the protocol performance or functionality prior to implementing it in a real environment. Using formal modeling and analyzing techniques (such as Petri Nets) have the advantage to perform both performance evaluation and model checking. Such techniques are widely used in traditional networks. Using formal techniques in modeling wireless networking protocols presents many challenges, some of which are addressed in [13]. As we are going to use Petri Nets in modeling our protocol, then in the rest of this section, we briefly describe some work in modeling using Petri Nets. In [14], a high level Petri Net named as finite population queuing system Petri nets (FPQSPN) is introduced for molding and simulation of medium access control layer in computer networks. Authors in [15], proposed a Petri Net model for a formal verification of IEEE 802.11 PCF protocol. A Petri Net model is presented for the SMAC [16] pro-

tocol in [17]. The proposed model is based on Hierarchical Colored Petri Nets, and using Petri Net tools, some results of certain performance measures are evaluated. In the remaining sections, we describe our protocol, proposed Petri Net model, and performance evaluation results.

## 3. EQ-MAC PROTOCOL DESCRIPTION

Since we are going to model the EQ-MAC protocol[1][18] in this paper, we briefly describe this protocol in this section.
EQ-MAC protocol composed of two components; a clustering algorithm and a channel access mechanism.

### 3.1. Clustering Algorithm

Sensor network clustering is done through the execution of a modified version of Ext-HEED[19]. Ext-HEED is originally inspired by HEED algorithm [20]. Election of CHs is based on two main criteria; first the amount of residual energy of the node, thus a node with high residual energy has a higher chance to be elected and become a CH. Second criterion is the intra-cluster communication cost. This criterion used by nodes to determine the cluster to join. This is especially useful if a given node falls within the range of more than one CH.
The clustering algorithm achieves its task through the execution of the following four phases:

- ***Initialization phase***; Initially the algorithm sets a certain number of cluster heads among all sensors. This value is used to limit the initial cluster head announcements to the other sensors. As well each sensor sets its probability of becoming a cluster head.
- ***Repetition phase***; During this phase, every sensor node goes through several iterations until it finds the cluster head that it can transmit to with the least transmission power. Finally, each sensor doubles its cluster head probability value and goes to the next iteration of this phase. It stops executing this phase when its cluster head probability reaches 1.
- ***Optimization phase***; In this phase, all uncovered nodes must run the original HEED algorithm[20] to elect some extra cluster heads. Each uncovered node selects a node with the highest priority in its neighborhood (including it self) as a cluster head to cover itself. Reducing cluster head count reduces the

inter-cluster head communication and thus prolongs the network lifetime and limits the data collection latency.

- ***Finalization phase***; During this phase, each sensor makes a final decision on its status. It either picks the least cost cluster head or pronounces itself as a cluster head.

### 3.2. Channel Access Mechanism

The channel access protocol is composed of two sub-protocols: Classifier MAC (C-MAC), and Channel Access MAC (CA-MAC). The two sub-protocols are described below.

#### 3.2.1. Classifier MAC (C-MAC)

C-MAC protocol uses a modified version of the queuing architecture of Q-MAC [11]. C-MAC classifies packets based on their importance and stores them into the appropriate queue. The source node knows the degree of importance of each data packet it is sending which can be translated into predefined priority levels. The application layer sets the required priority level for each data packet by appending two extra bits at the end of each data packet. The number of bits used to distinguish priorities could be set according to the number of priority levels required.
The queuing architecture of the C-MAC is composed of four queues (see Fig 1). Each packet is placed in one of the four queues -high (instant queue), medium, normal, or low- based on the assigned priority. During transmission, the CA-MAC gives higher priority queues absolute preferential treatment over low priority queues.

#### 3.2.2. Channel Access MAC (CA-MAC)

The CA-MAC sub-protocol uses a hybrid mechanism that adapts scheduled and unscheduled schemes in an attempt to utilize the strengths of both mechanisms to gain a save in energy resources of the sensor node, and hence prolonging the lifetime of the sensor network. CA-MAC provides scheduled slots with no contention (based on TDMA) for data messages and random access slots (based on CSMA/CA) for periodic control messages. In the design of our protocol, we assume that the underlying synchronization protocol can provide nearly perfect synchronization, so that synchronization errors can be neglected.
CA-MAC classifies sensor nodes within the sensor network into two types: normal

sensor nodes, and head nodes. The head node is responsible for controlling the channel access between sensor nodes within the cluster and collects sensory data from them, as well as reporting gathered data into the Base Station (BS). This classification is done through the execution of the clustering algorithm at the beginning of each round.
The communication process is composed of two steps; transferring data from sensor nodes to cluster head (intra-cluster communication), then from cluster heads to the BS. Before going further in describing our protocol, we define some assumptions:

- All sensor nodes are in the radio range of the BS.
- Always we consider the number of heads generated by the cluster head after each round is fixed.
- The clustering algorithm is repeated every certain period (in our implementation is set to 15 minutes, this time is chosen based on the relax interval of the alkaline batteries) to re-elect new CHs in order to evenly distribute the consumed energy between sensor nodes. (i.e. the role of CH is rotated between nodes according to the residual energy of each node)

Refer to our papers [1] and [18] for more details.

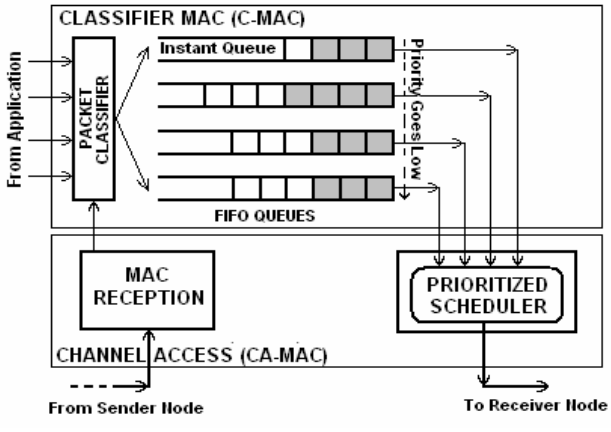



Fig. 1. EQ-MAC Structure

**Intra-Cluster Communication** The channel access mechanism in CA-MAC is based on dividing communication time into frames (see Fig. 2), which are controlled by the head node. The frame is composed of two slots: mini slot and

dynamic normal slot. Mini-slot is used to transmit and receive control signals, and consists of three parts; Frame Synchronization (SYNC), Request, and Receive Scheduling. Dynamic normal slot is used to control the transmission of the gathered data to the head node. The frame length is dynamic (i.e. the number of time slots is increased or decreased according to the number of nodes that have data to send).
CA-MAC accomplishes its task through the following four phases: *Synchronization*, *Request*, *Receive Scheduling*, and *Data Transfer*. Nodes that have data to send should content for the channel during the Request phase and send their requests along with the appropriate priority level of its traffic to the head node. (The contention interval should be long enough to give all sensor nodes which have data to transmit a chance to send their requests). Then, sensor nodes use the TDMA slots during the data transfer phase to send their data packets to CHs. Sensor nodes that have no data to transmit go to sleep directly after the end of the mini-slot.More details are given below about the operation of the CA-MAC in each phase:

- ***Synchronization phase***: At the beginning of each frame, the CH broadcasts a SYNC message to all sensor nodes within its cluster - all sensor nodes should be in receive mode during this phase to be able to capture the SYNC message. The SYNC message contains synchronization information for the packet transmission.
- ***Request phase***: During this phase, sensor nodes that have data to transmit content for the channel in order to acquire the access to send its request to the CH along with the required priority level.
- ***Receive Scheduling phase***: The CH broadcasts a scheduling message to all sensor nodes within its cluster that contains the TDMA slots for the subsequent phase ”data transfer phase”. All sensor nodes that have no data to transmit or receive should turn their radios transceivers off and enter sleep mode until the beginning of next frame. Making sensor nodes sleep early results in significant save in energy.
- ***Data Transfer phase***: In this phase, sensor nodes use the TDMA slots to transmit their data to the CH or to communicate with their neighbors.

**Reporting Data to Base Station** Accessing the channel to report data to the base station nearly uses the same frame structure used in intra-cluster communication. As the number of CHs is fixed after each execution of the clustering algorithm, then the BS schedules directly the cluster heads, and distributes the time slots between

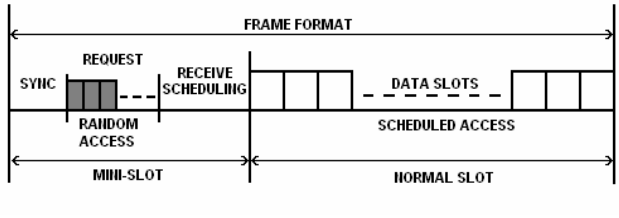



Fig. 2. Frame Format Structure

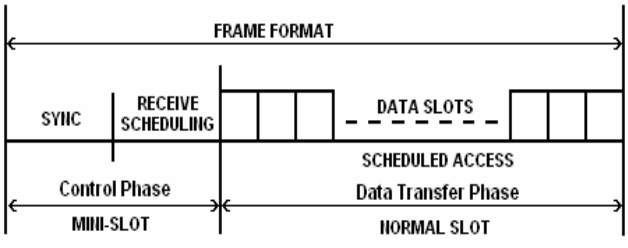



Fig. 3. Frame Structure:Used for data communication between CHs and BS

CHs. We assume that all CHs have data to report to the BS. As a result, the random access period is removed, and the frame structure becomes as shown in 3. The communication procedure is done through the following phases:

- ***Synchronization phase***: At the beginning of each frame, the BS broadcasts a SYNC message to all sensor nodes - all sensor nodes should be in receive mode during this phase to be able to capture the SYNC message.
- ***Receive Scheduling phase***: as CHs are announced after executing the clustering algorithm, then the BS knows the current elected CHs. As a result, there is no need for CHs to content for the channel to acquire the access to send their request messages. Moreover, we assume that CHs always have data to report to the BS. The BS broadcasts a scheduling message to all CHs that contains the TDMA slots for the subsequent phase "data transfer phase".
- ***Data Transfer phase***: In this phase, CHs use the received TDMA schedule to transmit their collected data to the BS.

## 4. Modeling the EQ-MAC protocol with Petri Nets

In this section, we describe the proposed Petri Net model for the EQ-MAC protocol. We consider two cases. In the first case (Case-1), we consider the frame has

enough data slots to accommodate all requests arriving from different sensor nodes in the networks (see figure 4). The second case (Case-2) considers a frame where data slots are not enough to accommodate all nodes requests number, and therefore only some part of the arrived requests are processed (see figure 5).

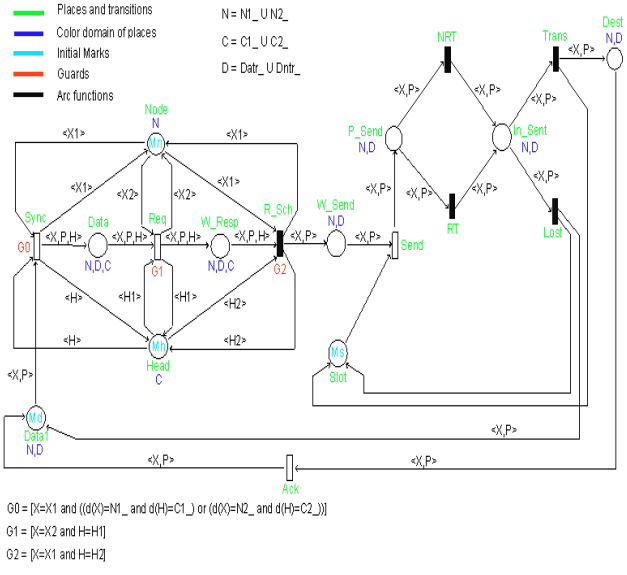



Fig. 4. Case-1: Data slots are enough

### 4.1. Describing Elements of the Model

In this section, we describe the components of our models. We use the SWN (Stochastic Well Formed Petri Net) model for our system. By referring to figures 4 and 5, we have various elements:

1. **Places:** A place in a Petri net model contains the various resources of the system. In our system, we have three kinds of resources: stations, slots, and data. The model contains **ten places** which are:

   (a) **Data**: This place contains the data to be sent,

   (b) **Data1**: After the synchronization, this place contains the data for which the resource requests should be sent,

   (c) **Node**: It contains all the nodes (stations) of the system. The nodes of various clusters are distinguished by color classes,

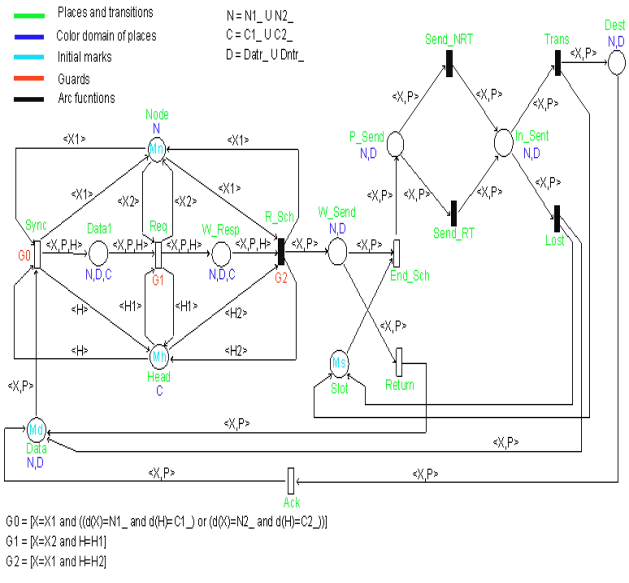



Fig. 5. Case-2: Data slots are not enough

(d) **Head**: It contains the cluster heads. Note that a cluster head is a station which can communicate with all nodes in the cluster,

(e) **W_Resp**: It is the place in which the nodes wait for the response of the cluster head, after the transmission of their requests,

(f) **Slot**: It is the place which carries the available slots for the packets transmission,

(g) **W_Send**: In this place the nodes wait to receive slots

(h) **P_Send**: It is the place in which the nodes are in the last step before the transmission of their packets,

(i) **In_Sent**: The tokens in this place are the data in the support of transmission,

(j) **Dest**: This place models the queue of the sink node.

2. **Transitions:** A transition is an action. In our system, we have many actions. These actions are modeled as follows:

(a) **Sync**: This transition represents the broadcast of SYNC messages by cluster heads,

(b) **Req**: It is the submission of resource requests to cluster heads,

(c) **R_Sch**: This transition models the broadcast of the receive scheduling message by cluster heads,

(d) **End_Sch**: It represents the attribution of slots to nodes,

(e) **Return**: This transition represents the packets that are are returned back to nodes, because of slot insufficiency,

(f) **Send_RT**: It is the beginning of real time packets transmission,

(g) **Send_NRT**: It is the beginning of non-real time packets transmission,

(h) **Trans**: This transition represents the arrival of packets in the sink node,

(i) **Lost**: When a packet is lost in the network a message is sent to the sender,

(j) **Ack**: When a packet arrives to a sink node an Ack message is sent to the sender.

3. **Color classes:** In a SWN, the color classes represent the resources of the system. Each resource is modeled by a color class. In each color class, we can have a subclass. In our system, the places contain the follow resources: nodes, cluster heads, slots, and data packets. We have only three color classes. Slots are defined as neutral tokens.

   (a) The color class of data packets (**D**): It contains all packets that are sent by nodes in the network. We have two subclasses in the color class D:

      i. The first subclass contains the real time packets (**Datr_**),

      ii. The second subclass contains the non-real time packets (**Dntr_**)

   (b) The color class of nodes (**N**): This color class contains all nodes in the system. Each subclass of this color class represents the nodes of a cluster. Then we have so many subclasses as we have many clusters. If the number of clusters in the network is n, then the subclasses are denoted by Ni_, where (1 =¡ i ¡= n),

   (c) The color class of cluster heads (**C**): It contains the cluster heads of the clusters that compose the network. In our system, each cluster contains one head and the heads are distinguish by different subclasses. We have many subclasses in the color class C as we have many clusters. If the number of clusters in the network is n, then subclasses are denoted by Ci_ (1 =¡ i ¡= n),

4. **Guards:** A guard is a function which gives the facility to add restrictions in a model. In our model we have three guards:

(a) **G0** is concerning with the transition **Sync**. It prevents a CH from sending a synchronization message to a node of another cluster,

(b) **G1** is concerning with the transition **Req**. It prevents a node from sending a request when it has not data to send (X = X2) and a CH from receiving a request from a node that belongs to another cluster (H = H1),

(c) **G2** is concerning with the transition **R_Sch**. It prevents a CH from sending the scheduling message to nodes of another cluster (H = H2) and a node from receiving the scheduling message of other CHs (X = X1).

5. **Initial markings:** The initial marking defines the resources effectively available in the system. A marking is attributed to a place in the model. If we attribute an initial marking to a place, the tokens in this marking are the initial resources in this place before running the simulation. We have four initial markings, which are defined as follows:

   (a) The marking **Mn** contains all nodes in the network. It is the initial marking of the place **Node**.

   $$Mn = \sum_{i=1}^{n} Ni\_$$

   n represents the number of clusters

   (b) The marking **Mh** contains all CHs in the network. It is the initial marking of the place **Head**:

   $$Mh = \sum_{i=1}^{n} Ci\_$$

   n represents the number of clusters

   (c) The marking **Ms** is a marking of neutral tokens. It contains the initial number of data slots in the frame. It is the initial marking of the place **Slot**:

   $$Ms = m$$

   m represents the number of slots in the frame

(d) The marking **Md** contains all data packets of all nodes in the network. It is the initial marking of the place **Data**.

## 5. Performance Evaluation

After the model construction, we can extract some performance measures using Petri Net simulation tools. We have used GreatSPN [21] and WNSIM to carry out our simulation experiments. We investigate the performance of our protocol in terms of average delay and delivery ratio.

### 5.1. Simulation Setup

The simulated network is composed of 26 nodes including the base station. We test the protocol under two scenarios; in the first scenario (we call it case-1), we suppose that we have enough slots in the time frame, and then all packets will be processed and sent to the base station (sink node). Under the second scenario (we call it case-2), we suppose that that the time frame is short and hence the number of slots are not sufficient to accommodate all requests from different nodes in the network.
In both scenarios, we change the number of packets generated at source nodes from 1 to 10 packets. Here the real time traffic is set to 1/3rd of the non-real time traffic.

### 5.2. Performance Metrics

We use the *Average Delay*, and *Average Delivery Ratio* as performance metrics to assess our protocol. Each metric is computed as a function in the traffic load of the network, and is defined as fellow:

***Average Delay***: the average delay is defined as the average time between the moment a data packet is sent by the source and the moment that packet has been received by the base station (sink).

***Percentage Delivery Ratio (PDR)***: This metric is computed as the ratio of the total number of packets successfully received by the base station ($T_{SRPs}$) to the total number of packets transmitted by the sources $T_{TPs}$ multiplied by 100.

$$PDR = \frac{T_{SRPs}}{T_{TPs}} * 100$$

### 5.3. Simulation Results

In this section, we illustrate and discuss our results.

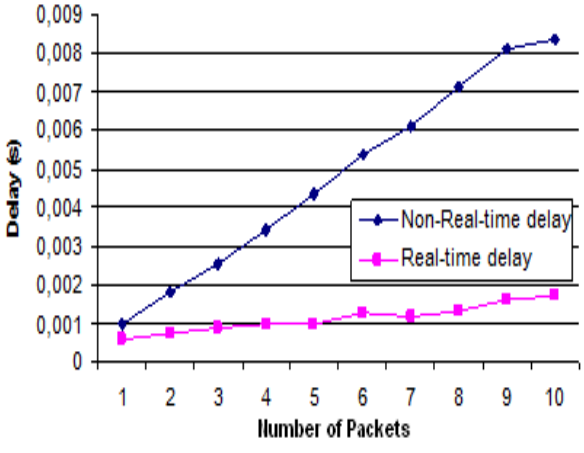



Fig. 6. Average Delay under Case-1

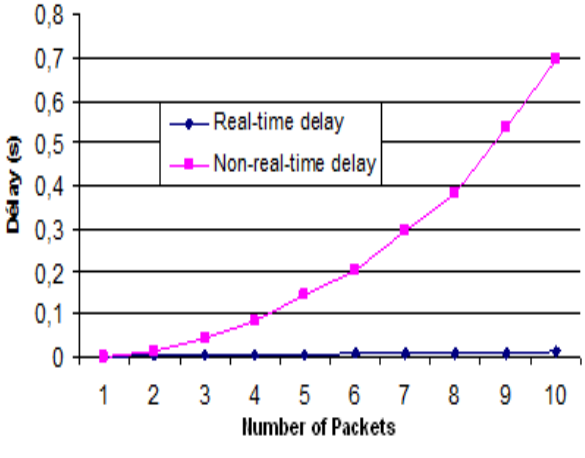



Fig. 7. Average Delay under Case-2

#### 5.3.1. Average Packet Delay

Figures 6 and 7 show the average packet delay for our protocol under scenarios one and two respectively. Obviously, the results indicate that EQ-MAC successfully differentiates network services. The higher priority packets are always accompanied with low latency.
Under case-1, where we have engouh number of slots, all data packets (either real time or non-real time) are processed, and all nodes get a chance to send their data to the cluster head. This explains why the delays in the first case (figure 6) are less than the delays in the second case (figure 7). When we compare the results of the two cases; in the second case, we note that the non-real time traffic takes longer time to be processed, this is because, in case-2, the number of slots per frame is limited, and hence many non-real time packets are buffered to gives high priority traffic (real time) absolute preferential treatment over low priority traffic (non-real time).

#### 5.3.2. Percentage Delivery Ratio

Figures 8 and 9 show the percentage delivery ratio of the packets successfully delivered to sink nodes under Case-1 and Case-2 respectively.
Under Case-1, we observe that the delivery ratio for both types of traffic is high and the the variation in the delivery ratio (as traffic rate increases) is within a short range (between 84.5 and 87.5), this is because we have enough number of time slots to accommodate the traffic from different nodes. As well, we note that the delivery ratio for both types of traffic is lower than 100, because there are lost packets in the network.
Under Case-2, we observe that the delivery ratio for non-real time traffic decreases significantly, this is due to the limited number of time slots per frame. In this case, the protocol gives higher priority to process the real time traffic by, and assigns any remaining slots to the non-real time traffic.

## 6. CONCLUSION

In this paper, we presented a Petri Net model for our protocol (EQ-MAC). The EQ-MAC is an energy efficient and QoS aware protocol introduced for wireless sensor networks. EQ-MAC combines the benefits of contention based and scheduled based protocols to achieve a significant amount of energy savings and offers QoS by differentiating network services based on priority levels. EQ-MAC enables

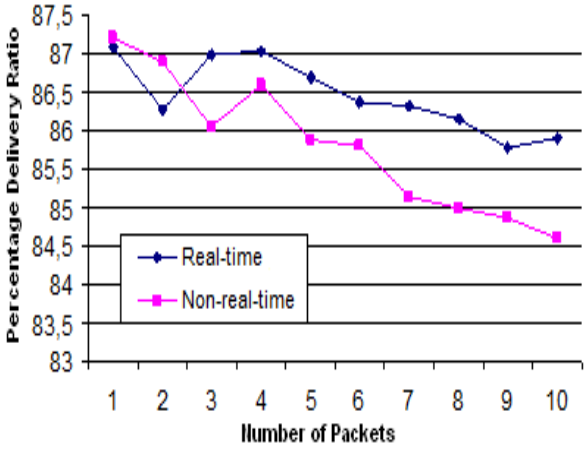



Fig. 8. Percentage Delivery Ratio under Case-1

only the nodes which have a data to transmit to access the channel according to their traffic priority levels; this avoids wasting slots by excluding those nodes which have no data to transmit from the TDMA schedule, and to switch nodes to sleep mode when they are not included in the communication process. Furthermore, prioritizing traffic according to its importance and criticality provides a greater chance for extremely highest priority nodes to access the channel and acquire the medium and hence rapidly served with minimum delay.

Using GreatSPN and WNSIM tools, we have implemented our Petri Net model and evaluated some performance measures. This work demonstrates and shows the capability of Petri Nets for modeling and evaluation of sensor networks.

The extracted results show the benefits of the EQ-MAC protocol, and can help in improving the design of real MAC protocols for wireless sensor networks.

## References

[1] B. Yahya, J. Ben-Othman, ”An Energy Efficient Hybrid Medium Access Control Scheme for Wireless Sensor Networks with Quality of Service Guarantees”, In the Proceedings of GLOBECOM’08, New Orleans, LA, USA, November 30 to December 4

[2] T. Bokareva, W. Hu, S. Kanhere, B. Ristic, N. Gordon, T. Bessell, M. Rutten and S. Jha ”,Wireless Sensor Networks for Battlefield Surveillance”,In Proc. of LWC - 2006.

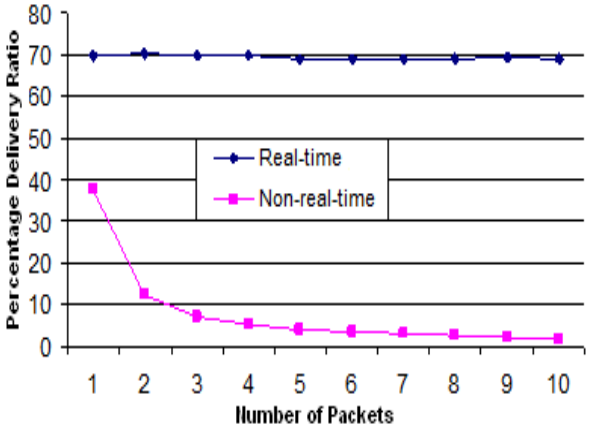



Fig. 9. Percentage Delivery Ratio under Case-2

[3] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring," in the Proceedings of ACM-WSNA Conf., Sep. 2002.

[4] N. Xu, S. Rangwala, K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, "A Wireless Sensor Network for structural Monitoring," in Proc. ACM SenSys Conf., Nov.2004.

[5] K. Srinivasan, M. Ndoh, H. Nie, H. Xia, K. Kaluri, and D. Ingraham, "Wireless Technologies for Condition-Based Maintenance (CBM) in Petroleum Plants," Proc. of DCOSS'05 (Poster Session), 2005.

[6] B. Yahya, J. Ben-Othman, "Towards a classification of energy aware MAC protocols for wireless sensor networks", Journal of Wireless Communication and Mobile Computing, Published online in Wiley InterScience, Feb. 4, 2009.

[7] M. Younis et.al, "On handling qos traffic in wireless sensor networks," Proc. of the HICSS, 2004.

[8] D. Chen, P.K. Varshney: QoS Support in Wireless Sensor Networks: A Survey. International Conference on Wireless Networks 2004: 227-233

[9] [1] V. Raghunatham, et al., "Energy-Aware Wireless Micro Sensor Networks," IEEE Signal processing magazine; March 2002; 40-50

[10] R. Iyer and L. Kleinrock, "QoS Control for Sensor Networks," in Proc. of ICC, May 2003.

[11] Yang Liu, Elhanany, I.; Hairong Qi," An energy-efficient QoS-aware media access control protocol for wireless sensor networks", Mobile Adhoc and Sensor Systems Conference, 2005.

[12] V. Bharghavan et. Al, MACAW: A Media Access Protocol for Wireless LANS," Proc. ACM SIGCOMM, vol. 24, no. 4 1994.

[13] Olveczky, P.C., Throvaldsen, S., "Formal Modeling and Analysis of the OGDC Wireless Sensor Network Algorithm in Real Time Maude, Lecture Notes in Computer Science, Vol. 448 (2007), 1611-3349.

[14] Capek, J., Petri Net Simulation of Non-deterministic MAC Layers of Computer Communication Networks, Ph.D. Thesis, Czech Technical University, 2003.

[15] Haines, R.; Clemo, G.; Munro, A., "Toward Formal Verification of 802.11 MAC Protocols: Verifying a Petri-Net Model of 802.11 PCF", IEEE 64th Vehicular Technology Conference, VTC-2006 Fall, 25-28 Sept. 2006 Page(s):1 - 5.

[16] W. Ye, J. Heidenmann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks,. in Proc. of IEEE INFOCOM, New York, NY, June 2002.

[17] Mohammad Abdollahi Azgomi and Ali Khalili, "Performance Evaluation of Sensor Medium Access Control Protocol Using Coloured Petri Nets", Proceedings of the First Workshop on Formal Methods for Wireless Systems (FMWS 2008).

[18] Bashir Yahya and Jalel Ben-Othman, "A Scalable and Energy-Efficient Hybrid-Based MAC Protocol for Wireless Sensor Networks", in the proceedingis of PM2HW2N'08, October 31, 2008, Vancouver, BC, Canada.

[19] [2] Hesong Huang; Jie Wu, " A probabilistic clustering algorithm in wireless sensor networks," Proc. Of 62nd IEEE VTC, 25-28 Sept., 2005 Page(s): 1796 - 1798

[20] [7] Younis and S. Fahmy. Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach. In Proceedings of IEEE INFOCOM, March 2004.

[21] http://www.di.unito.it/ greatspn/index.html

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 355–366

# The Throughput Maximization in the MIMO-OFDMA Systems

JERZY MARTYNA

Institute of Computer Science
Jagiellonian University
*Jerzy.Martyna@ii.uj.edu.pl*

**Abstract:** In this paper, we introduce a new method for the throughput maximization in the downlink multiple-input-multiple-output orthogonal frequency division multiple access (MIMO-OFDMA) in a single-cell multiuser environment with the channel side information of the transmitter. The throughput maximization was formulated as an optimization concept and the optimal results were obtained by solving the Kuhn-Tucker conditions. In order to find the optimal values we proposed an approach to solving this problem. With the use of the special properties of the throughput maximization, we introduced a scheduling algorithm that yields the optimal transmission schedule. We show that the throughput maximiztion resulting from the suggested scheduling algorithm is comparable to the simulation results.

**Keywords:** : MIMO-OFDMA systems, Throughput maximization

## 1. Introduction

The modern wireless mobile communication system requires a high robustness and a high spectral efficiency. Based on the OFDM (Orthogonal Frequency Division Multplexing) the orthogonal frequency division multiple access (OFDMA) has emerged as one of the best candidates for a high data-rate broadband transmission system in a number of standards, e.g. IEEE 802.11 a/g, IEEE 802.16 d - e, DVB-T, etc. [1], [2]. In every OFDMA system, each subcarrier is exclusively assigned to only one user. The key problem in the use of the OFDMA systems is the resource allocation algorithm. According to this algorithm the whole frequency spectrum is divided into subbands, and each subband is subdivided into slots in the time domain. The major problem is assigning these two-dimensional resources to different users under several constraints, such as the minimum data rate requirement, delay, etc.

The MIMO (Multiple Input Multiple Output) as another promising technology can be employed at both the transmitter and the receiver to introduce spatial diversity. With the help of additional antennas, MIMO receivers are more complex and with OFDMA they make the multi-path channel more efficiently. The MIMO-OFDMA has been used especially in the IEEE 802.16e standard and recommended in the IEEE 802.11n standard.

A number of low-complexity channel estimation schemes have been proposed for the OFDM systems. For instance, in the paper by Chang [3] the polynomial models were proposed in the design of the OFDM systems. Their design was also based on the maximum likelihood channel estimation and signal detection [4], [5]. For the sake of the performance analysis of these systems the least-squares [6] and the minimum mean square error (MMSE) were also proposed [7]. Nevertheless, the use of these methods for design of the OFDM system is possible, if identification at each spatial layer, i.e. a unique transmit-receive antenna path, is known. In other solution of this problem has been established the best channel estimation performance by loading pilot symbols on separable equispaced [8] and equipowered [9] subcarriers.

The main goal of this paper is to introduce a resource-allocation algorithm for a multiclass MIMO-OFDMA system where the spatial correlations could be different accross the subcarriers. The algorithm endeavors to maximize the system throughput while ensuring the full fillement of each user's QoS requirements including the data rate and the bit-error-rate (BER). In contrast to [10], [11], the subcarrier allocation, power distribution, and modulation modes are jointly optimized according to the users' channel state, locations, and QoS requirements. As regards the throughput maximization problem, we analyze the properties of optimal transmission schedules and propose an iterative algorithm exploiting these properties. The proposed algorithm has low complexity and can be implemented efficiently.

The rest of the paper is as follows. In section 2 we present the system model. In section 3 we show our optimal algorithm for the throughput maximization in the MIMO-OFDMA system. Section 4 provides some simulation results. Finally, section 6 concludes this paper.

## 2. System Model

In Fig. 1 we present the block diagram of a downlink MIMO-OFDMA system. The system is composed of a MIMO-OFDMA transceiver and a MIMO-OFDMA receiver. We assume that each of $K$ users have $M_r$ receiving antennas and the transmitters has $M_t$ transmitting antennas. Thus, for user $k$, $k = 1, 2, \ldots, K$, on subcarrier $n$, $n = 1, 2, \ldots, N$, the channel state matrix is given by $\mathbf{H}_{k,n}$ with

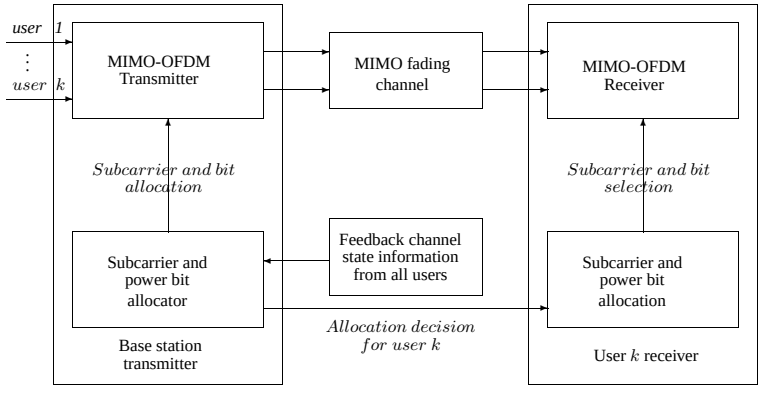



Fig. 1. Downlink MIMO-OFDMA system diagram

dimension $M_r \times M_t$.

The channel state matrix $\mathbf{H}_{k,n}$ can be decomposed as

$$\mathbf{H}_{k,n} = \sum_{i=1}^{M} u_{k,n}^{(i)} \sigma_{k,n}^{(i)} (v_{k,n}^{(i)})^H \tag{1}$$

where $M = min(M_r, M_t)$ is the rank of $\mathbf{H}_{k,n}$, $\{\sigma_{k,n}^{(i)}\}_{i=1}^{M}$ are the singular values of $\mathbf{H}_{k,n}$ in descending order, $\{u_k^{(i)}\}_{i=1}^{M}$ and $\{v_{k,r}^{(i)}\}_{i=1}^{M}$ are corresponding left and right singular vectors, respectively.

The activity of the MIMO-OFDMA system, depicted in Fig. 1, is as follows. The decision about the use of the allocation decision is obtained via the feedback signalling channel. Thus, each user can decode the data on the assigned subcarriers. With the help of the of the resource allocation scheme the parameters of the system will be updated as fast as the channel information is collected. To achieve the optimality of the MIMO-OFDMA system in accordance with the results obtained in the paper by Li [12] the following assumptions must be accepted:

1) The subcarriers cannot be shared by users.
2) Each subcarrier assigned to only one user can experience an independent downlink MIMO fading.
3) The MIMO channel is assumed to be constant during the subcarrier and the power allocation process.

3) At the receiver only the perfect channel state information (CSI) of all users is achievable.

Based on the above given assumptions, we assign for every subcarrier in a subband a coherence bandwidth possessing the same channel amplitude. Thus, the rate of user $k$ in an OFDMA system can be given as follows:

$$R_k^{OFDMA} = N_s \sum_{b=1}^{N_B} \rho(k,b) \sum_{p=1}^{N_p} log_2(1 + \gamma_{k,b}^{(p)} \frac{P_{k,b}^{(p)}}{N_s}) \tag{2}$$

where $N_B$ is a total number of subbands, each containing $N_s$ subcarriers, $\gamma_{k,b}^{(p)}$ and $P_{k,b}^{(p)}$ are the effective power gain to noise ratio and the power function of user $k$ on subband $b$ and spatial channel $p$, respectively;
$\rho(k,b) \in \{0,1\}$ is an indicator function with "1" denoting code $c$ being allocated to user $k$ and "0" otherwise.

## 3. The Resource Allocation for the MIMO-OFDM System with the Throughput Maximization

The throughput maximization problem in the MIMO-OFDMA system can be formulated as follows:

$$\text{maximize} \quad \sum_{k=1}^{K} R_k^{OFDMA}(P_k) \cdot t_k \tag{3}$$

$$\text{subject to:} \quad \sum_{k=1}^{K} \rho(k,b) = 1 \quad \forall b \in \{1,\ldots,N_B\} \tag{4}$$

$$\sum_{b=1}^{N_B} \sum_{k=1}^{K} \sum_{p=1}^{N_p} P_{k,b}^{(p)} = P_{max} \tag{5}$$

$$\sum_{b=1}^{N_b} \sum_{k=1}^{K} P_{k,b}^{(p)} \cdot t_k - E = 0 \tag{6}$$

$$\sum_{k=1}^{K} t_k - T = 0 \tag{7}$$

$$\frac{P_{k,b}^{(p)} (\sigma_{k,b}^{(p)})^2}{[\sigma_{k,b}^{(p)}]^2} = f_{BER}(b_{k,b}) \ \ \forall k,b \tag{8}$$

$$d_k \leq R_k^{OFDMA}(P_k) \cdot t_k \leq D_k,\ \forall k \in \{1, \ldots, K\} \tag{9}$$

where $f_{BER}(b_{k,b})$ is a threshold in order to satisfy the BER constraints, where $f_{BER}(b_{k,b})$ depends on the adopted modulation scheme; $P_{k,b}^{(p)}$ is the transmission power assigned to user $k$ in a subband $b$, $D_k$ is the total amont of data units. We assumed that $0 < d_k < D_k$ and $P_{max}$ is the maximum transmission power; $R_k^{OFDMA}(P_k) \cdot t_k$ is the data throughput of the user $k$ in the given $t_k$ time units and $R_k^{OFDMA}(P_k)$ is the date rate function for the user $k$; $T$ is the interval time when the channel state is static.

When the maximum throughput in the MIMO-OFDM system is considered, the optimal user assignment for subband $b$ and the corresponding power allocated can be solved as follows. The Langrangian is

$$\begin{aligned} L \;=\; & R_k^{OFDMA}(P_k) \cdot t_k + \mu_1(\rho(k,b) - 1) \\ & + \mu_2(P_{k,b}^{(p)} - P_{max}) + \mu_3(P_{k,b}^{(p)} \cdot t_k - E) \\ & + \mu_4(t_k - T) + \mu_5(\frac{P_{k,b}^{(p)}(\sigma_{k,b}^{(p)})^2}{[\sigma_{k,b}^{(p)})]^2} \\ & - f_{BER}(b_{k,b})) + \mu_6(R_k^{OFDMA}(P_k) \cdot t_k \\ & - D_k) + \mu_7(-R_k(P_k) \cdot t_k + d_k) \\ & + \mu_8(R_k(P_k) - D_k) + \mu_9 \cdot P_{k,m}^{(p)} \\ & + \mu_{10}(P_{k,b}^{(p)} - P_{max}) \end{aligned} \tag{10}$$

where $\mu_1 \div \mu_9$ are generalized Lagrange multipliers. Applying the method of the Lagrange multiplier and the Kuhn-Tucker conditions, we can obtain the necessary conditions for the optimal solution for all $k$, $1, 2, \ldots, K$.

### 3.1. The Optimal Solution of the Throughput Maximization in the MIMO-OFDM System

We can take into considerations the data subcarriers that are the data transmitted at power $P_{k,b}^{(p)}$, where $0 \leq P_{k,b}^{(b)} \leq P_{max}$. From the Lagrangian multiplier we can obtain the following equations:

$$R_k'(P_k) - \mu_7 R_k' P_k + \mu_8 R_k'(P_k) = 0 \tag{11}$$

$$R_k(P_k) + \mu_3 P_{k,b}^{(p)} + \mu_4 - \mu_7 R_k(P_k) + \mu_8 R_k(P_k) = 0 \tag{12}$$

We can consider the following two cases:

1) At least in one subcarrier is transmitted at power $P_{k,b}^{(p)}, 0 < P_{k,b}^{(p)} < P_{max}$. Thus, we can obtain

$$(1 - \mu_7 + \mu_8) R_k'(P_k) = 0 \tag{13}$$

As $R_k(P_k)$ is increasing, $R_k' \neq 0$. Therefore, $(1 - \mu_7 + \mu_8) = 0$. Thus, from dependence $R_k(P_k) \cdot t_k - D_k = 0$ we get $s_k = R_k(P_k) \cdot t_k = D_k$.

From Eq. (12) we obtain

$$(1 - \mu_7 + \mu_8) R_k(P_k) + \mu_3 P_{k,b}^{(p)} + \mu_4 = 0 \tag{14}$$

For all other subbands, that is $P_i = P_{max}$. From dependence $R_k(P_k) \cdot t_k - D_k = 0$, we obtain $s_i = R_i(P_i) \cdot t_i = D_i$.

2) Let in all the subcarriers are transmitted the data at power $P_{max}$. In this case $P_{max} \cdot T = E$. Hence, our problem is simplified to a time allocation problem. In order to obtain the optimal solution, we can use the following scheme. Every subcarrier is first allocated in such a way that $d_k$, $1 < k < K$ units of data can be transmitted. The remaining time will be allocated to the subcarriers by their transmission rates $R_k(P_{max})$ in the descending order. It means that the time will be allocated to the subcarrier with the highest transmission rate first until it has enough time to transmit all its $D_k$ units of data. Then, the subcarrier with the next highest transmission rate will proceed, and so on.

### 3.2. The Proposed Algorithm for the Throughput Maximization in the MIMO-OFDM System

Now we can give a new algorithm which can maximize throughput in the MIMO-OFDM system. This algorithm does not the special cases when $T \cdot P_{max} \leq E$ or $s_k = D_k$ for $\forall k, 1 \leq k \leq n$, studied in previous subsection.

In this algorithm, we suppose that at least one subcarrier will be transmitted at a power level lower than $P_{max}$ and not all subcarriers transmit $D_k$ units of data in

the optimal solution.

The first phase of our algorithm is as follows. We assume that all data streams are sorted in decreasing order of $R'_k(0)$ - first derivative of rate function for subcarrier $k$. Next, we use *required_energy* procedure to compute the required energy to transmission at given time constraint $T$. This function determines the transmission power of each subcarrier as follows. At first it allocates time to each subcarrier such that every subcarrier can transmit $d_k$, $1 \leq k \leq n$, units of data. The remaining time of transmission is allocated to the subcarriers in decreasing order of $R'_k(0)$. Therefore, a subcarrier will receive additional time only if all subcarriers before it have been allocated adequate time to transmit all their data.

In the second phase of our algorithm, an additional simple procedure is used to residual subcarrier allocation with the objective of enhancing the total system capacity.To compare the data rate of user $k$ is used the following dependence:

$$R_k = R_k + \frac{B}{N_s} \sum_{i=1}^{M} \log_2(1 + \gamma_{k,b}^{(p)} \frac{P_{k,b}^{(p)}}{N_s}) \tag{15}$$

where $N_s$ is the total number of subcarriers, etc.

The algorithm is described in Fig. 3. and its performance will be studied in the next section.

**procedure** *MIMO_OFDMA_subcarrier_allocation*;
**begin**
*consumed_energy* := 0;
**while** $\mid consumed\_energy - E \mid < \epsilon$ **do**
 **begin**
  *initialization_of_all_parameters_of_MIMO-OFDMA*;
  *sorting_of_data_streams_in_decreasing_order*;
  *required_energy*;
  *residual_subcarrier_allocation*;
 **end**;
**end**;

**Fig. 2.** Scheduling algorithm for MIMO-OFDMA system for throughput maximization.

In the first step of our algorithm, each user will be allocated its first subcarrier under the scheduling principle that the user with the least ratio of the achieved rate to its required proportion has the priority to choose one subcarrier at a time.

## 4. Simulation Results

In this section we present simulation results regarding the efficiency of the proposed algorithm presented in section 3. The goal of the simulation experiments is to compare the performance of the scheduling algorithm for MIMO-OFDMA system with the OFDM-FDMA system, as a special case of adaptive OFDMA scheme. We assumed that OFDM-FDMA scheme allocates fixed sequences of subcarriers to each user according to their proportionality constraints and the allocated subcarriers of each user cannot be changed over the time.

In our simlation, up to 12 users are to be allocated the resource. There are a total 512 subcarriers with 16 subbands, each having 32 subcarriers. Fig. 3 shows the total system capacity in dependence on the number of users for both the OFDM-FDMA method and the proposed algorithm. It can be observed that the proposed algorithm without a subcarrier rearrangement can achieve the maximum system capacity. We can observe that the system capacity under this algorithm increases with the increasing number of users. It is caused by the multi-user diversity. The OFDM-FDMA method does not have such a property.

Figure 4 depicts the throughput under the time and energy constraint in the simulated OFDM-FDMA system. We have observed that if the energy constraint was higher than the given 1000 mW units, all the data could be transmitted. If the energy constraint was lower than 250 units, no feasible solution could exist. The optimal data throughput was depicted by a concave function of the energy constraint. It is obvious that the computation overhead of the proposed algorithm is low.

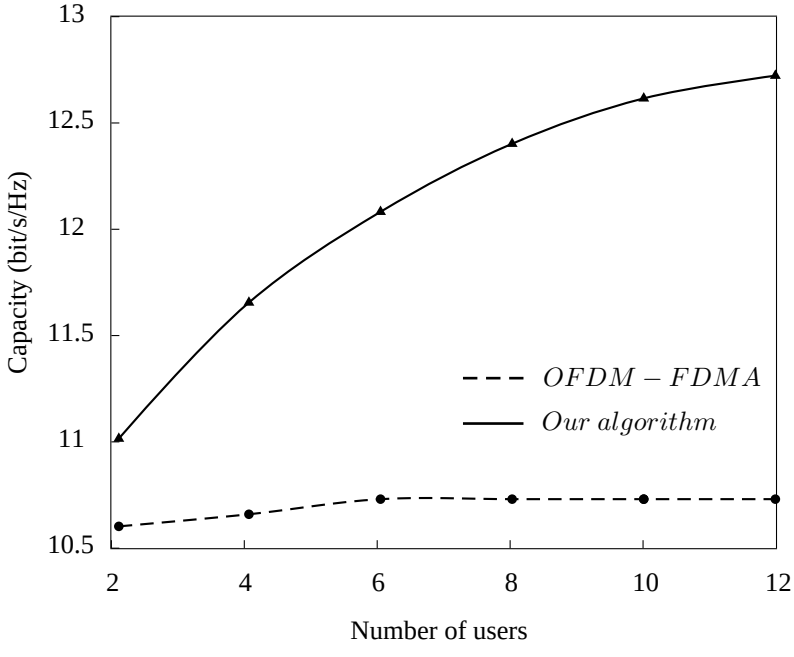



**Fig. 3.** Capacity of MIMO-OFDMA system in dependence of number of users.

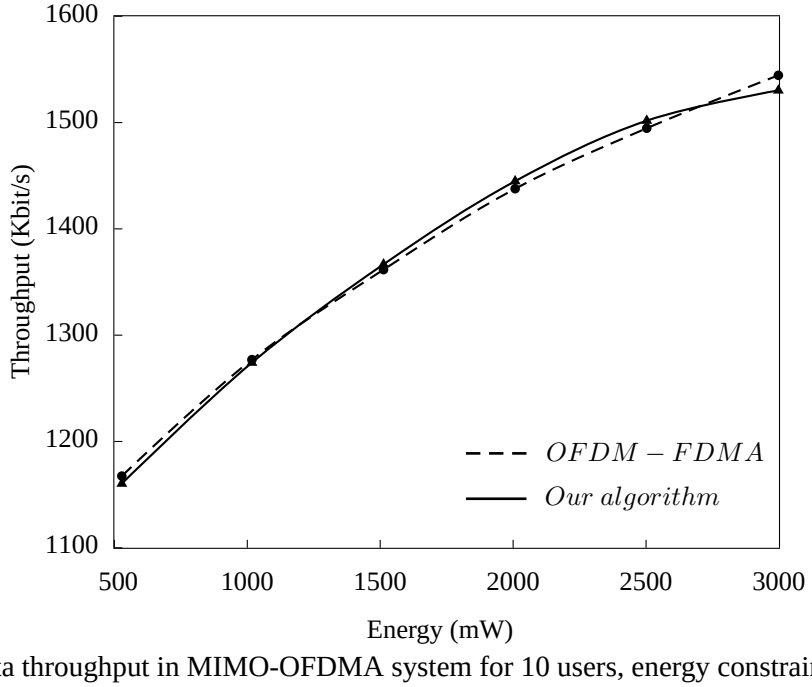



**Fig. 4.** Data throughput in MIMO-OFDMA system for 10 users, energy constraint equal to 1000 mW and time constraint equal to 100 ms.

## 5. Conclusion

A new efficient algorithm based on the heuristics for MIMO-OFDM systems was proposed. The objective is to maximize the throughput of data transmission while ensuring the fulfillment of every user's QoS requirements. The suggested algorithm allows us to make a good diversity of the multiuser data requirement, as well as the channel parameter changing and frequency guaranteeing. Additionally, the power efficiently decreases in comparison to the conventional FDMA system. It means that the proposed algorithm can be used in a dynamic change of the environment, as well as in the frequency and space domains.

## References

[1] Institute of Electrical and Electronics Engineers, Part 16: *Air Interface for Fixed Broadband Wireless Access Systems*, IEEE 802.16, June 2004.

[2] Institute of Electrical and Electronics Engineers, Part 16: *Air Interface for Fixed nd Mobile Broadband Wireless Broadband Wireless Access Systems, Amendment 2*, IEEE 802.16e, Dec. 2005 [URL http://www.ieee.org/web/standards/home/index.html].

[3] M.-X. Chang, Y.-T. Su, *Model-based Channel Estimation for OFDM Signals in Rayleigh Fading*, IEEE Trans. on Comm., Vol. 50, pp. 540 - 545, 2002.

[4] L. Deneire, P. Vandenameerle, L. van der Perre, B. Cyselinckx, M. Engels, *A Low Complexity ML Channel Estimator for OFDM Communications*, IEEE Trans. on Comm., Vol. 51, pp. 135 - 140, 2003.

[5] D. Chen, H. Kobayashi, *Maximum Likelihood Channel Estimation and Signal Detection for OFDM Systems*, Proc. IEEE International Conference Communication (ICC), pp. 1640 - 1645, 2002.

[6] E. Golovins, N. Ventura, Design and Performance *Analysis of Low-complexity Pilot-aided OFDM Channel Estimators*, Proc. 6th Int. Workshop on Multi-Carrier and Spread Spectrum (MC-SS), May 2007.

[7] E. Golovins, N. Ventura, *Reduced-complexity Recursive MMSE Channel Estimator for the Wireless OFDM Systems*, Proc. IEEE Wireless Communication and Networks Conference (WCNC), March 2008. Signals, Systems and Computers Conference, Vol. 1, pp. 324 - 328, 2008.

[8] R. Negi, J. Goffi, *Pilot Tone Selection for Channel Estimation in a Mobile OFDM System*, IEEE Trans. on Consumer Electron., Vol. 44, pp. 1122-1128, 1998.

[9] S. Ohno, G.B. Giannakis, *Optimal Training and Redundant Precoding for Block Transmissions with Application to Wireless OFDM*, IEEE Trans. on Commun., Vol. 50, pp. 2113 - 2123, 2002.

[10] I. Koutsopoulos, L. Tassiulas, *Adaptive Resource Allocation in SDMA-based Wireless Broadband Networks with OFDM Signaling*, IEEE Proc. INFOCOM, Vol. 3, pp. 1376 - 1385, 2002.

[11] I. Koutsopoulos, T. Ren, L. Tassiulas, *The Impact of Space-division Multiplexing on Resource Allocation: A Unified Approach*, Proc. IEEE INFOCOM, Vol. 1, pp. 533 - 543, 2003.

[12] G. Li, H. Liu, *On the Optimality of Downlink OFDMA-MIMO Systems*, Proc. Signals, Systems and Computers Conference, Vol. 1, pp. 324 - 328, 2004.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 367–378

# Moderate deviations of retransmission buffers over a wireless fading channel

KOEN DE TURCK [a] MARC MOENECLAEY[a] SABINE WITTEVRONGEL[a]

[a]Department of Telecommunications and Information Processing
Ghent University, Belgium
kdeturck@telin.ugent.be

**Abstract:** We study the buffer performance of retransmission protocols over a wireless link. The channel gain is modeled as a discrete-time complex Gaussian model. The advantage of this channel model over simpler alternatives (such as finite-state Markov models) is that the correspondence with the physical situation is more transparent. In order to keep the performance analysis of the buffer content tractable, we opt for heavy-traffic and moderate deviations scalings. We add some notes on the optimal selection of the signal to noise ratio.

**Keywords:** Moderate deviations, wireless channels, fading models.

## 1. Introduction

The popularity of wireless telecommunications is increasing rapidly. However, apart from the obvious advantages over wired networks, such as increased user mobility and easier deployment, wireless communications also have a number of drawbacks. For example, the dependence on batteries requires a more careful energy management. Secondly, the presence of fading and interference is also particular to wireless links and may cause a severe degradation of performance.

In this paper, we look at the performance loss due to fading. This loss manifests itself in a reduced throughput of the link, but other performance metrics may be severely affected as well, such as the mean packet delay, the overflow probability of the buffer at the transmitter's side or the delay jitter. This is by no means a new topic; we refer to [8, 7, 3] as a sample of how this problem has been tackled, some of the papers focus on throughput only, others on the complete buffer performance. The novelty of this paper resides in the fact that we combine a couple of

elements in a way that has not been done before, which leads to an elegant analysis. Firstly, we make use of a complex Gaussian process as a model of the channel gain. This type of model is more popular in communications theory circles, than for queueing analyses, because a direct computation of the buffer content distribution is too resource-intensive to be useful. For this reason, researchers interested in the buffer content have hitherto focused on finite-state Markov models. The undeniable advantage of Gaussian models however is that the correspondence with physical parameters is transparent: metrics such as the SNR (signal-to-noise-ratio) and coherence time feature directly.

During recent years, scaling has become a respected and full-fledged analysis tool. When confronted with a problem for which a direct computation is very costly or plainly impossible, the probabilist might opt to scale the original problem in such manner that a much simpler model arises, one in which the salient features of the original model are retained, but other stochastic fluctuations get filtered away. We look at two scaling methods in particular, namely heavy-traffic and moderate deviations. Heavy-traffic analysis is easily the oldest scaling method known in queueing theory. Kingman was the first to exploit the deep link between queueing systems operating under the limit load $\rho \to 1$ and diffusion processes. Moderate deviations do not have such a long history. Its promise is to combine the strong points of large deviations and heavy traffic methods. Essentially, it is a rigourous way of looking at tails of asymptotically Gaussian processes. We are indebted to the scaling approach taken in [11, 10].

The structure of the paper is as follows. In section 2., we detail the channel and the buffer model; in section 3., we review the moderate deviations and heavy traffic scalings and apply them to the model at hand. We look at some numerical examples in section 5. and finally, we draw conclusions in section 6.

## 2. Model

Consider a wireless station (the transmitter) delivering data packets to another wireless station (the receiver). Time is considered to be slotted, where the duration of a slot corresponds to the transmission time of a data packet of length $L$ bits. The transmission buffer has room for $B$ data packets. The channel over which the information is sent is subject to fading, which we model as follows. The channel gain $\mathsf{h}_t$ during slot $t \in \mathbb{N}$ forms a discrete-time complex Gaussian process. We assume wide-sense stationarity, and moreover $\mathrm{E}\,\mathsf{h}_t = 0$. The process is thus characterized completely by an autocorrelation function $r_t$:

$$r_t \doteq \mathrm{E}(\mathsf{h}_s^* \mathsf{h}_{t+s}) = \mathrm{E}(\mathsf{h}_0^* \mathsf{h}_t),$$

where $z^*$ denotes the complex conjugate of $z$.

The process $\{\mathsf{h}_t\}$ can also be characterized as filtered white noise. Indeed, consider a sequence $\mathsf{u}_t$ of independent and identically distributed (iid) complex normal variables with zero mean and unit variance, and a filter bank with parameters $g_t$ such that:

$$\mathsf{h}_t = \sum_s g_s \mathsf{u}_{t-s}.$$

The two representations are in fact equivalent. A popular choice in this case is the Butterworth filter. In this paper, however, we do not further elaborate on the filter representation of the channel process.

Two choices of $r_t$ are particularly popular. The so-called Jakes' model is perhaps the most well-known choice. It was derived from theoretical considerations, and expresses $r_t$ in terms of a Bessel function of the first kind:

$$r_t = J_0(2\pi f_d t),$$

where $f_d$ is the Doppler frequency. There is a simple relation between the the Doppler frequency $f_d$, the carrier frequency $f_c$ and the velocity of the receiver:

$$f_d = \frac{v}{c} f_c,$$

where $c$ denotes the speed of light. This shows the strong influence of the carrier frequency and the velocity on the nature of the fading process. The Doppler frequency also corresponds to the cut-off frequency of the filter.

The other popular form for the autocorrelation function is a Gaussian form:

$$r_t = \exp\left(-\frac{t^2}{2\alpha^2}\right), \tag{1}$$

where $\alpha$ is a form factor regulating the 'width' of the autocorrelation function. We can relate this to the Doppler frequency by determining the cut-off frequency in the frequency domain. A Gaussian function with form factor $\alpha$ in the time domain is mapped unto a Gaussian function with form factor $\alpha^{-1}$ in the frequency domain. Some manipulations yield the following formula for the $n$-dB cut-off frequency:

$$f_d = \sqrt{\frac{n}{5}\log 10}\alpha^{-1}. \tag{2}$$

Our overview of the channel model is completed by the link between the channel gain and the transmission error process. The bit error probability is a function of the channel gain $h$ as follows:

$$p_b(h) = Q\left(\sqrt{\frac{2E_b}{N_0}|h|^2}\right),$$

where $\frac{E_b}{N_0}$ denotes the SNR and $Q(x)$ denotes the error function; it is equal to the probability that a normal random variable with zero mean and unit variance is larger than $x$. The packet error probability $p(h)$ is the probability that at least one bit of the packet is incorrect:

$$p(h) = 1 - (1 - p_b(h))^L.$$

Let $\{\mathsf{c}_t\}$ denote the transmission process: $\mathsf{c}_t$ is equal to 1 when the transmission during slot $t$ is successful and 0 otherwise. We have that

$$\mathsf{c}_t = \mathsf{i}_{1-p(\mathsf{h}_t)},$$

where $\mathsf{i}_q$ denotes a Bernoulli random variable with success probability $q$.

Let $\{\mathsf{a}_t\}$ be the random process of the number of packet arrivals during slot $t$. A natural class of arrival processes for this kind of analysis is that they are asymptotically Gaussian under the scaling we are considering. We will provide more details as we go along. Stationarity is another natural condition that we impose throughout this paper. Let $\lambda \doteq \mathrm{E}\,\mathsf{a}_0; \mu \doteq \mathrm{E}\,\mathsf{c}_0$. The load of the system is defined as $\rho = \lambda/\mu$.

In this paper, we look at the transmitter buffer performance, with the so-called 'ideal ARQ' (ARQ stands for automatic repeat request) protocol: packets are retransmitted until they are received correctly, (until $\mathsf{c}_t = 1$) with the assumption that there is no feedback delay. That is, the transmission status of a packet is directly known. The scalings that we consider in this paper involve letting the load approach 1, and under such conditions ARQ with non-zero feedback delay converges to ideal ARQ. The queue content process $\{\mathsf{q}_t\}$ is formulated in terms of the arrival and transmission processes, by means of the well-known Lindley recursion:

$$\mathsf{q}_{t+1} = [\mathsf{q}_t + \mathsf{a}_t - \mathsf{c}_t]_0^B.$$

where $[x]_0^B \doteq \max(0, \min(B, x))$.

## 3. Scalings

We obtain asymptotic results on the queue content distribution by appropriately scaling the arrival and transmission streams. In this context, it is customary to define the net-input process $\mathsf{w}_t \doteq \mathsf{a}_t - \mathsf{c}_t$. Even within the class of scaling methods (which are by themselves already approximative) we have to be careful as to which methods offer good approximations for a reasonably low computational effort.

### 3.1. Fast-time scaling

We consider a set of scalings that involve speeding up the net-input process. Let $\mathsf{w}^{\otimes L}$ denote the net-input process sped up by a factor $L$:

$$\mathsf{w}_t^{\otimes L} \doteq \sum_{s=tL}^{(t+1)L-1} \mathsf{w}_s.$$

Let us look at a family of scalings of the form:

$$\hat{\mathsf{w}} = L^{(1-\beta)/2}(L^{-1}\mathsf{w}^{\otimes L} - (\lambda - \mu)\mathsf{1}). \tag{3}$$

where $\beta \in [0, 1]$ and $\mathsf{1}$ denotes a constant process and equal to 1. For $\beta = 0$, we get the so-called central limit scaling, whereas $\beta = 1$ is known as the large deviations scaling, (which is essentially the same as the scaling used for the law of large numbers).

Let us first have a look at the central limit scaling $\beta = 0$. Under mild conditions (typically the existence of the first two moments, and some mixing condition), the scaled process converges to a (discrete sample of) Brownian motion with zero drift and diffusion parameter $V^{\mathsf{w}}$:

$$V^{\mathsf{w}} = \lim_{t\to\infty} \mathrm{Var}[\sum_{s=1}^{t} \mathsf{w}_s].$$

The queue content process under this scaling converges to a reflected Brownian motion with drift $\lambda - \mu$, diffusion parameter $\sigma^2_{\mathsf{w}}$, and boundaries at 0 and $B$. This leads to a couple of simple performance formulae:

$$\mathrm{E}[\mathsf{q}] \approx \frac{V^{\mathsf{w}}}{2(\mu - \lambda)}, \tag{4}$$

and

$$\Pr[\mathsf{q} \geq b] \approx \exp\left(-\frac{2b(\mu - \lambda)}{V^{\mathsf{w}}}\right) \tag{5}$$

Large and moderate deviations scalings are less useful for this application. They involve computing a rate function, which for the transmission process at hand is either computationally complex (in the large deviations case), or leads only to the asymptotic formula (5) of the central limit result (in the moderate deviations case). Indeed, for the large deviations case, computations center around the scaled cumulant generating function (scgf) $\Lambda(\theta)$:

$$\Lambda(\theta) = \lim_{T\to\infty} \frac{1}{T} \log \mathrm{E}[\exp(\theta \mathsf{w}_0^{\otimes T})],$$

from which the rate function can be obtained by a Legendre-Fenchel transform. The computation of $\Lambda(\theta)$ requires the evaluation of a high-dimensional integral (with the dimension tending to infinity). This can only be solved by costly Monte-Carlo techniques, and as we set out to find easy to compute performance formulae, we will not pursue this path further. We note that the moderate deviations limit in fact corresponds to a second-order

### 3.2. Many flows scalings

We now look at a scaling that preserves the time-covariance structure of the original net-input process: instead of speeding up this process, we denote by $\mathsf{w}^{\oplus L}$ an aggregate of $L$ independent copies of the same net-input process. The family of scalings has now the following form:

$$\hat{\mathsf{w}} = L^{(1-\beta)/2}(L^{-1}\mathsf{w}^{\oplus L} - (\lambda - \mu)1), \tag{6}$$

again for $\beta \in [0, 1]$. In the central limit scale $\beta = 0$, the scaled process now converges to a Gaussian process (not necessarily Brownian motion) with the same drift and covariance structure as the original net-input process $\mathsf{w}$. Although the queue-content process also converges to a Gaussian process, it is generally difficult to derive closed-form performance metrics for it. This is why we resort to moderate deviations in this case. Under some mild conditions, the scaled process satisfies a moderate deviations principle (MDP) for $\beta \in (0, 1)$ with rate function $I_t$:

$$\lim_{L\to\infty} L^{-\beta} \log \Pr[\hat{\mathsf{w}} \in \hat{S}] \asymp - \inf_{t>0} \inf_{\hat{x}\in\hat{S}} I_t(\hat{x}), \tag{7}$$

where $I_t(x)$ is equal to

$$I_t(x) = \sup_{\theta\in\mathbb{R}^t} \theta^{\mathsf{T}} x - \frac{1}{2}\theta^{\mathsf{T}} C_t \theta, \tag{8}$$

with $C_t$ the covariance matrix of the net-input process (with dimension $t \times t$):

$$[C_t]_{ij} = \gamma_{|i-j|} = \mathrm{Cov}[\mathsf{w}_i, \mathsf{w}_j]. \tag{9}$$

The tail asymptotics of the queue content process are given by [11]:

$$\log \Pr[\mathsf{q} \geq b] \asymp -I, \tag{10}$$

where

$$I = \inf_{t\geq 0} \frac{(b + (\mu - \lambda)t)^2}{2V_t}. \tag{11}$$

We detail in the next subsection how to compute the variance function $V_t$, which is defined as follows: $V_t = \sum_{i,j}[C_t]_{ij}$. One can also use the 'refined asymptotics' of the Bahadur-Rao type [12]:

$$\Pr[\mathsf{q} \geq b] \approx \frac{1}{\theta^* \sqrt{2\pi V_{t^*}}} e^{-I}, \tag{12}$$

where $t^*$ is the $t$ that minimizes (11), and $\theta^* = (b + (\mu - \lambda)t)/2V_{t^*}$.

### 3.3. Computing the covariance structure

In this section, we show how to compute the function $V_t$ that appears in the asymptotic performance measures of the previous section. First, note that the net-input process is the sum of two independent processes: the arrival process and the transmission process, which means that $V_t$ can be split up likewise:

$$V_t = V_t^{\mathsf{a}} + V_t^{\mathsf{c}}. \tag{13}$$

For the arrival process, we opt in this paper for the parsimonious fractional Brownian process, which has three parameters: a drift $\lambda$, a diffusion parameter $\sigma^2$ and a Hurst parameter $H$, where $H \in (0, 1)$. For $H = \frac{1}{2}$, we have the standard Brownian motion with independent increments, In case of $H < \frac{1}{2}$ the increments are negatively correlated, and positively correlated for $H > \frac{1}{2}$. We have that $V_t^{\mathsf{a}} = \sigma^2 t^{2H}$.

The function $V_t^{\mathsf{c}}$ of the transmission process can be found via the auxiliary sequence $\gamma_t$:

$$\begin{aligned} \gamma_t &\doteq \mathrm{E}[(\mathsf{c}_0 - \mu)(\mathsf{c}_t - \mu)] \\ &= \mathrm{E}[(1 - p(\mathsf{h}_0) - \mu)(1 - p(\mathsf{h}_t) - \mu)]. \end{aligned} \tag{14}$$

where the last transition is due to the definition of $\mathsf{i}(.)$. The computation of this sequence is best done numerically. The computational complexity is relatively minor, however: for each $t$ we must evaluate a four-dimensional integral (recall that $\mathsf{h}_t$ are complex-valued random variables, thus yielding two dimensions each). Sequences $V_t^{\mathsf{c}}$ and $\gamma_t$ are related as follows:

$$V_t^{\mathsf{c}} = \sum_{i=0}^{t-1} \sum_{j=0}^{t-1} \gamma_{|i-j|}. \tag{15}$$

The asymptotic variance, which plays a central role in fast-time scalings, is equal to the limit $V = \lim_{t\to\infty} V_t/t$. Note that this limit may not exist, for example for fractional Brownian with Hurst parameter $H > \frac{1}{2}$.

## 4. Optimal control

The transmission over wireless channels poses a challenging control problem to the designer of wireless networks: which level of the SNR represents the optimal trade-off between quality-of-service and energy consumption ? The parsimonious performance formulae presented in the previous section offer a feasible path to the static optimization. Indeed, assume given as a QoS constraint that the overflow probability must be smaller than $P$. The SNR influences the transmission rate $\mu$ and the variability of the transmission process $V_t^{\mathsf{c}}$. The buffer size $b$ influences the overflow probability. The control problem is thus as follows:

Find the minimal buffer size $b$ and SNR such that:

$$-\log P > \inf_{t \geq 0} \frac{(b + (\mu(\text{SNR}) - \lambda)t)^2}{2(V_t^{\mathsf{a}} + V_t^{\mathsf{c}}(\text{SNR}))}.$$

One can also adapt the SNR dynamically according to the perceived current channel and traffic state. This dynamic optimization problem is a lot harder, and is the subject of future research.

## 5. Some numerical results

Consider a scenario in which a transmitter sends data packets of $L = 10000$ bits to a receiver over a wireless channel subject to fading. The duration of a packet transmission is 5 ms. The carrier frequency $f_c$ of the transmission is $1Gz$, and the receiver moves relative to the sender with a velocity of $v$. In the first pair of figures, we show autocorrelation function of the channel gain for different velocities and for the Gaussian and Jakes' model respectively. Note that the manner in which the two models decay is completely different. We also show the corresponding autocorrelation function $\gamma_t$ of the transmission process $\mathsf{c}_t$ for the same scenarios in figure 2. Note that again, for a Jakes' model there are a lot of small bumps after the main bump, whereas for the Gaussian model there is only one bump. Although the bumps appear small they have an considerable influence on the function $V_t$.

Next, we turn our attention to the buffer performance proper. We plot the tail probabilities of the buffer occupancy for different velocities in the left subplot of figure 3. We see that the velocity has a huge effect on the buffer performance. In the right subplot, we look at the log-probability of the buffer exceeding a certain level ($b = 80$) versus the velocity $v$. We see that above some speed the influence gets minimal.

In the last figure, we show that when the arrival source is really bursty (Hurst parameter $H = 0.7$, signifiying a large positive correlation), the performance of the

buffer deteriorates to the extent that the influence of the fading channel is hardly seen.

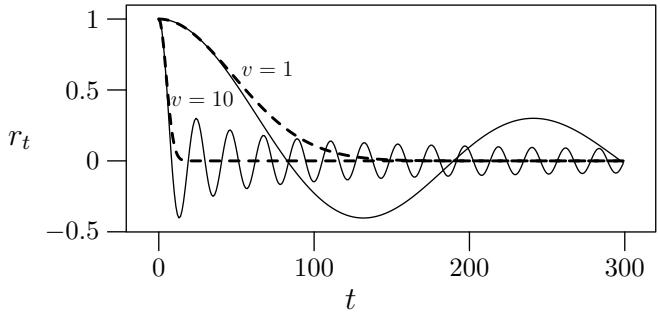



Fig. 1: Channel autocorrelation functions $r_t$ with 'Gaussian' form (dashed lines) and Besselian form (Jakes' model; full lines) for different values of $v$(in km/h).

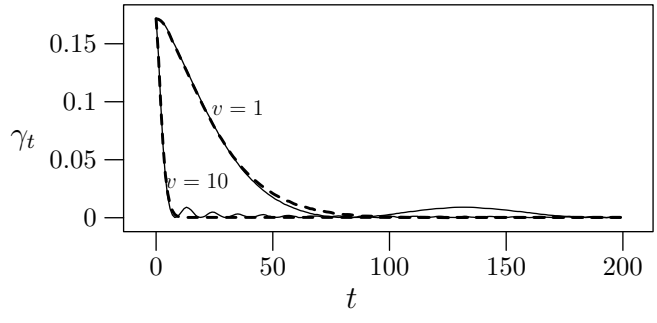



Fig. 2: Transmission autocorrelation function $\gamma_t$ for different values of $v$. Gaussian-form models are in dashed lines; Jakes' models are in full lines. Values of the other parameters are: $f_c = 1$ GHz; $T_p = 5$ms;$L = 10000$; SNR=14dB.

## 6. Conclusion

We studied the moderate deviations asymptotics of a retransmission buffer over a wireless fading channel. We found easy to evaluate performance formulae that link important physical parameters such as signal-to-noise ratios, coherence time and so on. The most important conclusions are: (1) the throughput alone does not suffice to characterize the buffer performance, (2) the lower the velocity of the receiver the worse the buffer performs (3) the effects of the fading channel might be swamped by really bursty (or 'Hursty') arrival sources, especially when the packet error probability is reasonably low, and (4) Gaussian and Jakes' fading models give different tail behavior.

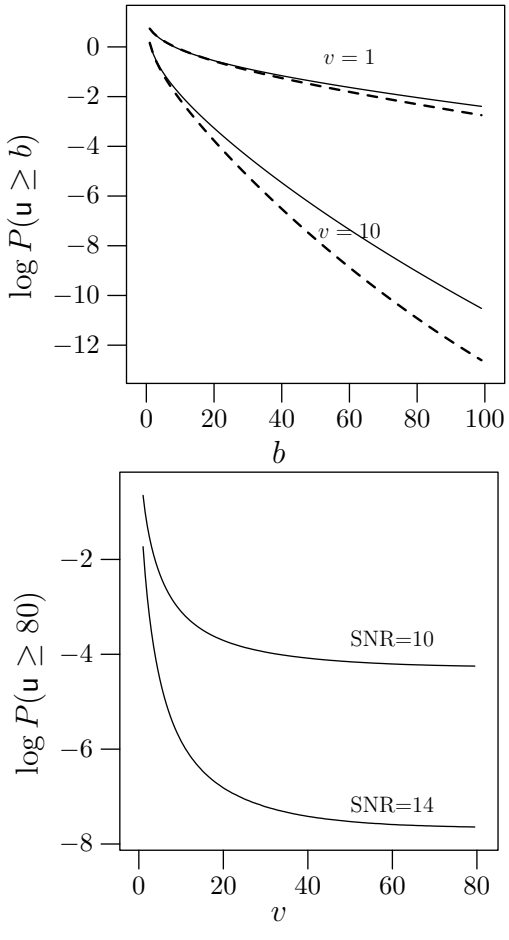



Fig. 3: Left: Log probabilities of the buffer content $\mathsf{u}$ for different values of $v$; Gaussian-form models are in dashed lines; Jakes' models are in full lines. Right: $\log P(\mathsf{u} \geq 80)$ versus the velocity $v$ of the receiver, for a Gaussian-form model. Values of other parameters are: SNR=14dB;$H = 0.5$;$V^{\mathsf{a}} = 0.1$.

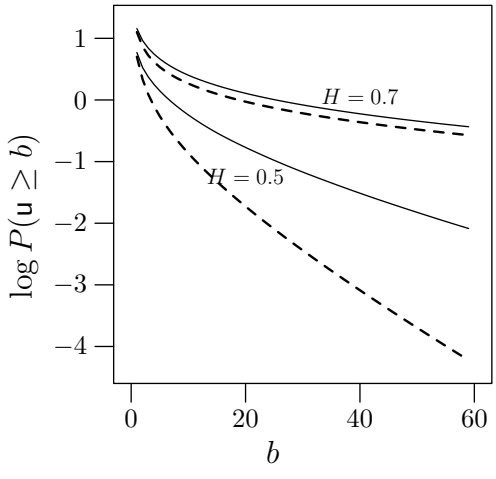



Fig. 4: A plot of $\log P(\mathsf{u} \geq b)$ versus $b$ for an SNR of 10dB (dashed lines) and of 14dB (full lines) for two values of the Hurst parameter. Values of the other parameters are: $V_{\mathsf{a}} = 1$; $v = 5$km/h.

## References


[1] Bhunia, C.T.: ARQ - Review and modifications. IETE Technical Review **18** (2001) 381–401

[2] Towsley, D., Wolf, J.K.: On the statistical analysis of queue lengths and waiting times for statistical multiplexers with ARQ retransmission schemes. IEEE Transactions on Communications **25** (1979) 693–703

[3] Konheim, A.G.: A queueing analysis of two ARQ protocols. IEEE Transactions on Communications **28** (1980) 1004–1014

[4] De Munnynck, M., Wittevrongel, S., Lootens, A., Bruneel, H.: Queueing analysis of some continuous ARQ strategies with repeated transmissions. Electronics Letters **38** (2002) 1295 – 1297

[5] Towsley, D.: A statistical analysis of ARQ protocols operating in a non-independent error environment. IEEE Transactions on Communications **27** (1981) 971–981

[6] De Vuyst, S., Wittevrongel, S., Bruneel, H.: Delay analysis of the Stop-and-Wait ARQ scheme under correlated errors. Proceedings of HET-NETs 2004, Performance Modelling and Evaluation of Heterogenous Networks (26-28 July 2004, Ilkley, West Yorkshire, UK), p. 21/1–21/11

[7] R. Fantacci. Queueing analysis of the Selective Repeat automatic repeat request protocol wireless packet networks. *IEEE Trans. Veh. Technol*, 45:258–264, 1996.

[8] Kim, J.G., Krunz, M.: Delay analysis of selective repeat ARQ for transporting Markovian sources over a wireless channel. IEEE Transactions on Vehicular Technology **49** (2000) 1968–1981

[9] Gilbert, E.N.: Capacity of a burst noise channel. The Bell System Technical Journal **39** (1960) 1253–1265

[10] Ayalvadi Ganesh, Neil O'Connell, Damon Wischik: Big Queues. Springer Verlag Berlin (2004)

[11] Damon Wischik: Moderate deviations in queueing theory (submitted for publication)

[12] Montgomery, M., DeVeciana, G.: On the relevance of time scales in performance oriented traffic characterizations. Proceedings of INFOCOM 1996.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 379–390

# TCP Congestion Control Algorithms Performance in 3G networks with moving client

MACIEJ ROSTAŃSKI [a] PIOTR PIKIEWICZ[a]

[a]Academy of Business
in Dabrowa Gornicza
{mrostanski|ppikiewicz}@wsb.edu.pl

**Abstract:** Presented article focuses on improving performance of the TCP/IP connection in specific condition - connection between the data server and client downloading data, using mobile (cellular) network as an Internet connection method, while driving. A way to affect mechanisms of transport layer, and achieve better performance, is method described as changing TCP's Congestion Control Algorithm (CCA), which is responsible for congestion window behaviour. Today's TCP flavours are presented. For experimental research, topology is created, and test scenarios are being discussed. Methodology and tools are presented, as well as comparison of different CCA performance in realized test runs. Presented research leads to conclusion there is a field of study on *cwnd* behaviour in 3G network while using a family of congestion control algorithms designed for fast networks, since those get better results than CCAs designed for wireless networks. The results and conclusions of this article address the infrastructure level of a typical, modern, european, urban region.

**Keywords:** : Congestion Control, TCP, CuBIC, Veno, Reno, Throughput, UMTS, 3G

## 1. Introduction

Presented article focuses on improving performance of the TCP/IP connection in specific condition - connection between the data server and client downloading data, using mobile (cellular) network as an Internet connection method. Performance and similar parameters of a connection using GSM / WCDMA technologies as an Internet access method is a subject widely researched. The variety of conditions and factors affecting performance of such connection is often a topic of scientific reports - for example [1], [2] or [3].

Problem, addressed in this article, is (in spite of fact that conclusions of similar research are useful for mobile network operators nad vendors), they present little

value to the end-user - who doesn't have the capability to alter any parameters of 3G infrastructure he uses.

There is, however, a way to affect mechanisms of transport layer, and achieve better performance. Method described in this article focuses on changing TCP's Congestion Control Algorithm (CCA), which is responsible for congestion window behaviour. Server administrator can easily recompile and merge any CCA (even his own) into Linux kernel and change CCAs using Linux kernel parameters.

## 2. TCP State of art

During data transfer, every packet is exposed to phenomenas slowing down or interrupting its travel through computer network. This slowing down, in the case of TCP segments in particular, may be caused by transmission channel congestion or interferences observable in wireless networks. In TCP protocol, mechanisms regulating sending data rate in dependence of network state, were introduced as a solution to network congestion problem.

First such modification was so called Tahoe [4] algorithm, including *slow-start, congestion-avoidance* and *fast-retransmission* mechanisms. In TCP Reno [5] algorithm, besides Tahoe modifications, an algorithm of Fast-Recovery of lost packets was introduced [6], [7]. Fast-Recovery uses fast retransmission mechanism, in which multiple acknowledges of the same packet indicate packet loss. So called New Reno algorithm [8] is capable of Selective Acknowledgments (SACK), allowing TCP protocol to continue fast retransmission procedure without Slow-Start, even after getting only partial acknowledgments (allowing for some ACK to arrive later).

Today there are many known versions of TCP algorithms, altering TCP window size in different manner, that are in majority modifications of Reno mechanism, but focusing on different problems of modern networks - some address specific wireless/satellite networks issues, other, for example, deal with TCP problems on Long Fat Networks. Algorithms BIC TCP (Binary Increase Congestion Control)[9] and CuBic [10], designed for use with networks with large BDP (Bandwidth-Delay Product), distinguish themselves with window growth function, specific around link saturation point. Mentioned CCAs include methods of new TCP window value designation, which cause no exaggerative growth, which in turn leads to maintaining optimal data transfer rate longer comparing to other algorithms. CuBIC Algorithm is used by default in modern Linux kernel.

Another approach is presented by Vegas algorithm [2] - Vegas tries to accomplish better bandwidth saturation, avoiding retransmissions, with appropriate flow reduction. Algorithm predicts (or tries to predict) congestion before it happens and

reduces packet sending rate, hoping to reduce packet delay, and, in effect, increase performance. This is known as proactive behaviour. Around 2003, Veno algorithm was proposed [11] - an attempt to merge the advantages of Reno and Vegas algorithms. Veno algorithm relies on Vegas proactive mechanism only to identify, whether specific packet loss incident is caused by network congestion or is it effect of random wireless environment issues. Congestion window regulation is made similar to Reno behaviour.

Using such algorithms as Veno or Vegas should bring good results in wireless networks, where packet loss probability is much higher compared to wired networks. In addition to those, specifically for application in wireless networks, where potential packet loss is a result of transmission error rather than network congestion, and network load is highly dynamic nature, Westwood algorithm [12] was also designed. In Westwood algorithm, the stream of acknowledgments is analyzed for estimation of Eligible Rate value, used in turn for congestion window optimization and accurate setting up *ssthresh* value in case of packet loss event discovery. Important difference between Westwood and Reno is a method of congestion window reduction. Westwood, contrary to Reno (which halves the *cwnd* - value of congestion window, after congestion event), uses information of last available bandwidth with window reduction calculation.

Another CCA worth mentioning, YeAH algorithm[13], is quite fair with competing Reno flows, and doesn't cause significant performance loss in case of random packet loss event. YeAH assumes cooperation of two phases - "quick" phase, when cwnd is increased following procedures like in STCP [14], and "slow", during which algorithm YeAH behaves like Reno algorithm. The state of algorithm phases is conditioned on predicted packet number in log queue.

## 3. 3G technologies

As forementioned, the performance of the connection to the Internet via cellular/mobile networks is a subject of vast scientific research, specifically for technologies in scope of this paper, articles like [15], [16], [17] . Some reason for this, and an important fact is that while using mobile terminal for data connection with the Internet, different cellular data transfer technologies may be used. This is also dependable of the distance between base stations and client, base stations and/or client capabilities, radio conditions etc. [18],[19]. Therefore, typical 3G cellular network offers its clients a gradable performance, deploying specific technologies for infrastructure of crowded cities rather than suburban region, or rural areas, downgrading when necessary to 2.5G or even 2G technologies.

In table 1 we placed main cellular access technologies with their key perfor-

| Technologies (generation) | Available bandwidth / throughput at end-user | Observable packet loss (end-to-end Internet conn.) | Latency (RTT values) | Jitter |
|---|---|---|---|---|
| GPRS (2.5G) [22], [25] | Very Low (115kbps / 40kbps) | up to 10%, due to timeouts and delay spikes | Bad (up to 2000ms) | High |
| EDGE (2.5G) [26],[27] | Mediocre (474kbps / 100-130kbps) | up to 10%, due to timeouts and delay spikes | Bad (up to 1500ms) | Medium |
| UMTS (3G) [27] | High (2Mbps / 300kbps) | low (sometimes timeouts due to poor radio conditions) | Good | Low |
| UMTS - HSDPA (3.5G) [28], [29] | Relatively very high (1,5Mbps-4Mbps / 900kbps ) | very low, omittable | Good (20-100ms) | Low |

Table 1. Cellular data connection comparison

mance metrics, judging from user and transport layer perspective. As described in [20], [21] we concentrate on RTT, throughput, packet loss and jitter - those are the key factors of a TCP connection performance [22]. In research presented herein, mostly UMTS technology was available during test trials.

One must note that the classification herein (Table 1) is not in any way comprehensive - we concentrate on the packet-switched technologies, and GSM-originated (deployed mostly european-wide). More information was presented in our recent papers, e.g. [22].

In addition, one must have in mind that the scope of this paper is limited - discussed technologies are continuously advancing, especially 3.5G for example such as HSPA+, MIMO HSDPA [23] and there are first commercial realizations of 4G family, based on LTE technology. Good presentation of all 3GPP standards may be found on [24].

## 4. Experimental setup

### 4.1. Methodology and tools

As the thesis of the article claims, effective throughput depends among others of applied congestion control algorithm at the sender side and may be significant in wireless 3G network. So, the test topology should realize following scenario: (1) The server (sender side) should be capable of using many pluggable CCAs, (2) Mobile client should be able to download data using 3G connection with the server, (3) The case is, there should be disturbances caused by moving client.

Figure 1 shows created topology.

As a traffic generator, *nuttcp* application was used - known and widely deployed in Solaris and Linux systems measurement application [31]. Nuttcp is capable of creating data transfer in any direction (client-to-server or server-to-client) which makes it very useful in researched network, as there is no possibility of creating connection with open ports (server) at the mobile operator (client) side.

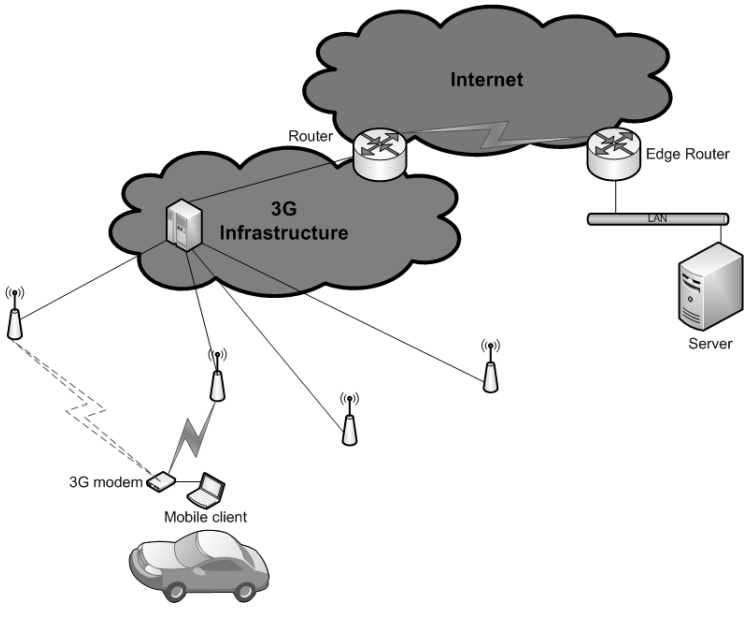



Fig. 1. Test research topology

Transfer tests were measured at the mobile station side, recorded using Windows XP and Wireshark, open source network analyzer, that permits analysis of network traffic as well.

Congestion window has to be observed on the sender-side (server). For this, in Linux system, *TCP Probe* module was used. Module is capable of saving TCP connection state, *cwnd* and *ssthresh* values, and packet sequence numbers in observed connection.

### 4.2. Topology setup

The testbed topology consists of three key elements (see Fig. 1): wireless (celullar) client with 3G capable modem, the Internet server, and a car. Wireless client, connected to the 3G operator is put in the car, allowing testing while driving. Regardless of the set up topology, some moving client conditions had to be put up to express typical moving client behavior. We propose a simple example of recreating good, average and bad conditions, as follows.

#### 4.2.1. Good conditions case

In order to create good conditions scenario, cellular client was placed in a slow cruising car. To ensure there are no roaming events during tests, cruising path

was carefully traced in vicinity of an operator access point. Speed did not exceed 30kmph, and there were frequent stops.

#### 4.2.2. Average conditions case

Creating an average conditions scenario we assumed there should be roaming events involved; speed of movement should vary between 20-60kmph and there will be stops, e.g. on the red lights. In other words, case would be about transferring data in average city traffic. The path of test was therefore placed between cities of Dabrowa Gornicza and Sosnowiec, mainly an urban area.

#### 4.2.3. Bad conditions case

Naturally bad conditions case is about data transfer during fast travel - test involves a highway run (80-100 kmph). In this research, the case was tested between Dabrowa Gornicza and Katowice cities in Silesia metropolis region.

Naturally, one of the main issues is the effect of 3G infrastructure level in tested areas on our research (this includes an effects of roaming, attenuation, disturbances and unknown quality of service rules at the operator network). This research does not address the problem of base stations localization and configuration, though. Suffice to say is that one has to have in mind that results and conclusions of this article address the infrastructure level of a typical, modern, european, urban region.

## 5. Results and conclusions

Tests were conducted with few distinct congestion control algorithms (Reno, CuBIC, Veno, Westwood) and analyzed for (1) congestion window behaviour, (2)RTT values, (3)achieved througput with given CCA. Results are given as follows.

### 5.1. Congestion window behaviour

As expected, CCAs try to achieve maximum allowable throughput very quickly. Such action is shown on Fig. 2 for three algorithms. CuBIC's specific growth function near saturation point effect can be seen. Worth noting is, algorithms predestined for wireless networks (such as Westwood or Veno) achieve sturation point longer than aggresive CCAs, like CuBIC, when beginning with slow-start phase.

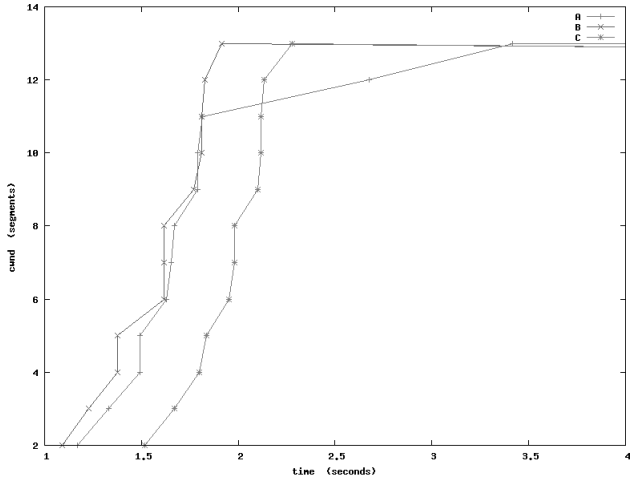



Fig. 2. Example of *cwnd* adjustment for different CCAs: A) CuBIC, B)Westwood, C)Reno

### 5.2. RTT values

Round-Trip Time values are consistent for every CCA; average RTT is low - around $20ms$. A small, but observable jitter exists. This is very important for any TCP CCA since *cwnd* may be altered by algorithm once every RTT occurance. Also, timeout values are derived from RTT.

### 5.3. Throughput comparison

For clearance, throughput achieved using CCAs diagrams were split onto three exemplary comparisons.

Fig.3 shows throughput achieved in similar operating conditions and time by using Westwood, and later, CuBIC algorithm. Even disregarding radio problems around 55th second for Westwood, it is clear that CuBIC achieves better throughput in any circumstance.

As shown on Fig. 4, comparing Veno to Cubic shows interesting difference. On this figure, first 30 seconds is bad radio conditions period, and other half is good conditions period. Under good conditions scenario, both algorithms perform similar and achieve similar results. The difference lies in bad conditions handling. Veno, using proactive mechanisms, predicts problems and stays within lower *cwnd* values. This causes uninterrupted transfer (with little jitter), but is not as effective as aggresive behaviour of CuBIC.

When comparing CuBIC to Reno results, it becomes obvious that in this scenario, CuBIC modifications perform very well. CuBIC achieves better performance

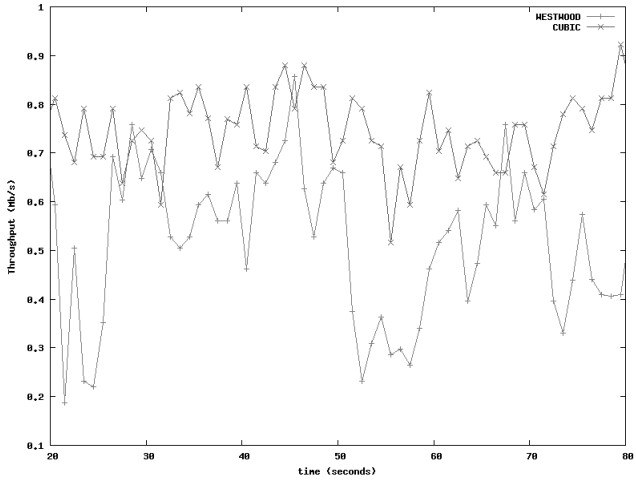



Fig. 3. Westwood and Cubic throughput comparison

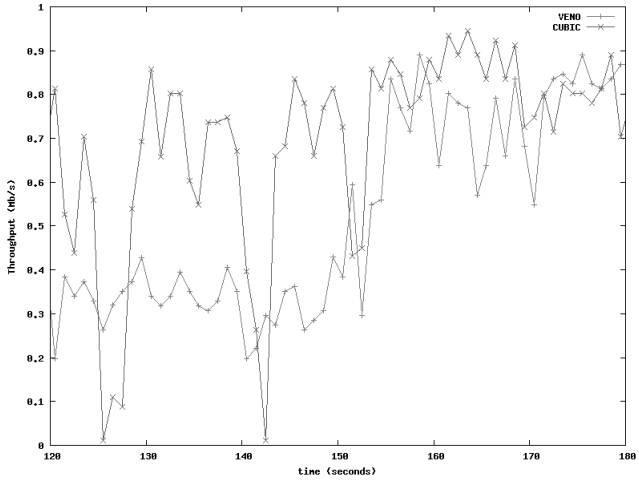



Fig. 4. Veno and Cubic throughput comparison

with this type of connection. Reno doesn't handle RTT jitter very well.

Main problems of experimental research are:

a) tests involving moving (driving) are hard to reproduce in identical manner - the only solution seems to be performing statistically big number of test runs, which leads to another problem:

b) 3-minute test requires around 10MB of data transfer, which leads to notifiable cost of research,

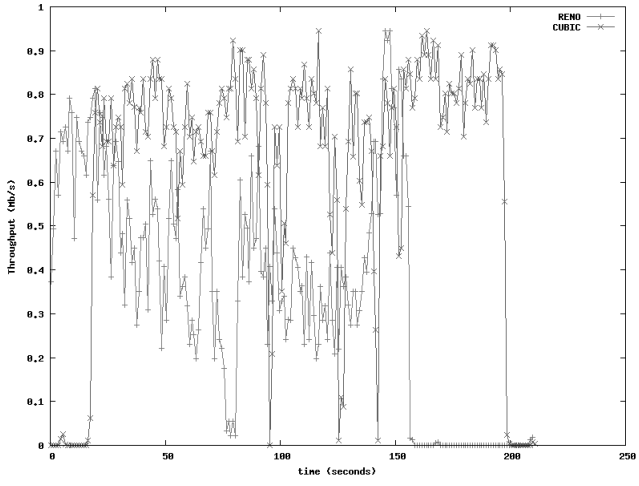



Fig. 5. Reno and Cubic throughput comparison

c) there is a possibility of many concurrent users in researched area, which will lead to lower maximum throughput allowed by an operator. Tests then have to be repeated to be comparable to other results.

Nevertheless, conducted tests lead to interesting conclusions.

Unlike technologies of lower mobile generations (2, 2.5G), scenario of Internet access using 3G connection delivers IP parameters similar to wired connection - low RTT, high throughput, almost no packet loss probability. Therefore, we observe better performance using congestion control algorithms designed with Long Fat Networks in mind - especially CuBIC. Wireless - friendly CCAs, as Westwood or Veno, do not improve throughput at all in this circumstances.

Presented research leads to conclusion there is a field of study on *cwnd* behaviour in 3G network while using a family of congestion control algorithms designed for fast networks - such as HS-TCP, Hamilton, YeAH. Also, should trials be consistent with observation, detailed study may be produced.

## References

[1] Inamura H., et al.: *TCP over Second (2.5G) and Third (3G) Generation Wireless Networks*, IETF RFC 3481, February 2003

[2] Brakmo L., O'Malley S., Peterson L., *TCP Vegas: New techniques for congestion detection and avoidance*, Proceedings of SIGCOMM '94 Symposium, 1994

[3] Ghaderi M., Boutaba R.: *Mobility Impact on Data Service Performance in GPRS Systems*, TR-CS University of Waterloo, Canada, 2004

[4] M. Allman, V. Paxson, W. Stevens: *TCP Congestion Control*, Network Working Group, IETF RFC 2581, April 1999

[5] V. Jacobson: Berkley TCP evolution from 4.3-Tahoe to 4.3 Reno, Proceedings of the 18 th Internet Engineering Task Force, University of British Colombia, Vancouver, Sept. 1990

[6] Stevens W.: *RFC 2001 - TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms*, http://www.ietf.org/rfc/rfc2001.txt, webpage version of June 2008

[7] S. Floyd, T. Henderson: RFC 2582: *The New Reno Modification to TCP's Fast Recovery Algorithm*, 1999

[8] Floyd S., Mahdavi J., Mathis M., Podolsky M.: *RFC 2883: An Extension to the Selective Acknowledgement (SACK) Option for TCP*

[9] Lisong X., Harfoush K., Rhee I.: *Binary Increase Congestion Control for Fast, Long-Distance Networks* in proc. of IEEE Infocom 2004

[10] Rhee I., Xu L.: *CUBIC: A New TCP-Friendly High-Speed TCP Variant*, in proc. of PFLDnet 2005, February 2005, Lyon, France

[11] Fu C., Liew S., *TCP Veno: TCP Enhancement for Transmission Over Wireless Access Networks*, IEEE Journal on Selected Areas in Communications, Vol. 21, No. 2, February 2003

[12] Chen J., Paganini F., Wang R., Sanadidi M. Y., Gerla M.: *Fluidflow analysis of TCP Westwood with RED*, Computer Networks: The International Journal of Computer and Telecomunication Networking, Vol. 50, Iss. 9, June 2006

[13] Baiocchi A., Castellani A., Francesco Vacirca F.: *YeAH-TCP: Yet Another Highspeed TCP*, http://wil.cs.caltech.edu/pfldnet2007/paper/YeAH_TCP.pdf, webpage version of June 2008

[14] Tom Kelly, *Scalable TCP: Improving Performance in Highspeed Wide Area Networks*. Computer Communication Review 32(2), April 2003

[15] Chakravorty R., Cartwright J., Pratt I.: *Practical Experience with TCP over GPRS*, in proc. IEEE GLOBECOM Conference, 2002

[16] Bhandarkar S., Reddy A.L.N., Allman M., Blanton E.: *Improving the Robustness of TCP to Non-Congestion Events*, Network Working Group Request for Comments: RFC 4653, 2006

[17] Benko P., Malicsko G., Veres A.: *A Large-scale, Passive Analysis of End-to-End TCP Performance over GPRS*, Ericsson Research, in proc. INFOCOM Conference, 2004

[18] Halonen T., Romero J., Melero J.: *GSM, GPRS, and EDGE performance. Evolution Towards 3G/UMTS*, John Wiley & Sons, England 2003

[19] Park K.: *QoS in Packet Networks*, wyd. Springer, USA 2005

[20] Blum R.: *Network Performance: Open Source Toolkit*, Wiley and Sons, USA 2003

[21] Hassan M., Jain R.: *High Performance TCP/IP Networking*, Wiley 2004

[22] M. Rostanski: *Poprawa wydajnosci przesylu w bezprzewodowej, radiowej, pakietowej transmisji danych*, PhD thesis, Institute of Theoretical and Applied Computer Science, Gliwice, Poland 2009

[23] *MIMO HSDPA Throughput Measurement Results in an Urban Scenario*, In proceedings of VTC2009 conference, Anchorage USA, Sept. 2009

[24] 3G Americas Standard Page, http://www.3gamericas.org/index.cfm?fuseaction=page&sectionid=345

[25] Heine G., Sagkob H.: *GPRS: Gateway to Third Generation Mobile Networks*, Artech House, USA 2003

[26] Rostanski M.: *Symulacja przesylu danych GPRS*, in: Internet w spoleczenstwie informacyjnym, w WSB Dabrowa Gornicza, Poland 2005

[27] The 3rd Generation Partnership Project (3GPP) Webpage: http://www.3gpp.org/

[28] J. Derksen, R. Jansen, M. Maijala and E. Westerberg: *HSDPA performance and evolution*, Ericsson 2006

[29] Spirent Communications White Paper: *HSDPA Throughput: Do Today's Devices Really Perform?* http://spcprev.spirentcom.com/documents/4683.pdf, January 2007

[30] Hiroyuki Ishii: *Experiments on HSDPA Throughput Performance in W-CDMA Systems*, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences archive, Volume E89-A , Issue 7 (July 2006), pp. 1903-1912

[31] Nuttcp man page: http://www.die.net/doc/linux/man/man8/nuttcp.8.html

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 391–404

# Analysis of transmission errors in 869.4-869.65 MHz frequency band

AGNIESZKA BRACHMAN ZBIGNIEW LASKARZEWSKI
LUKASZ CHROST

Institute of Computer Science
Silesian Technical University
*{agnieszka.brachman, zbigniew.laskarzewski, lukasz.chrost}@polsl.pl*

**Abstract:** Correctness and reliability are very important in wireless transmission. Communication errors cause the need of the data retransmission, which increases the bandwidth usage. Furthermore, transmitting the additional packets consumes additional energy of devices, which implicates shortening of their life-time, if they are battery powered. Understanding the characteristics of the transmission errors is significant for proper choice, design and calibration of correcting mechanisms intended for improving communication. The knowledge of error patterns or knowledge of lack of such patterns will allow more deliberate choice of encoding schemes for packet transmission.

We carried out propagation measurements to determine realistic transmission loss and transmission error characteristics in the 869 MHz band. This paper presents summarized results and error analysis. The transmission traces were collected during 10 days in diversified, urban environment. In the process of characterizing these errors several analysis concerning Received Signal Strength Indicator (RSSI), bit error rates, bit lengths and bit patterns were performed.

**Keywords:** wireless sensor networks, 869 MHz, wireless transmission, bit error rate, error analysis

## 1. Introduction

The usage of radio bands and wireless devices is heavily regulated throughout the world. In most countries existing regulations allow using the license-free areas of the spectrum. The license-free bands are intended for industrial, scientific and medical purposes [1]. Probably the most common unlicensed radio band is the 2.4 GHz used for home microwave ovens, wireless LANs, Bluetooth, ZigBee and cordless phones. Other popular unlicensed radio bands are the 433 MHz used in home automation devices, wireless weather stations, remote light switches and the 868-870 MHz band for asset tracking systems, meter readers, industrial telemetry

and telecommunications equipment, data loggers, in-building environmental monitoring and control systems, social alarms, high-end security and vehicle data upload/download. Unlicensed bands are used also in radio modems, industrial equipment and Wireless Sensor Networks (WSN). Typical applications of WSNs are controlling and monitoring the building's equipment (lightning, ventilation, security systems, fire systems), security systems, habitat monitoring, vehicular tracking, Automated meter reading (AMR) for water, heat and gas. According to CEPT ERC/REC 70-03 recommendation [2] unlicensed 868-870 MHz band in Europe is intended for communication between Short Range Devices (SRD). Knowledge of propagation and error characteristics for this frequency band is fundamental for proper system planning.

Every band is split into frequency channels. Each channel has a regulated width and carries one wireless connection at one time. The operating frequency and channel-width has a big effect on the performance of a wireless device. Wider channel allows higher transmission rate. The higher frequencies, the wider channels, due to the relatively more spectrum, however increased transmission rate comes at the cost of radio propagation or radio distance [3, 4].

Direct communication between all network devices is needed in many applications of wireless networks, therefore the range of radio transceiver is under consideration during the system planning. The transmission range depends on the height of antennas, conditions of radio wave propagation, radio frequency, receiver sensitivity and RF power. The RF power has significant impact on range; increasing the RF power of 6 dB, doubles the range in open space [5, 6].

In Europe, 2.4 GHz devices are regulated to 100 mW of RF power, and the lower 433 MHz band allows 10 mW. 868-870 MHz band is divided into 8 subbands with different maximum transmission power from 5 mW to 500 mW. Only the frequency subband of 869.4-869.65 MHz allows transmitting with RF power up to 500 mW. Using the maximum available RF power in 869.4-869.65 MHz band results in reliable distances of a few kilometers in open-spaced areas.

Correctness and reliability are very important in wireless transmission, therefore wireless devices allow for transmission error occurrence and implement strategies to enhance connectivity. These strategies are acknowledgements, retransmissions, coding schemes, and forward error correction mechanisms, deployed in various layers of network stack protocol. Communication errors cause the need of the data retransmission, which increases the bandwidth usage. Furthermore, transmitting the additional packets consumes additional energy of devices, which implicates shortening of their life-time, if they are battery powered. Understanding the characteristics of the transmission errors is significant for proper choice, design and calibration of mechanisms intended for improving communication.

The 869 MHz subband seems very promising for applications of WSNs or AMRs systems therefore we carried out propagation measurements to determine realistic transmission loss and transmission error characteristics in the 869 MHz band. This paper presents summarized results and error analysis.

The rest of the paper is organized as follows. In section 2. the main sources of transmission errors are depicted. The experiment setup is described in section 3. This section also includes measurement scenario. The error analysis is described in section 4. In section 5. we summarize the results.

## 2. Errors in the wireless channel - sources and correcting strategies

### 2.1. Source of errors

In some percentage of transmitted packets it is observed that one or more bits are corrupted. The errors occur due to the noise and interferences, which are undesirable signals in communication channel. Discharge storms, rainfall or snowfall, humidity are weather conditions causing natural noise. Noise is also generated in electronic components of transceiver and receiver. Random motion of electrons in conductor produces thermal noise. Large amount of noise is artificial, caused by human existence and industry equipment. Error occurrence is also related to interference between radio devices, if at least two transmitters send signal simultaneously in the same or neighboring radio channel. Signal from one transmitter also can interfere at the receiver due to the multipath effect caused by reflection, scattering, and diffraction of transmitted signal. These mechanisms have an impact on the received signal strength in different distances from transmitter. If there is LOS path to the receiver, diffraction and scattering shouldn't dominate the propagation. Likewise, if there is no LOS to the receiver diffraction and scattering will dominate the propagation. The signal fluctuates when changing location because the received signal is a sum of many contributions coming from different directions [3, 7].

Interference can be limited by usage of the directional antennas, selection of free radio channel, implementation of nodes synchronization algorithm to assure that one device is transmitting at the time. Distortion of received data can be corrected by using forward error correction (FEC) [8] or can be corrected using retransmission mechanisms.

### 2.2. Correcting strategies

To apply proper error correction strategies the knowledge regarding error characteristics is necessary. Statistics on the average amount of corrupted bits (bit error rate, BER) or information if errors occur in groups are insufficient. If errors cumu-

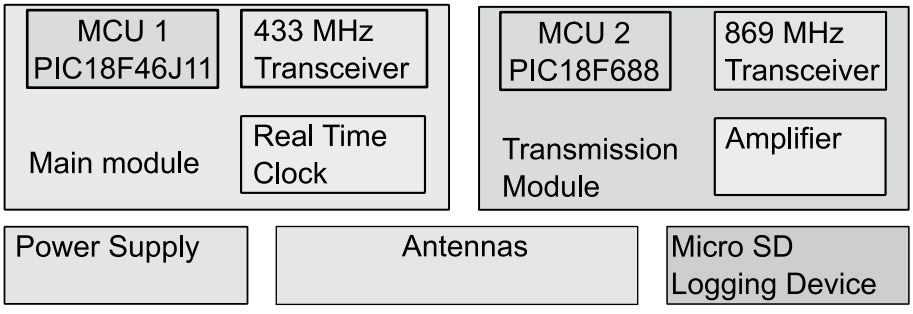



Fig. 1: A schematic diagram of device architecture. The device consists of several independent modules attached to the main board.

late, recognizing relationship between error occurrence is necessary. Large number of corrupted bits may be caused by packets overlap, bit-shift error (bit insertion or removal), jamming and transmitter or receiver failure. Typical transmission errors result from statistical error process and each bit in a packet may be corrupted with the same probability. If error rate is relatively low, the majority of packets have no errors, fewer packets have one bit error, and even fewer packets have more errors.

## 3. Experiment setup

### 3.1. Measurement devices

Devices used in experiment were designed to function as a universal, double-band (433MHz / 869MHz) devices capable of constantly logging the received signal on a replaceable flash card. Each device consists of 3 main parts - the main board, the transmission module and data logger supplemented with power source and antennas (for 433 MHz and 869MHz). The design diagram is presented in figure 1.

The main board consists of several built-in modules and is responsible for 433MHz transmission and power management. It acts as the main controller for the entire node. It is equipped with Microchip PIC18F46J11 MCU operated with 32MHz. A custom-designed operating system has been applied on the MCU. In terms of the system architecture, all the modules form a master-slave system, with main board MCU acting as master. Each sub-module (e.g. 433 Transciever) as well as all the external modules (e.g. MicroSD logger) are connected to the MCU using various types of buses, which allows simultaneous operation for some components. In particular, the 433MHz transciever, the MicroSD data logger and the 869MHz transmission modules are connected using separate buses. The system

allows asynchronous dispatching the bus activity.

The 869MHz transmission module comprises 3 elements - a MCU, 869 MHz transceiver and RF amplifier. The transceiver (Chipcon CC1020) allows usage of various link layer parameters, including diverse range of transmission speed, encodings (Manchester/NRZ) and modulations (ASK/OOK/FSK/GFSK). The transceiver is accompanied by RF amplifier allowing 500mW antennas output. The output power is controlled by the CC1020 power register. As CC1020 lacks MCU features (i.e. it is not programmable), the design includes a peripheral control processor, based on the Microchip PIC16F688. The MCU is responsible for controlling CC1020 and amplifier operations. A custom command set interface is used to communicate with the MCU, using a 115 200bps serial line. The module allows two types of operation: master-slave and constant receive. In master-slave mode, the module is acting as the slave and all the transmission is initiated by an external master. The master-slave mode is used for the transmission of data, configuration, diagnostics and for receiving radio frames. While in constant-receive mode, the module automatically sends data received by the radio transceiver via the serial link every time a byte completion event occurs. A special control line has been dedicated for switching between the modes.

The data logging module consists of a single 4D SYSTEMS uDRIVE-uSD-G1 logger equipped with a 2GB microSD card. The 3D SYSTEMS uDRIVE-uSD-G1 is a compact high performance Embedded Disk Drive module, easily addable to any MCU-based systems. It allows a fast (230kbps) I/O. The SD can be either FAT16-formatted (the reads and writes are performed on a per-file basis) or a RAW data transfer can be performed (the data are read/write operations are performed on the address-byte basis).

During the experiment, only the 869MHz transmission module and MicroSD logging module have been utilized. The main part of experiment utilized 25 kHz width channels, transmitting power was 500 mW. GFSK modulation has been used. The channel width defined the maximum transmission speed of 4800bps. The data has been transmitted using the NRZ encoding with different transmission powers. The transmission power has been modified sequentially, with single cycle defined by: $P_m\, P_4\, P_m\, P_3\, P_m\, P_2\, P_m\, P_1\, P_m\, P_0$, where $P_m = 27dBm(500mW), P_4 = 25dBm, P_3 = 24dBm, P_2 = 23dBm, P_1 = 21dBm, P_0 = 19dBm$.

The device operation consisted of two phases - the constant receive phase and the transmit phase. The transmit phase is performed sequentially every 32 seconds, using a transmission window of 1 second for each node. The transmission window have been bestowed according to the device address (e.g. node 2 should perform transmission every $2^n d$ second of each cycle). Each transmission frame consists of three elements - the preamble, system information part (header) and the test pattern.

The preamble allows frame recognition, while the system information part contains such elements as the node address, frame identification, transmission power (as set by the transmitter) and some other diagnostic information. The system information part is secured using the CRC checksum. The test pattern, described in 4.3., follows the system information part.

Since the nodes have RTC installed (it is included in the main module), it is primarily used for internal operations and time stamps. The network design requires every node to perform synchronized transmission, while the RTC settings among the nodes may be not-valid system wide. For this reason the main module MCU incorporates a synchronization procedure allowing automatic network setup. The algorithm allows automatic network reorganization and single node addition.

During the constant receive phase the main board MCU system performed 3 simultaneous data-related tasks: MicroSD data logging, frame analysis and synchronization. All the data received by the transmission module is logged in 100B chunks on the memory card, with each chunk having a time stamp. The data is saved for future off-line analysis. The second task, namely frame analysis is responsible for on-line extraction of data frames from the byte stream received observed on the 869MHz module bus. The data is extracted to perform some additional diagnostics and to allow proper working of the synchronization algorithm. Each recognized frame is followed by the RSSI readout for the last part of the test pattern. The additional data are stored in separate log file on the flash memory card.

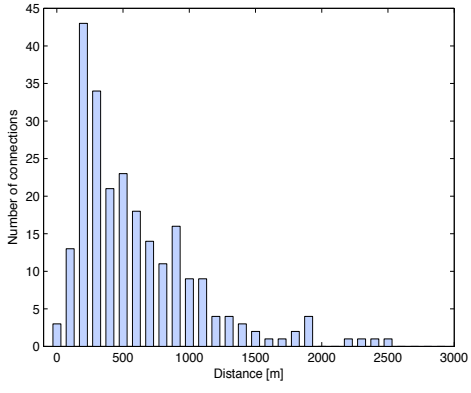
### 3.2. Measurement scenario

Measurements were performed in August and September 2009 in Warsaw. Each day we installed up to 20 devices: 4 concentrators (devices with highly sensitive antennas) and 16 end-devices with quarter-wave antennas. Concentrators were installed on locally high roofs; end-devices were mounted on the walls of buildings at the height of 2.5 meters. During three weeks we installed concentrators on 57 roofs and end-devices on 218 building walls, their location was changed every day. Radio range of concentrators covered almost the whole area (19 $km^2$) of Zoliborz, quarter of Warsaw territory. Surroundings of Zoliborz are differentiated: housing estates with buildings up to 20 floors, residential districts, green terrains.

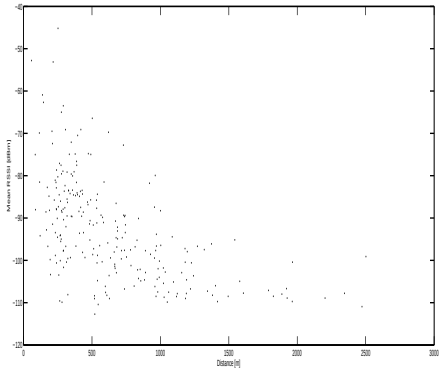
## 4. Propagation measurements and their analysis

Our analysis is based on a set of transmission traces collected during 10 days. Throughout this time 36 concentrators and 135 transmitters (end-devices) were installed in total. We analysed traces collected by concentrators. This resulted in 509 analyzed connections. Among 509 connections, we take into consideration

only those where at least 30 packets were received. This cuts down the number of analysed connections to 239 referred further as active connections. The distance between transmitters and receivers vary from 100 and 2500m. Distance distribution is presented in figure 2a. Most active connections (83%) were recognized for devices remaining at distance no longer than 1500m. Only several connections between devices at longer distance were noticed.




(a)



(b)

Fig. 2: Distance distribution and mean RSSI for all active connections

For each frame signal power was measured. Mean RSSI (Received Signal Strength Indicator) for each considered connection is gathered in figure 2b. Obviously the highest RSSI was measured for devices in direct proximity however the RSSI varies form maximum to the lowest possible value (around $-107$ dBm) even

| | Number | PER [%] | BER [%] |
|---|---|---|---|
| All frames | 390 154 | 25.27 | 2.19 |
| Frames with CRC OK | 320 722 | 13.02 | 0.8 |
| Frames with CRC not OK | 69 432 | 81.86 | 8.61 |

Table 1: Mean error rate in received frames

| | bits/bytes | bytes | bits |
|---|---|---|---|
| All frames | 2.78 | 6.30 | 17.49 |
| Frames with CRC OK | 3.03 | 2.10 | 6.37 |
| Frames with CRC not OK | 3.63 | 25.67 | 68.89 |

Table 2: Mean error length in received frames

at short distances.

Further analysis consists of:

- classification of errors, percentage rate of each error category, mean error rate and length,
- influence of eakening the RSSI on bit error rate,
- recognizing error patterns.

### 4.1. Error classification and statistical results

During 10 days, for all active connections 390 154 frames were recognized, from which 320 722 had a correct CRC. Table 1 presents number of packets in each category, mean packet error rate and bit error rate for all frames and separately for frames with correct and faulty CRC. CRC is send in packet header. Calculation of PER and BER is performed basing on pattern trace excluding header therefore not all frames without correct CRC are marked as corrupted.

Next table 2 contains information concerning error length. First column indicates how many bits on the average are corrupted in a corrupted byte. Next two columns contain mean number of corrupted bytes and bits respectively. All values are given for all frames and separately for frames with proper and false CRC. Once again mean error lengths are calculated for pattern traces excluding header and tail.

25% of all frames is corrupted, in most cases errors can be recognized basing on comparison of calculated and received CRC. The error rate among frames with

| | Synchronization | Hardware | Transmission |
|---|---|---|---|
| Frames CRC OK[%] | 18.34 | 14.34 | 67.32 |
| Frames CRC not OK[%] | 14.02 | 38.59 | 47.38 |

Table 3: Classification of errors

| | Synchronization | Hardware | Transmission |
|---|---|---|---|
| Frames CRC OK [bytes] | 45.73 | 43.28 | 2.30 |
| Frames CRC OK [bits] | 174.55 | 103.71 | 3.03 |
| Frames CRC not OK [bytes] | 39.55 | 62.30 | 3.73 |
| Frames CRC not OK [bits] | 129.59 | 165.57 | 4.39 |

Table 4: Mean error length in particular classes of errors

corrupted CRC exceeds 80%. Surprisingly 20% of frames with faulty CRC were received without any errors in pattern. The percentage of corrupted frames among frames with correct CRC surpasses 10% which seems rather high however very low bit error rate suggest low error rate per frame, which is confirmed by values in table 2.

Error were classified in three classes: synchronization errors, hardware errors and transmission errors. Synchronization errors occur due to the imperfections of synchronization algorithm allowing more than one device transmit at the same time. All hardware can be attributed to the receivers more precisely to the logger module. The synchronization errors along with hardware errors are network specific therefore we do not include them in further analysis. We focus on transmission errors that expose some coding and frequency disadvantages. The percentage rates of each error class are grouped in table 3.

Transmission errors make up from half to over two third of all errors. Maximum number of corrupted bits in packets classified as transmission errors doesn't exceed 12 bits which is 1.5% of the whole frame, mean error length for transmission errors is about 3 bits with frames with properly recognized header. If the errors don't show up in groups they are corrigible using correcting strategies such as forward error correction.

Mean error length for synchronization and hardware errors indicates that on average half of the frame is corrupted.

| | Synchronization | Hardware | Transmission |
|---|---|---|---|
| Frames CRC OK [dBm] | $-87.93$ | $-96.46$ | $-97.72$ |
| Frames CRC OK [dBm] | $-89.54$ | $-98.70$ | $-98.06$ |

Table 5: Mean RSSI in particulat classes of errors

### 4.2. RSSI influence on error rate

We evaluated mean RSSI for packets in all error classes. Table 5 contains calculated values. RSSI for synchronization errors is the highest which is rather obvious. Receiver adjusts to the stronger signal therefore measured RSSI is higher.

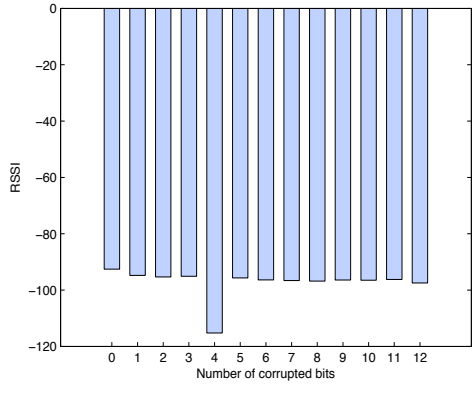
Figure 3 presents mean RSSI for frames with particular number of corrupted bits divided into frames with correct and corrupted header. For frames with properly calculated CRC the differences are negligible. The lowest value for frames with four bits corrupted results from significantly lower number of such frames and measurements. For frames with false CRC the difference in RSSI between frames without errors and frames with 12 bits corrupted is around 6 dBm but doesn't drop below -100 dBm. For those frames insignificant lowering of RSSI can be notices while number of corrupted bits increases.

### 4.3. Error patterns

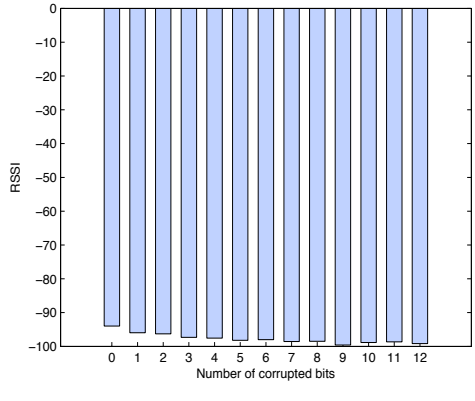
Last section contains analysis concerning error pattern recognition. For all frames with transmission errors, we calculated number of errors on every bit of the pattern. Figure 4 presents the whole transmitted pattern (the bottom of the plot) and calculated number of errors on all bits. 60% of pattern consists of randomly selected bits. Last 40% of the pattern is made up by alternating sequences of different lengths of 1s and 0s. Further analysis is divided for this two parts of the whole pattern.

Figure 5 presents error pattern for random bits. It can be observed that higher number of corrupted bits appears after longer sequence of 0s. This is the result of NRZ encoding. It is confirmed in the next two figures presenting results for alternating sequences of 0s and 1s 6.

The longer sequence of the same bits the higher probability of corrupting the first opposite bit. The probability is much higher when changing form 0 to 1 than otherwise however probability of error is increased. Sequence of similar length of 0s and 1s resulted in repetitive error pattern (fig. 6a). Longer sequences caused much higher error rate (fig. 6b).




(a) with correct CRC




(b) with corrupted CRC

Fig. 3: Mean RSSI for frames with particular number of corrupted bits

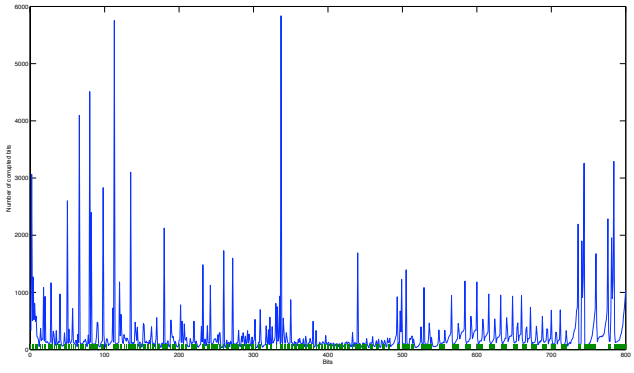



Fig. 4: Cumulative number of corrupted bits for the whole pattern for frames with transmission error

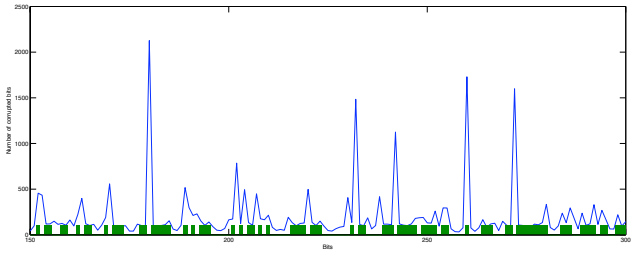
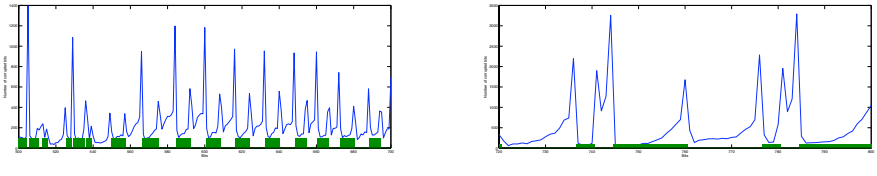



Fig. 5: Cumulative number of corrupted bits for the part of the pattern with random bytes for frames with transmission error



(a) (b)

Fig. 6: Cumulative number of corrupted bits for the part of the pattern with alternating sequences of 1s and 0s for frames with transmission error

## 5. Conclusions

Believing that 869 MHz band is very promising for application of WSN networks, we carried out extensive measurements to determine realistic transmission loss and transmission error characteristics.

The RSSI measured by the receiver mostly depends on the distance between transmitter and receiver. The effective range of devices transmitting with 500 mW is up to 1000 - 1500 meters in dense, urban environment. The longer distance, the lower RSSI which results in increasing error rate.

We recognized three classes of errors: synchronization, hardware and transmission errors. The first two categories are network and device specific therefore we do not include them in further analysis. Transmission errors are the most common however they do not introduce much distortion in a single frame. Mean error length for frames with properly received RSSI doesn't exceed 1% of the whole packet.

There is also connection between send data, encoding and probability of error corruption. We used NRZ coding and it could be observed that the longer sequence of 1s or 0s the higher probability of corrupting the first alternating bit. The probability was higher after the sequences of 0s.

## References

[1] ITU page on definitions of ISM bands. http://www.itu.int/ITU-R/terrestrial/faq/index.html

[2] ERC recommendation 70-03. www.erodocdb.dk/Docs/doc98/official/pdf/REC7003E.PDF

[3] Szostka, J.: *Anteny i fale*. Wydawnictwa Komunikacji i Lacznosci, WKL (2006)

[4] Wesolowski, K.: *Systemy radiokomunikacji ruchomej*. Wydawnictwa Komunikacji i Lacznosci, WKL (2003)

[5] Basagni, S., Conti, M., Giordano, S., Stojmenovic, I., eds.: *Mobile Ad Hoc Networking.* John Wiley & Sons, Inc., New York, NY, USA (August 2004)

[6] Goldsmith, A.: *Wireless Communications.* Cambridge University Press, New York, NY, USA (2005)

[7] Lee, J., Perkins, M., Kyperountas, S., Ji, Y.: *Rf propagation simulation in sensor networks.* In: SENSORCOMM '08: Proceedings of the 2008 Second International Conference on Sensor Technologies and Applications, Washington, DC, USA, IEEE Computer Society (2008) 604–609

[8] Lin, S., Costello, D.J.: *Error Control Coding, Second Edition.* Prentice Hall (April 2004)

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 405–418

# Measurement based model of wireless propagation for short range transmission

ZBIGNIEW LASKARZEWSKI AGNIESZKA BRACHMAN

Institute of Computer Science
Silesian Technical University
*{zbigniew.laskarzewski, agnieszka.brachman}@polsl.pl*

**Abstract:** Transmission in 869 MHz band with 500mW offers range of up to 5 kilometers in open space. Understanding of wireless propagation phenomena in this band allows us to predict RSSI in any place of interesting area with better accuracy. This knowledge helps us to choose installation places for concentrators or base stations to cover desired locations of end-points.

In this paper we present detailed description of proposed propagation model DUAS. Model uses elevation data in SHP format and is based on the Two-Ray Ground and ITM propagation models. Our model was parameterised and verified comparing to propagation measurements, which were collected during 10 days in diversified, urban environment. In addition we presents analysis of existing wireless propagation models, basic mechanisms of radio wave propagation, existing RF propagation tools and elevation data formats.

**Keywords:** wireless propagation, propagation models, WSN, 869 MHz

## 1. Introduction

Wireless Sensor Networks (WSNs) play an important role in many industrial and civil applications concerning asset tracking systems, controlling and monitoring the building's equipment (lightning, ventilation, security systems, fire systems), security systems, habitat monitoring, vehicular tracking, Automated Meter Reading (AMR) for water, electricity, heat and gas. The choice of radio frequency for WSN devices is determined by many issues. First of all, the regulations in most countries allow using without license specific areas of the spectrum. The license-free bands are intended for industrial, scientific and medical purposes [1]. The most common bands available throughout Europe, are: 2.4 GHz, 869 MHz, 433MHz. Every band is split into frequency channels. Each channel has a regulated width and carries one

wireless connection at one time. The operating frequency and channel-width has a significant effect on the performance of a wireless device. Wider channel allows higher transmission rate. The higher frequencies, the wider channels, due to the relatively more spectrum, however increased transmission rate comes at the cost of radio propagation or radio distance [2, 3, 4, 5]. In most cases the higher frequency will not be appropriate for WSNs where very short distance is a serious flaw.

The physical radio transmission is based on the emission of electromagnetic waves. Radio waves decrease in amplitude as they propagate and pass through obstacles [3, 4, 5]. The wireless propagation is influenced by many factors: transmission medium, transmitting power, number of devices in the network, radio frequency, channel width, modulations, coding, height of antennas, terrain characteristics, weather conditions and many others. Direct communication between all network devices is needed in many applications of wireless networks, therefore the range of radio transmitter is under consideration during the system planning.

Radio propagation path loss models are a very important tool for determining radio coverage and signal strength. These models are very useful during network planning, expanding and optimizing. In most cases WSNs are distributed networks. Information is collected from multiple locations simultaneously. Therefore, an important aspect during planning such a network is to design an appropriate system architecture for example sink arrangement.

A variety of analytical and measurement based models have been developed to predict radio propagation [5]. The natural decay of transmitted signal can be modeled using analytical approximations. One of the most commonly applied propagation models is Free Space Propagation Model [5]. In this model the receiving power is proportional to $1/d^2$, where $d$ is the distance between sender and receiver in open space. The Free Space Propagation Model can be extended with the Two-Ray Ground Model [5, 6]. This model is similar to the Free Space Propagation Model, except that for distances greater than reference distance, receiving power is modeled as proportional to $1/d^\beta$, $\beta > 2$. This is because the model considers both: the direct path and a ground reflection path. The Free Space Model and Two-Ray Ground Model assume ideal propagation over circular area around the transmitter.

The ITS model of radio propagation for frequencies between 20 MHz and 20 GHz (the Longley-Rice model/ ITM model), named for Anita Longley and Phil Rice is a general purpose model that predicts the median attenuation of a radio signal as a function of distance and the variability of the signal in time and in space [7].

Numerous empirical path loss models are available for predicting propagation in urban, suburban or rural areas. They are often limited to the specific spectrum range (from 900MHz to 2GHz) and to the large ranges (1 - 20 km). The most

popular model is proposed by Okumura [8]. He published some curves for predicting signal strength for a given terrain, for the frequency range from 150 MHz to 1.5 GHz,distances of 1 to 20km, effective antenna height from 30 - 200m and the effective receiver height 1 - 10m. Hata developed formulas for Okumura's result therefore model is known as Okumura-Hata model and is easy to use in computer systems [9]. Walfish and Ikegami [10, 11] proposed theoretical models that considers additional characteristics of the urban environment: height of buildings, width of roads, building separation, road orientation. Model is appropriate for systems with frequencies from 800 - 2000 MHz, distances of 20 to 5000m, effective transmitter and receiver heights: 4 - 50m and 1 - 3m. Model requires information concerning mean building height, road width and mean distance between buildings. Both models were extended under COST 231 project to adjust models to the European specific conditions [6].

Problems with the experimental models is that the formulas are based on the qualitative propagation environments (urban, suburban, rural) and they don't take into account differentiated infrastructure. Since the radio link is highly variable over short distances, these models do not provide satisfying accuracy. Ground elevation, the exact height of buildings and spaces between them significantly influence radio wave propagation and none of the aforementioned model takes these factors into account.

This paper shows how the Two-Ray Ground and the ITM model may be tuned to approximate results obtained during the experimental setup of 869 MHz network in Warsaw. The purpose of this study is to determine the propagation model for urban environment as a tool for WSN network deployment. We carried out the propagation measurements in the 869 MHz band to determine the realistic transmission loss.

The remainder of the paper is organized as follows. Section 2. indicates some difficulties in predicting radio wave propagation due to the physics of radio wave propagation. Section 3. provides a brief description of existing tools and elevation data formats that can be used for those applications. Section 4. describes the proposed model, necessary data conversions and transformations as well as achieved results. Concluding remarks and future work suggestions are provided in section 5.

## 2. The physics of propagation

The mechanisms of radio wave propagation are complex and diverse. There are three basic propagation mechanisms: reflection, diffraction and scattering that attributes to this complexity. Radio waves decrease in amplitude as they pass through walls. As the radio frequency increases, the rate of attenuation increases - that is,

the radio strength dies off faster, and the effect of passing through obstacles is much greater.

Reflection occurs when a propagating radio wave impinges upon an obstruction with dimensions very large compared to the wave length. Some part of the wave penetrates the obstacle (and refracts), the rest reflects from the surface. The condition regarding the size of an obstacle is usually fulfilled when waves encounter buildings, walls, mountains or simply the ground.

Diffraction occurs when the radio path between transmitter and receiver is obstructed by an impenetrable object. Secondary waves are formed behind the obstacle even there is no Line-Of-Sight (LOS) between the transmitter and receiver. This enables transmission even if there is no LOS which is very common in most urban environments. Diffraction happens when a signal hits on the irregularities of an obstacle, for example edges, corners.

Scattering occurs when the radio wave finds an object with dimensions that are on the order of the wave length or less. Scattering follows the same physical principles as diffraction, causes energy of wave to be radiated in many directions. It is the mechanism that introduces the great amount of unpredictability. Lamp posts, street signs, foliage of trees can scatter energy in many directions thereby disturb predicted propagation. The copies are much weaker than the hitting signal.

These mechanisms have an impact on the received signal strength in different distances from transmitter. If there is LOS path to the receiver, diffraction and scattering shouldn't dominate the propagation. Likewise, if there is no LOS to the receiver diffraction and scattering will dominate the propagation. The signal fluctuates when changing location because the received signal is a sum of many contributions coming from different directions [4, 5, 12].

The multipath propagation is one of the major problem causing inaccuracy of propagation models. Different frequency components of a signal are affected in different degrees and waves. For short distances and dense urban areas the multipath propagation requires better models and taking into consideration lay of the building and terrain.

## 3. RF propagation tools and elevation data

There are several existing applications for calculating radio propagation for example: SPLAT, Radio Mobile, TAP, Matlab, OPNET. Only SPLAT and Radio Mobile are available for free. Some of these applications use elevation data for better map generation. In this section we describe the different, available elevation data format, which are commonly used in applications for radio propagation prediction as well as applications that we used in further model development.

### 3.1. Elevation data

Geographic Information System (GIS) is a system for acquisition, storage, processing, analysis and sharing spatial information having a reference to the Earth's surface. There are two basic methods for spatial data representation, namely raster and vector based. In raster method, area is divided into rows and columns, which form a regular grid. Each cell within a grid contains location coordinates and an attribute value. In some file formats, coordinates are determined by a cell position in a file. Vector data is comprised of lines or arcs, defined by beginning and end points. The most common representation of a map using vector data consist of points, lines and polygons. We focused on three data formats: vector based - ESRI and raster based - SRTM, BIL.

ESRI Shape Format [13] is a collection of file formats for vector data representation developed by Environmental System Research Institute. The collection consists of a set of compulsory and optional files. There are three mandatory files: *.shp, *.dbf, *.shx. The *.shp file contains objects definition. Objects are represented by points, lines and polygons. *.dgn file allows attaching attributes i.e. elevation, material, color, etc. *.shx is a shape index format - a positional index of the feature geometry to allow seeking forwards and backwards quickly. ESRI Shape Files provide the most accurate description of the area with information on all objects, their mutual position and dimensions. These data are not available publicly.

The Shuttle Radar Topography Mission (SRTM) [14] obtained elevation data to generate high-resolution digital topographic database of Earth. SRTM at a 3 arc-second resolution ($\sim$ 90m at the equator) are freely available for most of the globe. Additionally, 1 arc second ($\sim$ 30m at the equator) SRTM are available in various formats for the continental United States. SRTM data are often distributed in files with .hgt extension. SRTMv3 data is divided into squares with dimensions of 1x1 degree, more precisely the side length is equal to 1.0083333 degree, which makes squares lying next to each other overlap. Grid in every square is a matrix of 1201x1201 signed shorts (requiring 2 bytes). Every cell has dimensions $\Delta\varphi = \Delta\lambda = 3'' = 0.00083333$. The most outer data in each file overlap. Every square is written in a binary file with .hgt extension. All binary files have identical organization and size of 2 884 802 bytes, they contain no header or ending, only raw data. The numbers in matrix are organized by row - the first row describes points lying at the north. First digit in the first row is the most western one.

BIL files contain elevation data in the raster format. Values in matrix are organized in rows, like in .hgt files. .bil file contains no header. In addition to binary file come two text files: .hdr and .blw. .hdr file contains basic information concerning organization of data (Little Indian, Big Indian), number of columns and rows, co-

ordinates of the left upper corner, horizontal and vertical distance between points. In .blw file there are repeated information concerning coordinates of the left upper point and horizontal and vertical distance between points. BIL files are available for selected areas of United States. The resolution of data is about 1/9 degree that is around 3 meters. Due to the full parametrisation through header files, theoretical resolution is voluntary.

### 3.2. RF prediction software

From aforementioned propagation tools, we have chosen Radio Mobile and Matlab to implement our model. We used Radio Mobile [15] to generate basic propagation maps. These maps were tunned using Matlab.

Radio Mobile is dedicated to amateur radio and humanitarian use. It uses digital terrain elevation data for automatic extraction of path profile between transmitter and receiver. Radio Mobile will determine if a radio link is Line-Of-Sight or not. In case of LOS the Two-Ray Ground calculation is used. In case of an obstructed path the Longley-Rice model (ITM) is applied. To evaluate LOS, the clearance of the first Fresnel zone is determined. The elevation data is added to system to feed the ITM radio propagation model. The main advantage of this program, in comparison to others, is the ability to use elevation data in BIL format. Since we focus on propagation over short distances, high resolution of input data was essential.

## 4. DUAS propagation model

Our propagation model DUAS (Dense Urban Area Simple propagation model) is a three step algorithm: the preparation of elevation data, usage od free available software tools for generating preliminary grid of signal strength and the post-processing of output data. We believe that using accurate elevation data is fundamental for improving prediction accuracy, especially for short distances and dense, urban environment. Therefore we focused on models and tool enabling using of such data.

A few tests with SRTM-3 showed us that the resolution of 90m is insufficient. Simulation results obtained with the SRTM-3 data were significantly different from measurements, especially for short distances between the transmitter and the receiver. The SRTM-3 data don't give information concerning particular location of buildings which results in high error of signal strength prediction. Only the BIL format offers desired resolution, therefore we decided to use it along with Radio Mobile - free software for modeling the wireless propagation channel. We prepared tentative, simple map in BIL format with area covered by several buildings and triggered generation process in Radio Mobile. Accomplished tests allowed

noting, that Radio Mobile takes into account the clearance of the first Fresnel zone and applies the Two-Ray Ground reflection model, otherwise the phenomenon of building shadowing is visible. Reflection, refraction, scattering and diffraction are not implemented in Radio Mobile, however we assumed that the post-processing of output data from Radio Mobile would improve accuracy of signal strength prediction.

### 4.1. Preparation of elevation data

In the beginning, we needed maps that would allow preparing BIL files in the satisfying resolution, i.e. of the high accuracy. We have obtained files containing the buildings locations along with their roof heights. Files are written in SHP format and the location of objects is stored using Gauss-Krüger coordinate system. This system is based on Gauss-Krüger projection, and it is intended for high resolution maps (map scale 1:10000 and less). Gauss-Kruger projection is a transverse Mercator map projection with tangential coincidence, here used with 3-degree zone. Transformation of SHP file to a grid, which we could use in the BIL file, required two steps: transformation of coordinates to WSG-84 system and generating grid in the required resolution. Transformation from Gauss-Kruger to WSG-84 projection is complex, therefore approximate methods for the transformation were used. Grid was generated using MapWindow GIS [16], free extensible geographic information application.

Grid file, generated with MapWindow GIS, contains invalid data regarding altitude among buildings. Filling these empty areas with fixed value e.g. 80 meters, decreases accuracy of land mapping and causes improper signal strength prediction. Empty areas were filled using data from SRTM-3 files. Interesting area of Zoliborz, district of Warsaw, is covered by bounding box with coordinates 52-53 degrees northern latitude and 20-21 degrees eastern longitude which corresponds to one file - N52E020.hgt, available online [17]. Low resolution and low accuracy of SRTM-3 data were reasons for regular distortions of wireless propagation, therefore SRTM-3 data was averaged with higher resolution, corresponding to the resolution of BIL file, to smooth edges of the map. Described transformation results are presented in a final map in figure 1.

### 4.2. Modelling in Radio Mobile

We decided to use Radio Mobile, the free software tool, which simulates RF propagation using the Two-Ray Ground and ITM model. Input data for Radio Mobile are: BIL file with elevation data, prepared as described in previous subsection, files with locations of transmitters (end-devices) and receivers (concentrators). Lo-

Fig. 1. Elevation data after transformations

cations were defined in XML files along with information about address, coordinates and height of device. Definition of single point is presented below:

```
<Placemark>
        <name> Heroldow19 </name>
        <visibility> 1 </visibility>
        <Point>
          <extrude> 1 </extrude>
          <altitudeMode> absolute </altitudeMode>
          <coordinates> 20.93005,52.30346,119
          </coordinates>
        </Point>
</Placemark>
```

Concentrators were installed on locally highest points of buildings' roofs, their coordinates were registered during installation. Due to the imperfections of all performed transformations, these coordinates had to be corrected manually. Program Radio Mobile doesn't run in batch mode; generation of propagation map for each concentrator had to be executed manually. The result of each program run is the

propagation map and the output text file. Header of the output file contains information about localization and parameters of the concentrator. The body of the output file contains the estimated signal strength value in every point of the map. File resolution is user-defined.

Radio Mobile model parameters like antenna gain and type, sensitivity of receivers, attenuation and absorption constants and others have an influence on estimated signal strength value. Selection of appropriate parameters for the propagation model required many generations of single propagation map. For parameterization we used measurement results between end-devices and concentrators in Line-Of-Sight only. We observed that estimated signal strength value, for places obscured by buildings, is significantly lower than measured signal strength. It is because Radio Mobile doesn't take into account the phenomena of radio waves reflections and diffractions among buildings, which is particularly important in built-up areas. Further processing of Radio Mobile results, reduces difference between estimated values and measurements.

### 4.3. Post-processing in Matlab

Basing on the results obtained with Radio Mobile, we implemented a mathematical model to improve accuracy of prediction in build-up areas. Implementation was accomplished in Matlab environment. At this stage we use also grid file for Matlab with elevation data. This should be a text file, its resolution must be identical to the resolution of the output file form Radio Mobile. Grid file can be prepared from BIL file used in previous task.

In order to improve the simulation model, completion of two steps was necessary: identification of areas causing highest errors and suitable correction of estimated signal strength value. We verified that calculation of signal strength for points without LOS, in dense urban areas, were encumbered with the highest error. Considering methods for improving prediction accuracy in dense urban areas, we wanted to preserve good prediction results in other areas. During implementation we took into account several values, which we calculated for each point at the map:

1. Signal strength from Radio Mobile without any transformations,
2. Average value of signal strength (from 1) in a square with analysed point in the center,
3. Average value of signal strength (from 1) in a square with analysed point in the center, excluding values calculated on the buildings' roofs,
4. Maximum value of signal strength (from 1) on the buildings' roofs in a square with analysed point in the center.

Square's side length for averaging ranged from 5 to 205 meters in tests. Every aforementioned value was taken into consideration and added with various significance.

After many tests we have chosen, for further calculations, two values. The first one is the mean value in a square with analysed point in the center, without values on the buildings' roofs (no. 3), which corresponds to the LOS wireless propagation and the Two-Ray Ground Model. The second value is the maximum value of a signal on the buildings' roof in a square with analysed point in the center (no. 4), reduced by 20 dB, which corresponds to probable reflection phenomenon. If the first value is greater then the second, the estimated signal is set with the first value. Otherwise, estimated signal includes 20% of the first value and 80% of the second value. Square's side length for averaging was set to 55 meters. In dense urban areas, where distances among buildings are less than 100 meters and buildings are high (signal strength from the roof is high), maximum value of signal strength on the building roof is included.

### 4.4. Model verification

During model parameterization, we evaluated model accuracy by calculating Total Error defined as average of absolute error values, i.e. average of absolute values of differences between measured and estimated signal strength indicators.

$$TotalError[dB] = \frac{\sum\limits_{n} |\Delta RSSI|}{n}$$

where $n$ is the number of analysed connections.

$\Delta$RSSI (Received Signal Strange Indicator) is calculated as difference between measured and estimated signal strength. Sign of $\Delta$RSSI indicates tendencies of false estimation - pessimistic or optimistic. The closer is the value of $\Delta$RSSI to zero, the better, however, a situation when we estimate lower level of signal strength than we measure (pessimistic estimation) is more preferable, than vice versa.

Estimated values were compared with measurements performed in August and September 2009 in Warsaw. We installed concentrators with highly sensitive antennas on locally high roofs and end-devices with quarter-wave antennas on the walls of buildings at the height of 2.5 meters. Radio range of all 57 concentrators covered almost the whole area (19 km$^2$) of Zoliborz, quarter of Warsaw territory. Surroundings of Zoliborz are differentiated: housing estates with buildings up to 20 floors, residential districts, and green terrains.

Our model accuracy was evaluated by comparing signal strength values of 272 connections between concentrators and end-devices. The total error amounts to 8.5

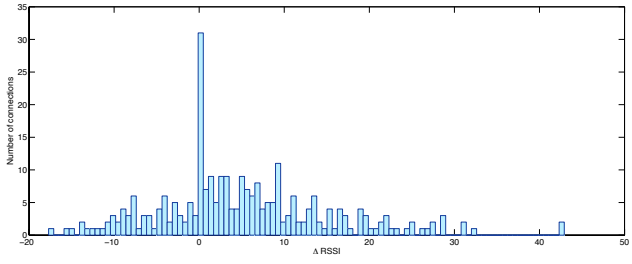



Fig. 2. ΔRSSI distribution for all analysed connections

dB for all analysed connections. Value of total error means that average error of estimation is equal to 8.5 dB. ΔRSSI distribution is presented in figure 2.

High number of connections, where estimation error is 0, is a result of proper prediction whether coordinator is not in the radio range of end-device. In our opinion, total error is quite small and allows the model to be used for signal strength estimation.

Estimation error of 8.5 dB is significant when end-device and concentrator are a long way apart. In this case, estimation error of signal strength may cause wrong assumption whether there is an active connection between end-device and concentrator or there is not. 217 from 272 analysed connections between end-device and concentrator were active and 55 of them didn't work during measurements. Depending on the estimated value of signal strength, we classified connections as:

- less than -109 dBm - no connection
- from -109 dBm to -103 dBm - possibly no connection
- from -103 dBm to -98 dBm - possibly active connection
- above -98 dBm - active connection

Quantities of connections classified to each category is presented in table 1.

Estimation error means that we wrongly assumed active (or possibly active) connection between end-device and concentrator and there was no (or possibly no) connection or vice versa. High estimation error means that we wrongly assumed no connection and there was a connection (underestimation) or vice versa (overestimation). High overestimation and underestimation probabilities amount to 16.4% and 18.0% respectively. Level of large errors is not so high and the model could be used to predict existence of connections between wireless devices.

| Measurements / Estimations | No connection | Active connection |
|---|---|---|
| No connection | 30 | 39 |
| Possibly no connection | 8 | 27 |
| Possibly active connection | 8 | 38 |
| Active connection | 9 | 113 |
| Sum | 55 | 217 |
| Estimation error [%] | 30.9 | 30.4 |
| High estimation error [%] | 16.4 | 18.0 |

Table 1. Quantities of classified connections

## 5. Conclusions

The RSSI measured by the concentrator mostly depends on the distance between end-device and concentrator. The range of devices in the 869 MHz 500 mW band with LOS path reaches up to 5 kilometers, but it decreases to 1000-1500 meters in dense urban area and RSSI is highly variable. RSSI in dense urban areas is hard to predict because of the growing importance of propagation mechanisms: reflection, diffraction, and scattering. There are several applications and models for calculating radio propagation, but they don't predict RSSI with satisfying accuracy in dense urban areas.

Using accurate elevation data is the proper method to improve RSSI prediction accuracy, especially for short distances and dense urban areas. Our model DUAS uses SHP and SRTM-3 maps as input data, Radio Mobile application for generating preliminary grid of signal strength and Matlab environment for further processing. Average error of RSSI estimation amounts to 8.5 dB, therefore the model can be used for signal strength estimation. This quite small average error allows us to predict existence of active wireless connection between end-device and concentrator with overestimation and underestimation probabilities amounting to 16.4% and 18.0% respectively.

## References

[1] ITU page on definitions of ISM bands. http://www.itu.int/ITU-R/terrestrial/faq/index.html

[2] Szostka, J.: *Anteny i fale*. Wydawnictwa Komunikacji i Lacznosci, WKL (2006)

[3] Wesolowski, K.: *Systemy radiokomunikacji ruchomej*. Wydawnictwa Komunikacji i Lacznosci, WKL (2003)

[4] Basagni, S., Conti, M., Giordano, S., Stojmenovic, I., eds.: *Mobile Ad Hoc Networking* John Wiley & Sons, Inc., New York, NY, USA (August 2004)

[5] Goldsmith, A.: *Wireless Communications.* Cambridge University Press, New York, NY, USA (2005)

[6] *COST 231 Final Report.* http://www.lx.it.pt/cost231/final-report.htm

[7] *Description of the ITM/Longley-Rice model.* http://flattop.its.bldrdoc.gov/itm.html

[8] Y. Okumura, E. Ohmori, T.K., Fukuda, K.: *Field strength and its variability in VHT and UHF land-mobile radio service.* In: Rev. Elec. Comm. Lab. (1968) 825–878

[9] Hata, M.: *Empirical formula for propagation loss in land mobile radio services.* In: IEEE Transactions of Vehicle Technology. (1980)

[10] Walfish, J., Bertoni, H.: *A theoretical model of UHF propagation in urban environments.* In: IEEE Transactions on Antennas and Propagation. (1988) 1788–1796

[11] Ikegami, F., Y.S.T.T., Umehira, M.: *Propagation factors controling mean field strength on urban streets.* In: IEEE Transactions on Antennas and Propagation. (1984) 822–829

[12] Lin, S., Costello, D.J.: *Error Control Coding, Second Edition.* Prentice Hall (April 2004)

[13] *ESRI shapefile technical description.* www.esri.com/library/whitepapers/pdfs/shapefile.pdf

[14] *Shuttle Radar Topography Mission.* www2.jpl.nasa.gov/srtm/

[15] *Documentation for Radio Mobile.* http://radiomobile.pe1mew.nl/?Welcome...

[16] *MapWindowGIS official site.* http://www.mapwindow.org/

[17] *Shuttle Radar Topography Mission, file repository.* http://dds.cr.usgs.gov/srtm/version2-1/SRTM3/Eurasia/

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 419 – 428

# Self healing in wireless mesh networks by channel switching

KRZYSZTOF GROCHLA KRZYSZTOF STASIAK

Proximetry Poland Sp. z o.o.
Katowice, Al. Rozdzieńskiego 91
{*kgrochla*|*kstasiak*}*@proximetry.pl*

**Abstract**: The wireless mesh networking is a technology for building very reliable and self-organizing wireless networks. The self-healing algorithms in mesh networks provide functions to automatically adapting and repairing the network in response to failures or other transmission problems. In this paper we present a novel algorithm for automatic channel switching in wireless mesh network, which helps to react to the interferences blocking the data transmission. The link state is constantly monitored using received signal strength (RSSI) and bit error rate. When problems are perceived one of the nodes starts the channel switching procedure and other mesh nodes follow it by monitoring the packed received from non-orthogonal channels. The method is discussed in detail, together with some results of measurements of sample implementation on OpenWRT platform.

**Keywords**: wireless mesh networks, reliability, self-healing, channel assignment.

## 1. Introduction

For the last few years we have been experiencing a rapid growth of interest in mobile ad-hoc networking. The wireless mesh networks, comprised of nodes with multiple radio interfaces routing the packets, are a promising technology for example for broadband residential internet access or to provide connectivity to temporal events. Wireless mesh networks (WMNs) consist of mesh routers (nodes) and mesh clients, where mesh routers have minimal mobility and form the backbone of WMNs [1]. They provide network access for both mesh and conventional clients. The links in WMN may use single or multiple wireless technologies. A single mesh node may be equipped with one or multiply wireless interfaces. A WMN is dynamically self-organized and self-configured, with the nodes in the network automatically establishing and maintaining mesh connectivity among themselves (creating, in effect, an ad hoc network). This feature brings many advantages to

WMNs such as low up-front cost, easy network maintenance, robustness, and reliable service coverage.

In order to simplify network deployment, the autoconfiguration procedures providing automatic network start-up with minimum manual configuration of the nodes are increasingly important [5]. To maximize the utilization of radio resources the efficient algorithms to select optimal channel to the current radio propagation condition are required [2]. The algorithms to manage quality of service resources reservation allows greatly increase the usability of the network. All these algorithms are being developed within the EU-MESH project [3], which aims to create novel configuration procedures, resource management, QoS routing, mobility support and self-healing algorithms that achieve efficient usage of both the wireless spectrum and fixed broadband access lines. In this paper we try to extend these algorithms by self-healing procedures providing the network methods to automatically react to interferences in data transmission.

The self-healing procedures inside the mesh network provide methods for repairing the network connectivity in response to failures or interferences. They should continuously monitor the state of the network and reconfigure the wireless interfaces when misbehaviour is detected.

The self-healing actions may be executed on two levels:

- locally on mesh node,
- centrally, on the network management server.

In this paper we concentrate on locally executed actions. The locally executed actions allow for very fast reaction to communication problems. The agent working on mesh device constantly monitors the required parameters and statistics of a mesh node. When an anomaly is detected a local repair action may be triggered. The actions are executed in distributed manner on the mesh nodes, without global coordination and without execution of communication protocol. We have developed the distributed actions for channel switching.

The self-healing action limit the total cost of ownership and minimize the time when the network is not operational, by performing automatic reconfiguration of the network in response to failures or lower network performance. The drop of performance is often caused by interferences, especially in unlicensed bands where many other devices may transmit on the same channels. Most of the works related to interferences in WMN take them into account as a part of routing metric – see e.g. [6], [7], unfortunately this method does not allow to avoid the interferences by reconfiguration of the devices – the traffic will be rerouted using another links. Another common method is to use channel assignment algorithm for selecting channels which experience low interferences [8] [9]. This provides methods for selecting optimal channel for the wireless links, but typically the channel assignment algorithm works periodically and the interferences may appear at any

moment of the transmission. In this work we try to join the approach of interference aware channel selection with monitoring of the link quality to provide automatic, local procedure to change the channel when interferences appear.

In this work we concentrate on the wireless mesh networks build of devices having multiple IEEE 802.11 b/g interfaces. The solution has been implemented and tested on Mikrotik Routerboard RB532 devices with OpenWRT Linux software installed. We assume that the mesh nodes and antennas are fixed. In such network the radio signal propagation changes mainly due to some interferences e.g. by neighbouring transmission in overlapping frequencies or by change in the physical signal propagation conditions e.g. by appearance of new obstacle on the link.

## 2. Triggering the self-healing action

The self-healing procedures are able to repair the mesh network. However before start of the repair the decision when start the self-healing action should take place must be made. In classic, manually managed mesh network both this decision and the repair action is performed manually by network operator. In the mesh networks the devices should trigger the automatic repair action when the network does not perform as good as it used to or as good as it should. The signal to trigger the action will come from the observation of the network performance. We propose that the self-healing actions are triggered in three cases:

- when failure of some of the network elements is observed,
- when the observed performance is lower than the typical for this network,
- when the observed performance is lower than preconfigured threshold.

The statistics representing the current performance of the mesh network needs to be compared not only to the preconfigured values, but also to values representing the observed average performance. The self-healing action will be triggered not only when the observed value is worse than the preconfigured threshold, but also when it is worse than the low-pass filtered value.

### 2.1 Detection of the link quality degradation

The goal is to use algorithms for detecting rapid drop in the wireless link quality. The detection is later used as a trigger for wireless mesh network management procedures to perform a self-healing action.

To detect the drop of link quality the constant monitoring of received signal strength must be performed. The monitoring module must collect the data reported by the network interface card driver, analyze the quality of the received

signal and apply some filtering or other methods of statistical analysis. The output of this module is a binary value – 1 if the quality of the link remains stable, 0 otherwise.

The Linux wireless network drivers use Received Signal Strength Index as a value representing the received radio signal strength (energy integral, not the quality). In an IEEE 802.11 system RSSI is the received signal strength in a wireless environment, in arbitrary units. RSSI can be used internally in a wireless networking card to determine when the amount of radio energy in the channel is below a certain threshold at which point the network card is clear to send (CTS). Once the card is clear to send, a packet of information can be sent. The end-user will likely observe an RSSI value when measuring the signal strength of a wireless network through the use of a wireless network monitoring tool like Network Stumbler.

In MadWiFi, the reported RSSI for each packet is actually equivalent to the Signal-to-Noise Ratio (SNR) and hence we can use the terms interchangeably. This does not necessarily hold for other drivers though. This is because the RSSI reported by the MadWiFi HAL is a value in dBm that specifies the difference between the signal level and noise level for each packet. Hence the driver calculates a packet's absolute signal level by adding the RSSI to the absolute noise level. In general, an RSSI of 10 or less represents a weak signal although the chips can often decode low bit-rate signals down to -94dBm. An RSSI of 20 or so is decent. An RSSI of 40 or more is very strong and will easily support both 54MBit/s and 108MBit/s operation.

The RSSI may fluctuate over time and very short interferences may cause a rapid change in RSSI value, which does not mean link breakage. To successfully detect the link loss some kind of low pass filter must be used, to ignore very short fluctuations of RSSI and do not trigger false detections. On the other hand, whenever the link quality goes down for a period of few seconds it is certainly the situation which should be detected.

We evaluated 3 simple methods to monitor the RSSI values and trigger the decision, whenever the link quality drop has appeared:

- threshold value on simple moving average,
- result of subtraction between short and long simple moving averages,
- a modified moving average.

The link quality degradation triggers the action, which is run as a procedure on the mesh devices which reconfigures the node according to some preconfigured parameters.

### 2.2.1 Simple moving average

This formula uses averaging window *aw*. This widow contains latest RSSI values from aw periods. Average from this value is compared with current value of RSSI. At time *k* anomaly is detected if the percentage of change of the current RSSI value exceeds threshold value *h*.

$$Ma\_alarm_k = Boolean\left( \frac{\frac{1}{aw}\sum_{i=k-aw}^{k-1} RSSI_i - RSSI_k}{\frac{1}{aw}\sum_{i=k-aw}^{k-1} RSSI_i} > h \right)$$

where

*aw* - size of the averaging window,

$RSSI_i$ - value of the RSSI at time equal to *i*

Simple moving average filters values which have random character in window *aw*. As a result we get average level of RSSI value. Comparing calculated moving average with current RSSI value we can get information about rapid RSSI changes. Unfortunately, when the decrease of RSSI value has an impulse character it may be misdetected by this algorithm. The size of the averaging window is a parameter of the algorithm and can be tuned to provide lower or higher cutoff frequency of the filter.

### 2.2.2 Result of subtraction between short and long simple moving averages

$$ML_k = \frac{1}{wl}\sum_{i=k-wl}^{k-1} RSSI_i$$

$$MS_k = \frac{1}{ws}\sum_{i=k-ws}^{k-1} RSSI_i$$

$$SL\_alarm_k = Boolean\left( \frac{ML_k - MS_k}{ML_k} > h \right)$$

This formula uses a two simple moving average. *ML* – is a long simple moving average. *MS* is a short simple moving average it uses much smaller averaging window. Anomaly is detected when *MS* is less than *h* percentage of *MS* value. The anomaly detecting formula reacts slower than simple moving average in case of RSSI value changes.

### 2.2.3 Modified moving average

We propose to use a modified moving average formula to simplify computing of the values. With these formulas there is no need to have cyclic buffer for latest value of RSSI. Only average if short and average of long modified moving average should be remembered.

$$WL_n = p_1 W_{n-1} + (1 - p_1) RSSI_k$$

$$WS_n = p_2 W_{n-1} + (1 - p_2) RSSI_k$$

$$p_1, p_2 \in (0,1)$$

$$W\_alarm_k = Boolean\left(\frac{WL_k - WS_k}{WL_k}\right)$$

## 2.3 Implementation of the monitoring module

The proposed monitoring solution was implemented in Linux on OpenWRT platform. It consist of userspace program that periodically reads the RSSI from the MadWIFI driver by IOCTL exchange and online or offline processing tool that analyzes the RSSI values and generates the signal of link loss. In case of online processing the values are passed directly to another process realizing the monitoring. For offline processing the RSSI are written to text file on the device filesystem. Both these programs were implemented in C.

## 3. The channel switching in response to failures

The self–healing of the link quality by channel switching algorithm may seem very simple: if the link quality degradation is detected try to switch to other channel, as probably the interferences there are lower than on current one. Unfortunately, the channel switching must be performed on all nodes using the same link simultaneously to sustain the connectivity. In wireless mesh network based on IEEE 802.11 radios a single link may join two or more wireless interfaces, using the same channel and ESSID. There are two possible solutions: centralized or distributed. In centralized solution a single point in the network reconfigures all the nodes using management protocol. In distributed algorithm information about the channel change may be triggered by any node and is gathered from the network state or transmitted by a protocol to all other nodes.

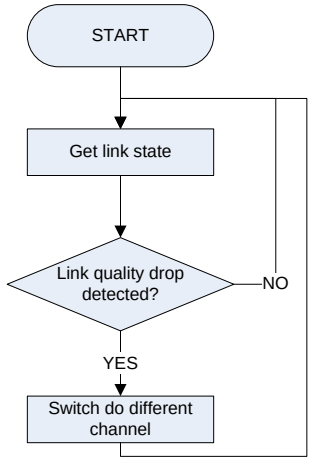



Fig. 1. Channel switching - general algorithm block diagram

### 2.1.1 Repair channel selection

In both the distributed and centralized solution the first step after detection of the link quality degradation is to select to which channel the link should be switched. This algorithm is run on the node that detects and triggers the change. To limit the time required to select the channel there is no additional scanning before the channel selection. The algorithm does not use the information about channel utilization reported by the interface driver (MadWIFI), because in the ad-hoc mode usually used in mesh networks these information is not reported correctly and for the infrastructure mode networks it requires background scanning which can be only performed on the client devices.

For the IEEE 802.11 b and g networks there are only 3 orthogonal channels available. The standard defines 13 channels each of width 22 MHz but spaced only 5 MHz apart. It is recommended that two networks on the same area should not use overlapping channels – the distance between them should be at least 5. However when the distance between the channels is greater than 3 the interferences are small. The algorithm takes, as a starting point, current channel and selects a channel which at least with distance of 3. The direction at which the channel is selected depends on the current channel and previous channel switching. When the current channel is lower than 7 the higher channels are selected, when it is greater or equal 7 higher one is chosen. If the channel change has already been started within some time in the past the channel is selected to be in the same direction as before.

## 4. Performance evaluation of the local channel switching algorithm

The tests of the channel switching algorithm have been performed in the Proximetry laboratory. Two Mikrotik routerboard RB532 devices were used as a source and sink for the traffic, third device was used to generate the interferences. The interferences were generated by sending as much packets as the MAC layer allowed on a channel next to the channel used for transmission. To flood the channel with the transmission the ACK messages on the MAC layer has been disabled. The *iperf* [4] program was used to generate the traffic. The source was configured to transmit 1MBit of data per second. The IEEE 802.11g frequencies were used. The data transmission started at channel 2, the interferences were generated at channel 1.

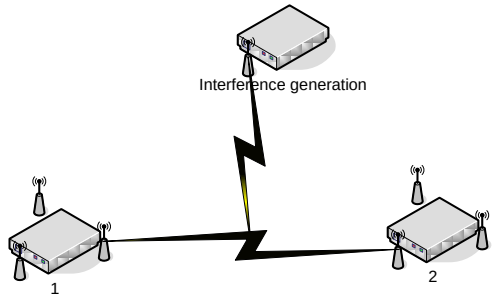



Fig. 2. Laboratory installation for evaluation of channel switching algorithm

As a first test a measurement of traffic without the channel switching has been done. The UDP protocol was used during the experiment. The interferences were generated starting from the 55s of the transmission. After start of the interference generation almost no packet could be received correctly.

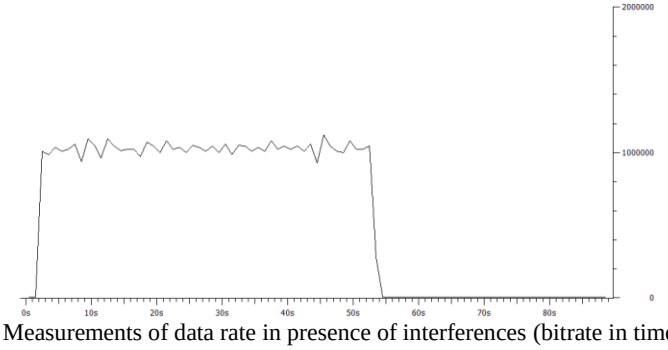



Fig. 3. Measurements of data rate in presence of interferences (bitrate in time)

The next test was executed in the same conditions, but the local channel switching algorithm was enabled. The interferences were generated starting from 52nd second of the transmission. The self-healing mechanism requires few seconds to detect the interference and start the channel switching procedure. As can be seen on the 0 the algorithm was able to restore the full throughput of the transmission within 6-7 seconds. The data transmission was switched to channel 7. Unfortunately during the experiment we have experienced also a false detection and the channel switch procedure was initiated at 62s of the transmission, which temporary brought back the transmission to frequencies that were subject to interferences. This situation was repaired within short time period.

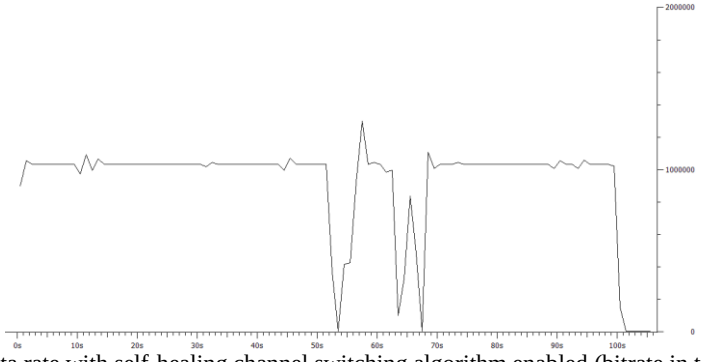



Fig. 3. Data rate with self-healing channel switching algorithm enabled (bitrate in time)

## 5. Summary

The presented channel switching algorithm provides quick and easy to implement function to automatically repair the connectivity between nodes in reaction to interferences. The sample implementation was presented, together with measurements evaluating the time required to switch the channel.

## 6. Acknowledgement

This work was supported in part by the European Commission in the 7th Framework Programme through project EU-MESH (Enhanced, Ubiquitous, and Dependable Broadband Access using MESH Networks), ICT-215320, www.eu-mesh.eu.

# References


[1] Ian F. Akyildiz, Xudong Wang, Weilin Wang, Wireless mesh networks: a survey, Computer Networks, Volume 47, Issue 4

[2] Vasilios Siris, Ioannis G. Askoxylakis, Marco Conti and Raffaele Bruno, "Enhanced, Ubiquitous and Dependable Broadband Access using MESH Networks". ERCIM Newsletter, Issue 73, pp 50-51, April 2008

[3] The EU-MESH project web site, http://www.eu-mesh.eu

[4] IPERF, http://www.noc.ucf.edu/Tools/Iperf/

[5] K. Grochla, W. Buga, P. Pacyna, J. Dzierżęga, A. Seman: "Autoconfiguration procedures for multi-radio wireless mesh networks based on DHCP protocol", IEEE Proceedings from HotMESH 09 Workshop, Kos, Greece 2009

[6] Prabhu Subramanian, Milind M. Buddhikot, Scott Miller: "Interference aware routing in multi-radio wireless mesh networks", Proc. of IEEE Workshop on Wireless Mesh Networks 2006

[7] Jian Tang and Guoliang Xue and Weiyi Zhang: "Interference-aware topology control and qos routing in multi-channel wireless mesh networks", Proceedings of ACM MOBIHOC 2005

[8] A. P. Subramaniana, H. Gupta, and S. Das, "Minimum-interference channel assignment in multi-radio wireless mesh networks," in IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07., 2007, pp. 481–490.

[9] K. N. Ramachandran, E. M. Belding, K. Almeroth, and M. Buddhikot, "Inteference-aware channel assignment in multi-radio wireless mesh networks," in IEEE Infocom, 2006, pp. 1–12.

HET-NETs 2010
ISBN 978-83-926054-4-7
pp. 429–442

# Modeling the IEEE 802.11 Networks MAC Layer Using Diffusion Approximation

TADEUSZ CZACHÓRSKI [a] KRZYSZTOF GROCHLA [a]

TOMASZ NYCZ [b] FERHAN PEKERGIN [c]

[a] Institute of Theoretical and Applied Informatics, Polish Academy of Science
ul. Bałtycka 5, Gliwice, Poland
*{tadek,kgrochla}@iitis.pl*

[b] Silesian University of Technology, Computer Center
44-100 Gliwice, Poland
*Tomasz.Nycz@polsl.pl*

[c] LIPN - CNRS UMR 7030, Université Paris-Nord
93 430 Villetaneuse, France
*pekergin@lipn.univ-paris13.fr*

**Abstract:** The article presents an analytical model of wireless networks using the IEEE 802.11 protocol to access the transport medium. The model allows to determine such key factors of the quality of service as transmission delays and losses. The model is based on diffusion approximation approach which was proposed three decades ago to model wired networks. We show that it can be adapted to take into consideration the input streams with general interarrival time distributions and servers with general service time distributions. The diffusion approximation has been chosen because of fairly general assumptions of models based on it, hard to be represented in Markov models. A queueing network model can have an arbitrary topology, the intensity of transmitted flows can be represented by non-Poisson (even self-similar) streams, the service times at nodes can be defined by general distributions. These assumptions are important: because of the CSMA/CA algorithm, the overall times needed to sent a packet are far from being exponentially distributed and therefore the flows between nodes are non-Poisson. Diffusion approximation allows us also to analyse the of transient behaviour of a network when traffic intensity is changing with time.

**Keywords:** Diffusion approximation; Queueing network models; Distributed coordination function; CSMA/CA; IEEE 802.11 protocol.

## 1. Introduction

The traffic transmitted by wireless networks has become more and more important, hence the performance and QoS issues of these networks should be carefully studied. The performance of IEEE 802.11 standard for wireless networks, its Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme with exponential backoff mechanism and its variants used to support asynchronous transfers, were thoroughly studied either analytically or by simulation e.g. in [15, 4, 2, 22, 26, 27]. The studies usually refer to the limit high traffic conditions. The relationships among throughput, blocking and collision probabilities are obtained, often with the use of a discrete-time Markov chain and then the performance of the backoff mechanism is studied.

Here, we propose a model which allows us to study not only the throughput of nodes using IEEE 802.11 standard, but also predicts the queue distributions at each node of the studied network, as well as waiting times distributions – hence the-end-to-end delays – and the loss probabilities due to the buffer overflows.

The model is based on the diffusion approximation which is a classical modelling method developed in 70-ties [12, 13] to study the performance of wired networks. The model is a typical queueing network one, where service stations represent nodes, service times represent the time needed to send a packet and queues at stations model the queues of packets at nodes. Once we obtain the queue distribution, we may also predict the waiting time distribution and the probability that the queue reaches its maximum value which approximates packet loss probability due to a saturated buffer.

The method can be used to model networks composed of a large number of nodes, e.g. mesh networks. It also allows the analysis of transient states occurring because of time-dependent traffic intensity or because of the changes in the network topology.

The main contribution of the article is a discussion how the CSMA/CA scheme with exponential backoff mechanism can be incorporated in this model and showing how the new model can be solved numerically.

The article is organised as follows. Section 2. recalls the principles of standard diffusion approximation model applied to a single station with limited queues with general independent distributions of interarrival and service times, i.e. the G/G/1/N queue. Both steady state and transient state solutions are given. The transient state model was proposed previously by the authors, [6, 7]. Section 3. presents the diffusion approximation model of a single node using CSMA/CA scheme with exponential backoff mechanism. Section 4. shows how the entire network of nodes with arbitrary topology can be solved. Some numerical examples are given.

## 2. Diffusion model of a G/G/1/N station

Let $A(x)$, $B(x)$ denote the interarrival and service time distributions at a service station. The distributions are general, but not specified, and the method requires only their first two moments. The means are $E[A] = 1/\lambda$, $E[B] = 1/\mu$ and variances are $\mathrm{Var}[A] = \sigma_A^2$, $\mathrm{Var}[B] = \sigma_B^2$. We also denote squared coefficients of variation as $C_A^2 = \sigma_A^2\lambda^2$ and $C_B^2 = \sigma_B^2\mu^2$. $N(t)$ represents the number of customers present in the system at time $t$.

As we assume that the interarrival times are independent and identically distributed random variables, hence, according to the central limit theorem, the number of customers arriving at the interval of length $t$ (sufficiently long to ensure a large number of arrivals) can be approximated by the normal distribution with mean $\lambda t$ and variance $\sigma_A^2\lambda^3 t$. Similarly, the number of customers served in this time is approximately normally distributed with mean $\mu t$ and variance $\sigma_B^2\mu^3 t$, provided that the server is busy all the time. Consequently, the changes of $N(t)$ within interval $[0, t]$, $N(t) - N(0)$, have approximately normal distribution with mean $(\lambda - \mu)t$ and variance $(\sigma_A^2\lambda^3 + \sigma_B^2\mu^3)t$.

Diffusion approximation [23, 24] replaces the process $N(t)$ by a continuous diffusion process $X(t)$. The incremental changes of $X(t)$, $dX(t) = X(t + dt) - X(t)$ are normally distributed with the mean $\beta dt$ and variance $\alpha dt$, where $\beta$, $\alpha$ are the coefficients of the diffusion equation

$$\frac{\partial f(x,t;x_0)}{\partial t} = \frac{\alpha}{2}\frac{\partial^2 f(x,t;x_0)}{\partial x^2} - \beta\frac{\partial f(x,t;x_0)}{\partial x} \tag{1}$$

which defines the conditional pdf of $X(t)$

$$f(x,t;x_0) = P[x \le X(t) < x + dx \mid X(0) = x_0].$$

Both processes $X(t)$ and $N(t)$ have normally distributed changes; the choice $\beta = \lambda - \mu$, $\alpha = \sigma_A^2\lambda^3 + \sigma_B^2\mu^3 = C_A^2\lambda + C_B^2\mu$ ensures the same ratio of time-growth of mean and variance of these distributions. The density of the diffusion process approximates the distribution of $N(t)$: $p(n,t;n_0) \approx f(n,t;n_0)$, and in steady state $p(n) \approx f(n)$.

More formal justification of diffusion approximation is in limit theorems for $G/G/1$ system given by Iglehart and Whitt [16, 17], but only for nonstationary processes.

The process $N(t)$ is never negative, hence $X(t)$ should be also restrained to $x \ge 0$. A simple solution is to put a *reflecting barrier* at $x = 0$ [19, 20].

The reflecting barrier excludes the stay at zero: the process is immediately reflected, therefore this version of diffusion with reflecting barrier is a heavy-load

approximation. This inconvenience can be removed by the introduction of another limit condition at $x = 0$: *a barrier with instantaneous (elementary) jumps* [12]. When the diffusion process comes to $x = 0$, it remains there for the time exponentially distributed with a parameter $\lambda_0$ and then it returns to $x = 1$. The time when the process is at $x = 0$ corresponds to the idle time of the system.

In the case of a queue limited to $N$ positions, the second barrier of the same type is placed at $x = N$. Coming to the barrier at $x = N$, the process stays there for a time corresponding to the period when the queue is full and incoming customers are lost and then, after the completion of the current service, the process jumps to $x = N - 1$. The model equations become [12]

$$\begin{aligned}
\frac{\partial f(x,t;x_0)}{\partial t} &= \frac{\alpha}{2}\frac{\partial^2 f(x,t;x_0)}{\partial x^2} - \beta\frac{\partial f(x,t;x_0)}{\partial x} + \\
&\quad + \lambda_0 p_0(t)\delta(x-1) + \lambda_N p_N(t)\delta(x-N+1)\ , \\
\frac{dp_0(t)}{dt} &= \lim_{x\to 0}\,[\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} - \beta f(x,t;x_0)] - \lambda_0 p_0(t)\ , \\
\frac{dp_N(t)}{dt} &= \lim_{x\to N}\,[-\frac{\alpha}{2}\frac{\partial f(x,t;x_0)}{\partial x} + \beta f(x,t;x_0)] - \lambda_N p_N(t)\ , \qquad (2)
\end{aligned}$$

where $\delta(x)$ is Dirac delta function.

The steady-state solution of the above equations is given in [12], and the transient-state solution in [6, 7].

This transient solution of $G/G/1/N$ model assumes that the parameters of this model are constant. If they are evolving, as it is in the case of time-dependent input rate, we should define the small time-periods where they can be considered constant and solve diffusion equation within these intervals separately. A transient solution obtained at the end of an interval serves as the initial condition for the next interval. We used this approach in several particular models, recently e.g. in [8, 9, 10] and we know it is stable numerically.

## 3. Diffusion model of a G/G/1/N station with CSMA/CA scheme and exponential backoff mechanism

In IEEE 802.11 networks the nodes use Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme in the MAC layer. A CSMA protocol works as follows: a station desiring to transmit senses the medium, if the medium is busy (i.e. some other station is transmitting) the station will defer its transmission to the later time, if the medium is sensed free then the station is allowed to transmit. The 802.11 standard extends this algorithm by positive acknowledge scheme – the receiver acknowledges the successful reception of a frame; the receipt of the ACK

will notify the transmitter that no collision occurred. When a collision occurs, the station waits for a random period of time. The exponential backoff algorithm is used to resolve contention: every time a station detects a collision or finds the channel busy, it will increase the top limit of random waiting time twice. In [27] a discrete-time Markov chain is given, describing this access policy and used to obtain the state probabilities, finally giving the station throughput in the case of heavy traffic. Below we consider the same sequence of events to define the random variable $T_o$ – the overall (total) time needed to transmit a packet. Its mean value and the coefficient of variation will represent the service time in the approximation diffusion model.

Let us denote:
$T_s$ — successful transmission time of one packet when a station gets the access to the channel and no collision occurs,
$T_c$ — time of a packet transmission failed due to a collision,
$W_i$ — the duration of $i$-th exponential backoff , $i = 1, 2, \ldots 6$, counted in time-slots, each of a fixed size $T_{slot}$
$N_i$ — the maximum number of steps to go through during $i$-th exponential backoff, $N_1 = 32, \ldots N_6 = 1024$,
$p_b$ — blocking probability, the probability that the channel is seen as occupied,
$p_c$ — collision probability, the probability that a started transmission fails.

$T_s$ and $T_c$ depend on the length of a packet, the speed of a link and the duration of additional operations defined by the standard, cf e.g. [27]. For example, if the packet length is 50, 500 or 1500 bytes with equal probability $1/3$, the link bit rate is $D = 54$ Mbit/s, the short interframe space SIFS = 10 $\mu s$, the distributed interframe space DIFS = 50 $\mu s$, and the slot time $T_{slot} = 20\mu s$ then $E[T_s] = 173\ \mu s$, $Var[T_s] = 8059\ (\mu s)^2$ giving $C^2_{T_s} = 0.269$. Similarily, $E[T_c] = 158\ \mu s$, $Var[T_c] = 8059\ \mu s^2$, $C^2_{T_c} = 0.323$.

During a backoff, at each time slot the availability of the channel is checked and with probability $1 - p_b$ – which is considered constant in this model – the number of time slots remaining to the end of the backoff is decreased by one or, with probability $p_b$, it does not change. Hence, the duration of a step is a geometrically distributed random variable $X$ with the rate $p_b$ having mean and variance

$$\mathrm{E}[X] = \frac{1}{1-p_b}, \qquad \mathrm{Var}[X] = \frac{p_b}{(1-p_b)^2}. \tag{3}$$

The duration $W_i$ of the whole $i$-th backoff which consists of $N$ steps, each of random duration $X$, where $N$ has the uniform distribution between 1 and $N_i$, as the sum of identically distributed independent random variables, is defined by a compound distribution. The Laplace transform $\bar{f}_{W_i}(s)$ of its probability density

function $f_{W_i}(x)$ has the form

$$\bar{f}_{W_i}(s) = \sum_{i=1}^{N_i} [\bar{f}_X(s)]^n p_i[n],$$

where $p_i(n) = 1/N_i$, and can be seen as the z-transform $N(z)$ of the distribution of random variable $N$, see e.g. [18, 11]. That allows us to determine the distribution of $W_i$ and, in particular, to determine its mean and variance as

$$\mathrm{E}[W_i] = \mathrm{E}[N_i]\mathrm{E}[X], \qquad \mathrm{Var}[W_i] = \mathrm{E}[N_i]\mathrm{Var}[X] + \mathrm{E}[X]^2\mathrm{Var}[N_i]^2.$$

We can also study the distribution of the backoff time using random walk formalism: the random walk starts at the point $x_0 = N$ and goes to the absorbing barrier at $x = 0$; the way to obtain the distribution of this time is given in [5]. We can even approximate this distribution using the diffusion process itself, using the first passage time density function $\gamma_{x_0,0}(t)$ introduced in the previous section

$$\gamma_{x_0,0}(t) \quad = \quad \lim_{x\to 0}[\frac{\alpha}{2}\frac{\partial}{\partial x}\phi(x,t;x_0) - \beta\phi(x,t;x_0)] = \frac{x_0}{\sqrt{2\Pi\alpha t^3}} e^{-\frac{(x_0-|\beta|t)^2}{2\alpha t}} \ .$$

Here, the diffusion process approximates the random walk from $x_0$ to 0 and its parameters $\alpha$, $\beta$ represent the mean and variance of this walk, hence they should be taken as

$$\beta = \frac{p_b - 1}{T_{slot}} \qquad \text{and} \qquad \alpha = \frac{1 - p_b}{T_{slot}} - \left(\frac{p_b - 1}{T_{slot}}\right)^2.$$

Then these distributions, computed for a fixed staring point $x_0$, are taken with the probabilities of the starting point.

The graph presented in Fig. 1 shows all possible sequences of events with random variables $T_s$, $T_c$, $W_i$ contributing to the duration of $T_o$ and their probabilities. The term $(1 - \varrho)$ says that two consecutive packets cannot be sent immediately one after another. Note that $T_o$ is the transmitter occupation time and not the time necessary for a succesful transmission because the case of a packet rejection is also taken into account. We may represent $T_o$ as a sum $T_o = a_1T_s + a_2T_c + \sum_{i=1}^{6} b_iW_i$ where the coefficients $a_i$, $b_i$ come from the graph and compute its mean value $\mathrm{E}[T_o] = a_1\mathrm{E}[T_s] + a_2\mathrm{E}[T_c] + \sum_{i=1}^{6} b_i\mathrm{E}[W_i]$ and variance $\mathrm{Var}[T_o] = \mathrm{E}[T_o^2] - \mathrm{E}^2[T_o]$.

The mean emission time is determined as the total mean time the link is occupied by the emission of one packet, $E[T_e] = E[T_s] + E[m_i]T_c$ where $E[m_i]$ is the mean number of collisions occurred, and the probability that the station is emitting is $p_e = E[T_e]/E[T_o]$. The probability that the channel is seen busy by a station reads $p_b = 1 - (1 - p_e)^{(k-1)}$, where $k$ is the number of stations competing to the

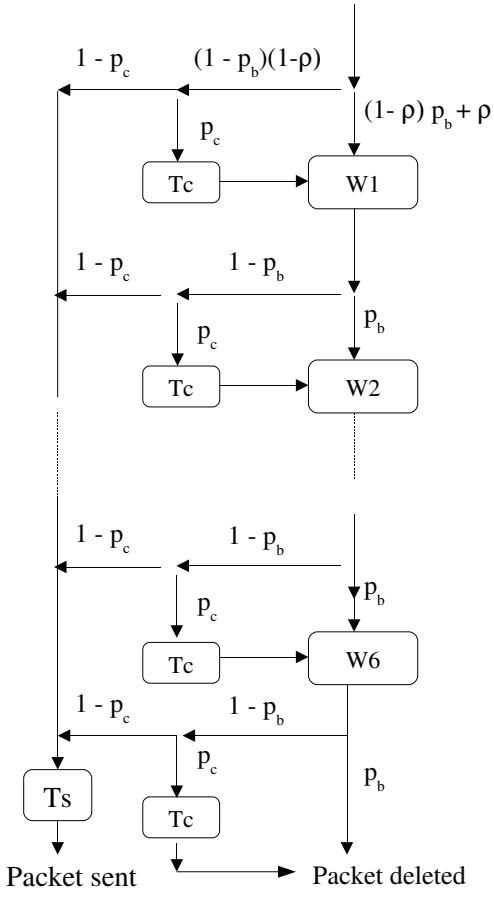



Fig. 1. Graph showing components of $T_o$.

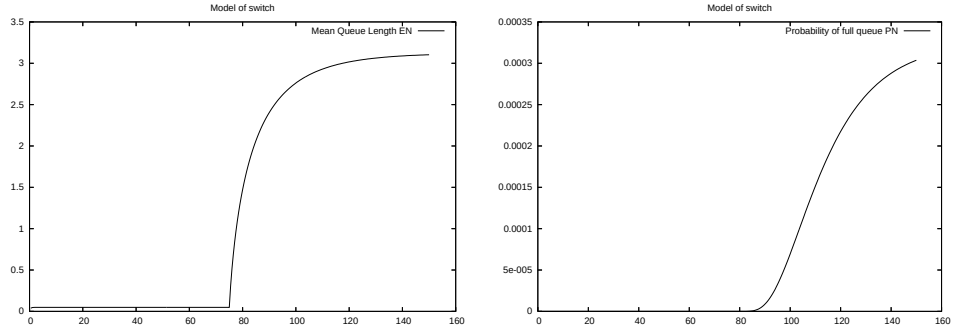



Fig. 2: left: The changes of the mean queue due to the changes of traffic intensity and following them the changes of blocking and collision probabilities, on horizontal axis time in milliseconds. right: Probability that the queue is saturated as a function of time in milliseconds.

channel, and the collision probability equals $p_c = p_b \cdot p_e$. We need an iterative procedure to determine these probabilities.

*Example 1*

Let us consider three nodes using the same channel and having identical parameters. The input stream intensity of each node is $\lambda = 200$ packets per second, $C_A^2 = 1$ (input stream is Poisson), $\mu = 1/T_o = 4904$ packets per second, $C_B^2 = 1.28$. These service time parameters correspond to previously computed $E[T_s] = 173$ $\mu$s with $C_{T_s}^2 = 0.269$ and $E[T_c] = 158$ $\mu$s, $C_{T_c}^2 = 0.323$ and $p_b = 0.105$, $p_c = 0.044$. At time $t = 75$ ms the transmitted flow rises at stations to $\lambda = 700$ packets/s, $C_A^2 = 1$, the blocking and collision probabilities change to $p_b = 0.399$, $p_c = 0.235$, and service parameters become $\mu = 1768$ packets/s, $C_B^2 = 8.574$. The maximum size of the queue is assumed 50.

The changes of the mean queue length and of the probability of queue saturation are displayed in Fig. 2. Fig. 3 displays steady state queue distributions given by the diffusion model in case of various traffic intensities $\lambda = 100 \ldots 1000$ packets per second.

Fig. 4 compares the queue length distributions in case of $\lambda = 300$ packets/s and $\lambda = 800$ packets/s, given by the diffusion model and by a detailed simulation model programmed in OMNET++ [25] and including all details of the CSMA/CA scheme and exponential backoff mechanism. We see that the distributions are very similar for a weak load and start to differ when the load is more significant. We suppose that one of the reasons of these divergences is the assumption of constant blocking probability $p_b$ in the model, while in fact it is a heavily autocorrelated variable.

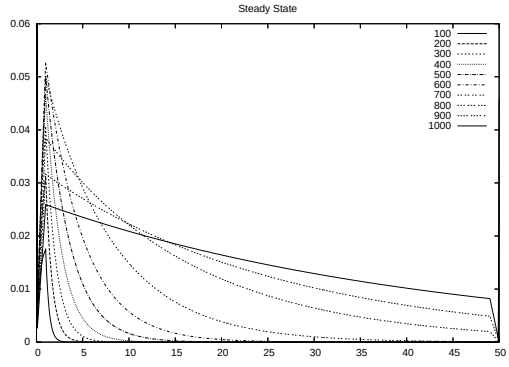



Fig. 3: Steady-state queue distributions for traffic intensity $\lambda = 100 \ldots 1000$ packets per second, $C_A^2 = 1$, and corresponding, resulting from blocking and collision probabilities, $\mu$ and $C_B^2$, for three identical stations using the same channel.

## 4. Open network of G/G/1/N queues

The diffusion steady state model of an open network of G/G/1 or G/G/1/N queues was presented in [13]. Below we present its short summary. Let $M$ be the number of stations, the throughput of station $i$ is, as usual, obtained from traffic equations

$$\lambda_i = \lambda_{0i} + \sum_{j=1}^{M} \lambda_j r_{ji}\,, \qquad i = 1, \ldots, M, \tag{4}$$

where $r_{ji}$ is the routing probability between station $j$ and station $i$; $\lambda_{0i}$ is external flow of customers coming from outside of the network.

The second moment of interarrival time distribution is obtained from two systems of equations; the first defines $C_{Di}^2$, the squared coefficient of variation of interdeparture times distribution at station $i$, as a function of $C_{Ai}^2$ and $C_{Bi}^2$; the second defines $C_{Aj}^2$ as another function of $C_{D1}^2, \ldots, C_{DM}^2$, see details in [13].

In case of a wireless network we should additionaly determine for each node $i$ the set $I_i = \{i_1, \ldots i_{n_i}\}$ of neighbouring stations influencing the transmission at station $i$, and compute iteratively the blocking and collision probabilities for each station; having the first moments of interarrival and service time distribution we can apply to each station the single station model from the previous section. In case of transient state models the whole task should be repeated at small time intervals and the solution of the network model (queues distributions, flow parameters) obtained at the end of an interval is used at the beginning of the next one.

*Example 2*

Let us consider 3 stations in line: traffic coming to the first station with intensity

$\lambda_1$ goes then to the second station and leaving it is directed to the third one. All stations communicate through the same channel and mutually interfere. We do not consider any other traffic, hence the traffic intensity at all stations is similar, we should only subtract the packets lost at each queue from the flow trespassing the stations. The buffer size admits up to 50 packets queued at each station.

The traffic intensity coming to the first station is changed in the similar way as in Example 1, i.e. at the first period $\lambda_1 = 200$ packets/s, $\mu = 4904$ packets/s, $C_A^2 = 1$, $C_B^2 = 1.28$, the service time parameters correspond to $p_b = 0.105$, $p_c = 0.044$. Then at $t = 75$ ms the transmitted flow rises to $\lambda_1 = 700$ packets/s, $C_A^2 = 1$, and $p_b = 0.399$, $p_c = 0.235$, hence the service parameters take values $\mu = 1768$ packets/s, $C_B^2 = 8.574$. In computations we neglect the time-dependent changes of the blocking and collision probabilities, although they might be computed for each time moment; here we assume that they change instantly at $t = 75$ ms.

Fig. 5 presents the dynamics of the changes of the mean queue length at three stations and the intensities of flows leaving each station and given by the diffusion model, compared to the changes of the input traffic at the first station.

*Example 3*

The network topology is the same as in the previous example. The nodes are initially empty. At $t = 0$ the traffic of $\lambda_1 = 500$ packets/s reaches the first station. We assume the same values of $E[T_s] = 173\mu s$, with $C_{T_s}^2 = 0.269$ and $E[T_c] = 158\mu s$, $C_{T_c}^2 = 0.323$ as previously. After a transition period the service at each station reaches the values $\mu_1 = 3666$, $\mu_2 = 2844$, $\mu_3 = 2320$, $C_{B_1}^2 = 2.4$, $C_{B_2}^2 = 7.767$, $C_{B_3}^2 = 15.76$, the channel blocking probability becomes $p_b = 0.271$, and collision probability seen by stations is $p_c = 0.101$. The dynamics of mean queue lengths and the changes of the squared coefficients of variation of interarrival times at each station are presented as the function of time in Fig. 6.

## 5. Conclusions

We believe that the diffusion approximation is one of most flexible and general approaches in queueing theory. Here we show how it can be used in modelling and performance analysis of wireless networks based on IEEE 802.11 protocol. In the article we emphasise the possibility of the analysis of transient states which is very rare in classical queueing models. The possibility to assume in the model the input streams with general interarrival time distributions and servers with general service time distributions gives us the means to represent easily way the whole Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme with exponential backoff mechanism inside the queueing model. The numerical results and some of their validations with detailed discrete event simulation indicate that

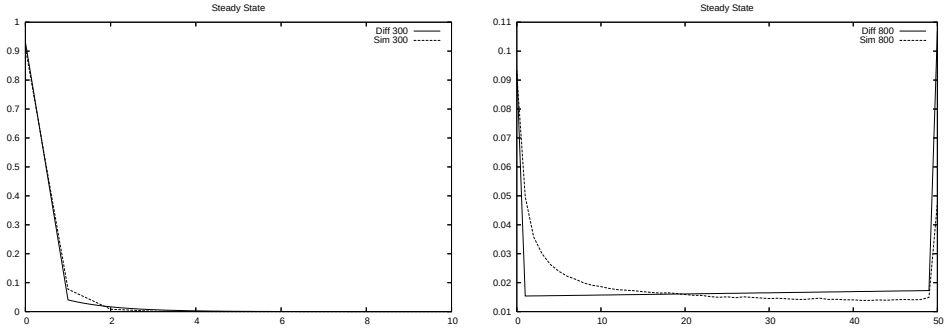



Fig. 4. Example 1: Steady-state queue distributions for traffic intensity left: $\lambda = 300$, right: $\lambda = 800$

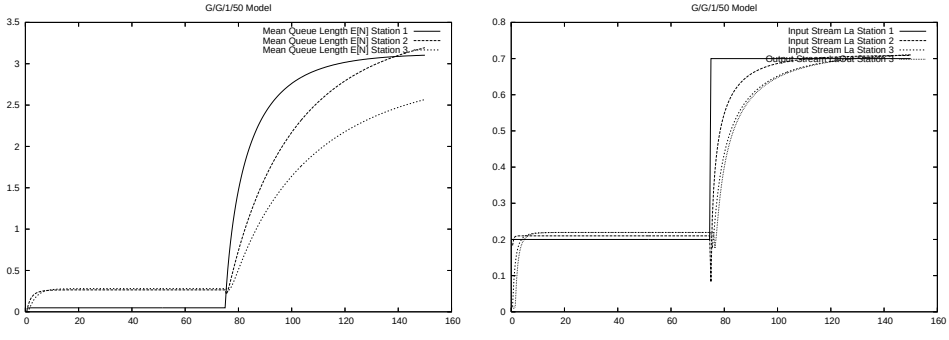



Fig. 5: Example 2: left: mean queues at stations as a function of time (in milliseconds), right: flows between stations as a function of time (in milliseconds)

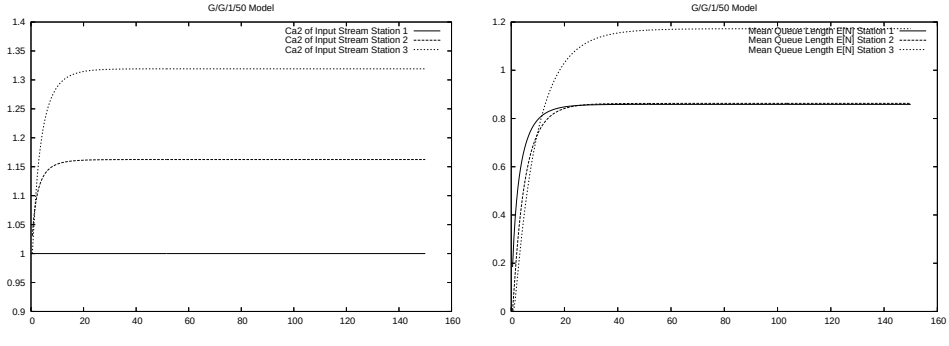



Fig. 6: Exaple 3: left: $C_{A_i}(t)$ for stations $i = 1, 2, 3$, time in milliseconds, right: Mean queue length for stations $i = 1, 2, 3$ changing with time, time in milliseconds.

the errors of the approach stay in reasonable range, although they are much larger than in the case of simple G/G1/N models. Therefore further analysis is needed.

## 6. Acknowledgements

This research was partially financed by the Polish Ministry of Science and Education grant N517 025 31/2997.

## References

[1] A. Banchs, A. Azcorra, C. García, R. Cuevas, Applications and Challenges of the 802.11e EDCA Mechanism: An Experimental Study, *IEEE Network*, July/August 2005, pp. 52-58.

[2] G. Bianchi, Performance Analysis of the IEEE 802.11 Distributed Coordination Function, *IEEE J. on Selected Areas in Communications*, vol. 18, no. 3, March 2000, pp. 535-547.

[3] P.J. Burke, The Output of a Queueing System, *Operations Research*, vol. 4, no. 6, pp. 699-704.

[4] G. R. Cantieni, Q. Ni, Ch. Barakat, T. Turletti, Performance analysis under finite load and improvements for multirate 802.11, *Computer Communications*, vol. 28 (2005) pp. 1095-1109.

[5] R. P. Cox, H. D. Miller, *The Theory of Stochastic Processes*, Chapman and Hall, London (1965).

[6] T. Czachórski, A Method to Solve Diffusion Equation with Instantaneous Return Processes Acting as Boundary Conditions, *Bulletin of Polish Academy of Sciences*, vol. 41 no. 4, 1993.

[7] T. Czachórski, J. M. Fourneau, F. Pekergin, Diffusion Model of an ATM Network Node, *Bulletin of Polish Academy of Sciences*, vol. 41 no. 4, 1993.

[8] T. Czachórski, J.-M. Fourneau, T. Nycz, F. Pekergin, Diffusion approximation model of multiserver stations with losses, *Electronic Notes in Theoretical Computer Science*, vol. 232, March 2009, pp. 125-143.

[9] T. Czachórski, T. Nycz, F. Pekergin, Transient states of priority queues – a diffusion approximation study, *Proc. of 2009 Fifth Advanced International Conference on Telecommunications* (AICT 2009), Venice, Italy, 24-28 May 2009.

[10] T. Czachórski, K. Grochla, F. Pekergin, Diffusion approximation model for the distribution of packet travel time at sensor networks, a chapter in a book: *Traffic and Performance Engineering for Heterogeneous Networks*, edited by D. Kouvatsos, River Publishers, 2009.

[11] W. Feller, *An introduction to probability theory and applications*, Vol. 1, pp. 287-301, Wiley, 3rd edition, 1968, New York.

[12] E. Gelenbe, On Approximate Computer Systems Models, *J. ACM*, vol. 22, no. 2, 1975.

[13] E. Gelenbe, G. Pujolle, The Behaviour of a Single Queue in a General Queueing Network, *Acta Informatica*, Vol. 7, Fasc. 2, pp.123-136, 1976.

[14] E. Gelenbe, Probabilistic models of computer systems. Part II, *Acta Informatica*, vol. 12, pp. 285-303, 1979.

[15] T-S. Ho, K.C. Chen, Performance Analysis of IEEE 802.11 CSMA/CA Medium Access Control Protocol, *Personal, Indoor and Mobile Radio Communications, 1996. Seventh IEEE International Symposium on* Volume 2, Issue , 15-18 Oct 1996 Page(s):407 - 411 vol.2pp. 407-411, 1996.

[16] D. Iglehart, W. Whitt, Multiple Channel Queues in Heavy Traffic, Part I-III, *Advances in Applied Probability*, vol. 2, pp. 150-177, 355-369, 1970.

[17] D. Iglehart, Weak Convergence in Queueing Theory, *Advances in Applied Probability*, vol. 5, pp. 570-594, 1973.

[18] L. Kleinrock, *Queueing Systems, vol. I: Theory, vol. II: Computer Applications* Wiley, New York 1975, 1976.

[19] H. Kobayashi, Application of the diffusion approximation to queueing networks, Part 1: Equilibrium queue distributions, J.ACM, vol. 21, no. 2, pp. 316-328, Part 2: Nonequilibrium distributions and applications to queueing modeling, *J.ACM*, vol. 21, no. 3, pp. 459-469, 1974.

[20] H. Kobayashi, *Modeling and Analysis: An Introduction to System Performance Evaluation Methodology*, Addison Wesley, Reading, Mass. 1978.

[21] H. Kobayashi, Q. Ren, A Diffusion Approximation Analysis of an ATM Statistical Multiplexer with Multiple Types of Traffic, Part I: Equilibrium State Solutions, *Proc. of IEEE International Conf. on Communications*, ICC '93, pp. 1047-1053, May 23-26, 1993, Geneva, Switzerland.

[22] D. Malone, K. Duffy, D. Leith, Modeling the 802.11 Distributed Coordination Function in Nonsaturated Heterogeneous Conditions, *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, February 2007, pp. 159-172.

[23] G. F. Newell, Queues with time-dependent rates, Part I: The transition through saturation; Part II: The maximum queue and return to equilibrium;, Part III: A mild rush hour, *J. Appl. Prob.* vol. 5, pp. 436-451, 579-590, 591-606. 1968.

[24] G. F. Newell, *Applications of Queueing Theory,* Chapman and Hall, London 1971.

[25] OMNET ++ site: http://www.omnetpp.org/

[26] L. Yun, L. Ke-ping, Z. Wei-Liang, W. Chong-Gang, Analyzing the Channel Access Delay of IEEE 802.11 DCF, *Globecom 2005*, pp. 2997-3001

[27] E. Ziouva, T. Antonakopoulos, CSMA/CA performance under high traffic conditions, throughput and delay analysis, *Computer Communications*, vol. 25, 2002, pp. 313-321.