JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

3/2013

Analysis of the System with Vacations Under Poissonian Input Stream and Constant Service Times
M. Sosnowski and W. Burakowski — *Paper* — *3*

Approximation of Message Inter-Arrival and Inter-Departure Time Distributions in IMS/NGN Architecture Using Phase-Type Distributions
S. Kaczmarek and M. Sac — *Paper* — *9*

Traffic Type Influence on Performance of OSPF QoS Routing
M. Czarkowski, S. Kaczmarek, and M. Wolff — *Paper* — *19*

Quality Aware Virtual Service Delivery System
M. Fraś and J. Kwiatkowski — *Paper* — *29*

Comparison of Resource Control Systems in Multi-layer Virtual Networks
B. Dabiński, D. Petrecki, and P. Świątek — *Paper* — *38*

Quality Management in 4G Wireless Networking Technology Allows to Attend High-Quality Users
M. Langer — *Paper* — *48*

ILP Modeling of Many-to-Many Replicated Multimedia Communication
K. Walkowiak, D. Bulira, and D. Careglio — *Paper* — *56*

Minimizing Cost of Network Upgrade for Overlay Multicast – Heuristic Approach
M. Szostak and K. Walkowiak — *Paper* — *66*

Performance Evaluation of the MSMPS Algorithm under Different Distribution Traffic
G. Danilewicz and M. Dziuba — *Paper* — *74*

(*Contents Continued on Back Cover*)

Editorial Board

Editor-in Chief: ..... ***Paweł Szczepański***

Associate Editors: ..... ***Krzysztof Borzycki***
Marek Jaworski

Managing Editor: ..... ***Robert Magdziak***

Technical Editor: ..... ***Ewa Kapuściarek***

Editorial Advisory Board

Chairman: ..... ***Andrzej Jajszczyk***
Marek Amanowicz
Daniel Bem
Wojciech Burakowski
Andrzej Dąbrowski
Andrzej Hildebrandt
Witold Hołubowicz
Andrzej Jakubowski
Marian Kowalewski
Andrzej Kowalski
Józef Lubacz
Tadeusz Łuba
Krzysztof Malinowski
Marian Marciniak
Józef Modelski
Ewa Orłowska
Andrzej Pach
Zdzisław Papir
Michał Pióro
Janusz Stokłosa
Andrzej P. Wierzbicki
Tadeusz Więckowski
Adam Wolisz
Józef Woźniak
Tadeusz A. Wysocki
Jan Zabrodzki
Andrzej Zieliński

ISSN 1509-4553 on-line: ISSN 1899-8852

© Copyright by National Institute of Telecommunications
Warsaw 2013

Circulation: 300 copies

Sowa – Druk na życzenie, www.sowadruk.pl, tel. 22 431-81-40

JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

Preface

The development of new network technologies which offer higher bandwidth as well as the deployment of mobile communications, modern services, multimedia, unlimited Internet access, etc. create numerous problems and questions, e.g., how to interwork existing services with Next Generation Internet, how to manage and optimize the transmission of growing traffic volumes, how to enhance network performance. In the panoply of tools which help to solve such problems one may find teletraffic theory, performance evaluation and modeling of communication protocols and devices by simulation and mathematical analysis. Mathematical and algorithmic tools are used to solve problems of assuring QoS/QoE, distribution of multimedia, modern routing and switching algorithms including broadcast and multicast traffic control and management, high speed networks, e.g., optical, mobile systems and mobility models, network virtualization. It is easy to conclude that the theoretical analysis of the issues associated with computer networks plays important role for any engineer involved in the development of new communications solutions.

The current issue of *JTIT* brings 15 articles on problems related to the wide range of the network mechanisms that may be included in future networks. What unites published materials, are the authors or co-authors coming from the leading Polish academic research centers specializing in telecommunications. Many of the subjects were inspired by their research inside Polish project Future Internet Engineering (IIP) led by Prof. Wojciech Burakowski and gathering nine Polish academic centers. Each of the articles represents an original look at the issues the authors are dealing with.

In the first paper, *Analysis of the System with Vacations under Poissonian Input Stream and Constant Service Times*, Maciej Sosnowski and Wojciech Burakowski, show how to use an abstract model with vacations, fed with Poisson type stream to dimensioning real system created within Future Internet Engineering project. In this place we encourage our readers to familiarize with results of the whole IIP project, starting from its web site, http://www.iip.net.pl.

Authors of next papers deal with an issue which is vital to current networks, namely Quality of Service, looking from different points of view. *Approximation of Message Inter-Arrival and Inter-Departure Time Distributions in IMS/NGN Architecture Using Phase-Type Distributions* by Sylwester Kaczmarek and Maciej Sac presents simulation results of QoS investi-

gation in Next Generation Network architecture. OSPF Routing in DiffServ architecture with advanced, self-similar traffic model was investigated in the next paper, *Traffic Type Influence on Performance of OSPF QoS Routing* (Michał Czarkowski, Sylwester Kaczmarek, and Maciej Wolff). Mariusz Fraś and Jan Kwiatkowski in the work *Quality Aware Virtual Service Delivery System* deal with QoS problems in Web-based systems based on SOA concept, and present original architecture of VSDS tool. The paper *Comparison of Resource Control Systems in Multi-layer Virtual Networks* by Bartłomiej Dabiński, Damian Petrecki, and Paweł Świątek make a step toward QoS-aware virtual networks and try to compare different methods and tools for QoS assurance and proposing their own solution.

QoS in wireless systems is harder to achieve than in wired ones, due to multiple factors influencing the transmition. Małgorzata Langer in the article entitled *Quality Management in 4G Wireless Networking Technology Allows to Attend High-Quality Users* discusses qualitative and quantitative indicators of telecommunication services, important for massive growth in traffic volume and in the number of connected devices is observed nowadays.

QoS is tightly coupled with problems of multimedia and content distribution, so we have a pleasure to present work of Krzysztof Walkowiak, Damian Bulira and Davide Careglio, *ILP Modeling of Many-to-Many Replicated Multimedia Communication*, where authors discuss optimization mechanisms dedicated for many-to-many communication, like on-line conferences and telepresence applications. Economical side of upgrading current infrastructure to modern overlay multicast is encountered in the paper *Minimizing Cost of Network Upgrade for Overlay Multicast – Heuristic Approach* by Maciej Szostak and Krzysztof Walkowiak. The authors try to show the way to minimize the costs of required upgrade.

Modeling is important tool for evaluation of new solution for present and future generation networks. Another group of articles shows how to use different modeling methods to uncover the properties of proposed solution. Grzegorz Danilewicz and Marcin Dziuba in *Performance Evaluation of the MSMPS Algorithm under Different Distribution Traffic* paper investigate with simulation scheduling algorithms, emphasizing advantages of their new MSMPS solution. The paper *Call and Connections Times in ASON/GMPLS Architecture* by Sylwester Kaczmarek, Magdalena Młynarczuk, and Paweł Zieńko is devoted to the simulation research on ASON/GMPLS architecture control plane functions. Markovian analysis can be sometimes used as an alternative to simulation modeling, and Maciej Sobieraj, Maciej Stasiak, Joanna Weissenberg, and Piotr Zwierzykowski in *Single Hysteresis Model for Limited-availability Group with BPP Traffic* paper present their results concerning radio interfaces with single hysteresis mechanism obtained with Markovian switching process.

Next part of the *JTIT* issue is devoted to the problems of wireless networks. Krzysztof Gierłowski in his work *Interworking and Cross-layer Service Discovery Extensions for IEEE 802.11s Wireless Mesh Standard* points at the mechanisms lacking in mesh version of popular Wi-Fi standard, 802.11s and proposes some enhancements. Jerzy Martyna uses game theory for Cognitive Radio algorithms in *Cooperative Games with Incomplete Information for Secondary Base Stations in Cognitive Radio Networks* paper, and in *Quasi-Offline Fair Scheduling in Third Generation Wireless Data Networks* the same author proposes new scheduling algorithm for 3G wireless networks demonstrating its advantages with numerical analysis.

The last paper focus on the performance of the generic part of a network – a single node. In the IIP project, virtualization of the network involves not only protocols and applications but the whole nodes which acting as virtual network nodes for different internets. Prototype implementation of virtual nodes was implemented with Xen, and the paper *Performance Tests of Xen-based Node for Future Internet IIP Initiative* by Grzegorz Rzym and Krzysztof Wajda presents the results of performance indices investigation of Xen-based node, used for designing the provisioning module for IIP system.

Tadeusz Czachórski
Mateusz Nowak
Guest Editors

Paper

Analysis of the System with Vacations under Poissonian Input Stream and Constant Service Times

Maciej Sosnowski[a,b] and Wojciech Burakowski[a,b]

[a] National Institute of Telecommunications, Warsaw, Poland
[b] Warsaw University of Technology, Warsaw, Poland

Abstract—In the paper approximate formulas for the mean waiting times and the buffer dimensioning in the system with vacations fed by the stream of Poissonian type with constant service times is shown. Furthermore, in the considered system the time intervals of the availability/not-availability of the service are constant and are run alternately according to the assumed cycle. More precisely, presented approach begin with derivation of the mean waiting times and, on the basis of this, the required buffer size for guaranteeing the losses less than predefined value is estimated. The accuracy of the presented analytical formulas is on a satisfactory level. The formulas were used for the System IIP dimensioning.

Keywords—approximation analysis, buffer dimensioning, mean waiting times, system with vacations.

1. Introduction

The paper studies the system with vacations fed by the stream of Poissonian type with constant service times. Furthermore, in the considered system the time intervals of the availability/not-availability of the service are constant and are run alternately according to the assumed cycle. The analysis of this system focuses on derivation of the analytical formulas to estimate the mean waiting times and next, on the basis thereof, to estimate required buffer size to satisfy assumed predefined level of losses.

The considered system well models a part of the IIP System [1] based on virtualized network infrastructure that corresponds to the organization of virtual links established for particular Parallel Internets (PIs). These Parallel Internets should work in isolation. For establishing separate virtual links delegated to particular PIs, access to a physical link by a cycle-based scheduler, as shown in Fig. 1 is managed.

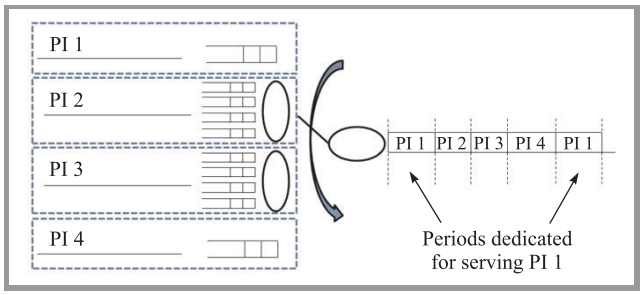


Fig. 1. Cycle-based scheduler for creating virtual links for 4 Parallel Internets working in isolation.

According to the best knowledge of the authors, such system was not analyzed in the literature. Most of papers, as i.e. [2], [3], [4], deal with TDMA systems, in which data are transmitted only in the chosen time-slots.

2. The System

2.1. The System with Vacations

The considered queuing system is depicted in Fig. 2. The system belongs to the family of systems with vacations. It means that periodically the system is in active and vacation periods. During the active periods (T_A) the packets are served while during the vacation periods (T_V) the service is not available. Moreover, we assume an infinite buffer size in the system. The queuing discipline is assumed to be FIFO.

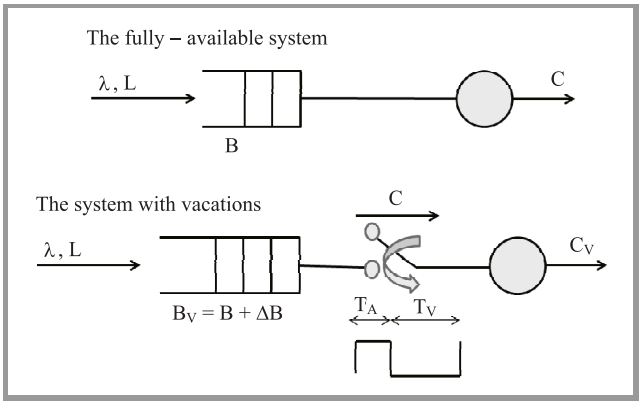


Fig. 2. Comparison of the systems: λ – arrival rate (Poissonian stream), L – packet size, B and B_V – buffers sizes, T_A – active period, T_V – vacation period, C and C_V – the output links rates.

Additional assumptions of the system are the following:

- the packets arrive to the system according to the Poisson process with the rate λ,
- the active (T_A) and vacation (T_V) periods are constant and they alternate,

– the link capacity is equal to C_V b/s,

– packet size (L) and, as a consequence, service times (h_v) of the packets are constant ($h_v = L/C_V$),

– the length of the active period T_A is multiples of h_v ($T_A = nh_v,\ n = 1,2,\dots$).

In this system, the available capacity for the considered stream, denoted by C, is:

$$C = \frac{C_V T_A}{T_A + T_V}. \tag{1}$$

2.2. Fully Available System

This analysis refers to the fully available system with Poissonian input and deterministic service time, the system $M/D/1$. Especially, the formula for mean waiting times in such system will be exploited when the arrival rate λ, packets size L, and output link C (see Eq. (1), equivalent to the available link rate in system with vacations) are the same for both considered system. It should be noted that the average load (in the system with vacation, during the active period) ρ in both systems is also the same. The difference between the system with and without vacations is the service time. In the fully available system the service time is $h = L/C$.

For the M/D/1, the mean waiting time $E[W_F]$ for well-known Pollachek-Khinchin formula is:

$$E[W_F] = \frac{\rho h_{res}}{1-\rho}, \tag{2}$$

where $\rho = \lambda h$ and h_{res} is the residual service time (in the case of $h = constant$, $h_{res} = h/2$).

3. Analysis

3.1. Mean Waiting Time in the System with Vacations

A brief description of the approach to calculate mean waiting times for the considered system with vacations can be found in [5].

The analysis starts from the moment of the test packet arrives to the system. Thanks to the PASTA (Poisson Arrivals See Time Averages) principle, this test packet sees the system at a random moment. This packet can arrive when the system is on the active period or on the vacation period. When the packet arrives during the vacation period it cannot be served immediately even if there are no other packets in the system, but it should wait for a transmission at least, if no other packets are in the system, by the remaining time of the period T_V. On the other hand, when the packet arrives during the active period it can be served immediately (when there are no other tasks in the system) when the remaining part of this period is not less than h_v. This period is called a pure active period. Let's define:

$$P_V = \frac{T_V}{T_V + T_A}, \quad P_{A'} = \frac{T_A - h}{T_V + T_A}, \quad P_{h_v} = \frac{h_v}{T_V + T_A}, \tag{3}$$

where P_V, $P_{A'}$, P_{h_v} denotes the probability that a packet arrives during the vacation period, the active period (without the last part equal h_v), and the last part (equal h_v) of the active period, respectively.

The approximate formula for the mean waiting time has the following form:

$$\begin{aligned} E[W_V] \;=\; & P_{A'} E[W_F] + P_V (T_{V_{res}} + E[W_F]) + \\ & + P_{h_v} (h_{v_{res}} + T_V + E[W_F]), \end{aligned} \tag{4}$$

where $E[W_F]$ is calculated by Eq. (2), $T_{V res}$ is the residual time of the vacation time ($T_{V res} = \frac{T_V}{2}$) and $h_{v_{res}}$ is the residual time of the service packet time ($h_{v_{res}} = \frac{h_v}{2}$).

Equation (4) can be simplified to:

$$E[W_V] = E[W_F] + \frac{(T_V + h_v)^2}{2(T_A + T_V)}. \tag{5}$$

For the limit case, when $T_V = 0$, the mean waiting time is the same as in the fully available system. On the other hand, when T_V tends to infinity, the value of the mean waiting time also tends to infinity.

Unfortunately, the Eq. (5) is not proved in a clear mathematical way, but it is only deduced. It was assumed that if the task arrives during the pure active period, it expects a similar delay as in the fully available reference system. It happens with probability $P_{A'}$. Furthermore, when the task arrives at the periods when it cannot be transmitted immediately (even when no other tasks are in the system), it should wait for its transmission when the active period starts. So, in this case, we deduce that the packet will wait by the time to the moment when the active period starts plus the service times of the packets being in the system already.

Equation (4) and (5) does not take into account the situation, e.g., when a task should wait a number of active periods until it starts transmission. Therefore, waiting times calculated from Eq. (5) will be lower than exact value.

Nevertheless, the Eq. (5) is relatively simple and it takes into account in a direct way the impact of the length of active and vacation periods on the packet delay.

In Tables 1 and 2, the values of the mean waiting times for the systems with vacations differing in lengths of active and vacation periods under different traffic load ρ and various T_A/T_V relations is presented.

It can be observed that for cases presented in Table 1, analytical results are very close to simulation results and the difference is only by few percent. For cases presented in Table 2, a bit less accuracy of the analytical results compared to the simulation can be observed, but the dif-

Table 1
Comparison of mean waiting times (short cycles)

	$T_A/T_V = 2h_v/4h_v$			$T_A/T_V = 10h_v/20h_v$		
ρ	Anal.	Sim.	Diff.	Anal.	Sim.	Diff.
0.2	2.5	2.4	3%	7.7	7.9	–2%
0.4	3.1	3.0	4%	8.4	8.6	–3%
0.6	4.3	4.2	3%	9.6	9.8	–2%
0.8	8.1	7.9	3%	13.4	13.3	0%
0.9	15.6	15.4	1%	20.9	20.6	1%
0.94	25.6	25.2	2%	30.9	30.6	1%
0.96	38.1	37.3	2%	43.4	43.5	0%

Table 2
Comparison of mean waiting times (long cycles)

	$T_A/T_V = 50h_v/100h_v$			$T_A/T_V = 100h_v/200h_v$		
ρ	Anal.	Sim.	Diff.	Anal.	Sim.	Diff.
0.2	34.4	36.5	–6%	67.7	72.2	–6%
0.4	35.0	39.4	–11%	68.3	77.8	–12%
0.6	36.3	42.7	–15%	69.6	84.4	–18%
0.8	40.0	47.5	–16%	73.3	92.3	–21%
0.9	47.5	54.6	–13%	80.8	100.1	–19%
0.94	57.5	63.9	–10%	90.8	110.0	–17%
0.96	70.0	76.1	–8%	103.3	122.2	–15%

ference is still on the satisfactory level. The difference is about 15% for most of the studied cases. The accuracy of Eq. (5) was also verified for other values of cycle durations and the results were similar to the ones presented above.

We can conclude that the Eq. (5) gives very accurate results for the system with vacations if at least one of these two conditions is met: cycle is short (∼15 h max.) or T_A/T_V ratio is small (∼1/4 max.). The results are also accurate if both conditions are close to these borders (i.e., $T_A/T_V = 10h_v/20h_v$).

3.2. Buffer Dimensioning

At present, an approximation for buffer dimensioning in the system with vacations is shown. The target is to dimension the buffer size as small as possible to assure that packet losses are less than a predefined value P_{loss}, e.g., $P_{loss} \leq 10^{-3}$. In order to do it in an exact way, the queue size distribution should be known. The presented approach assumes that only knowledge of the mean waiting times in the system, as calculated from Eq. (5) is available. Of course, the well-known Marcov's inequality can be used, see Eq. (6)), but it was checked that it gives an essential over-dimensioning of the buffer size and, as a consequence, the results are not reported in the paper.

Marcov's inequality:

$$P(X \geq n) \leq \frac{E[X]}{n}, \tag{6}$$

where $E[X]$ is the mean value of the random variable X.

Therefore, the approach investigated in the paper assumes an approximation of the queue size distribution, which is described by only one parameter. In this case, queue size distribution in the system with vacation can be approximated by a M/M/1 queue size distribution.

3.2.1. Queue State Distribution for the System with Vacations – Simulation Results

In this subsection the queue state distribution for selected system with vacations differing in the lengths of active and vacations periods is shown.

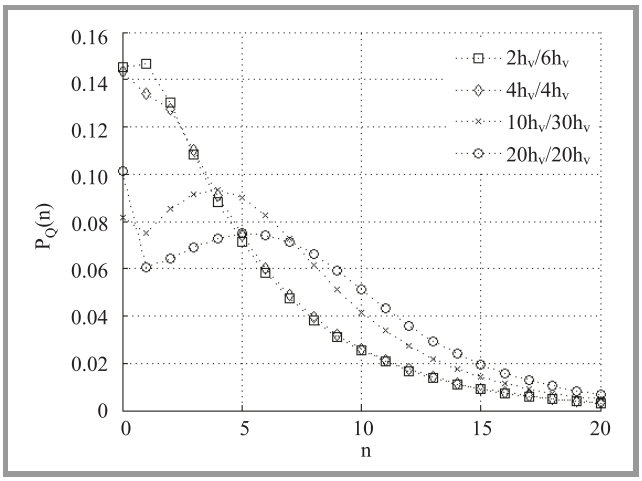


Fig. 3. Queue state distribution obtained from simulation ($n = 0 \ldots 20$), $\rho = 0.9$.

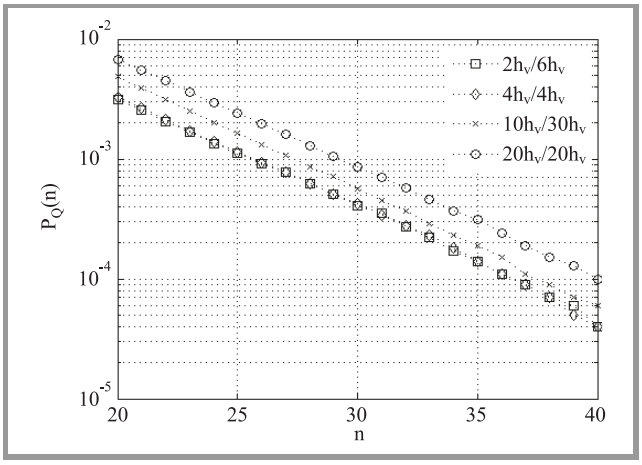


Fig. 4. Queue state distribution obtained from simulation ($n = 20 \ldots 40$), $\rho = 0.9$.

In Figs. 3 and 4 one can observe queue state distribution in the system with vacations for different values of T_A/T_V.

In the presented curves the confidence intervals at the 95% level are negligibly small and they are not depicted. The simulation results show that the approximation of the presented characteristics by the geometric distributions is justified although one can observe the differences in the first phase of the curves.

3.2.2. Approximation by the M/M/1 Queue Size Distribution

As it is known, the system state distribution follows the geometric distribution in the $M/M/1$ system. Queue is empty if 0 or 1 task in system, therefore the queue state distribution has the following form:

$$\begin{cases} P_Q(0) = P(0) + P(1) \\ P_Q(n) = P(n+1), \ n > 0 \end{cases} \tag{7}$$

where: $P(n) = \rho_g^n(1-\rho_g)$ – probability, that system is in the n state, ρ_g – server load.
Therefore,

$$\begin{cases} P_Q(0) = 1 - \rho_g^2 \\ P_Q(n) = \rho_g^{n+1}(1-\rho_g), \ n > 0 \end{cases} \tag{8}$$

So, the mean queue state in the $M/M/1$ system is:

$$E[n] = \sum_{n=0}^{\infty} nP_Q(n) = \frac{\rho_g^2}{1-\rho_g}. \tag{9}$$

Then ρ_g (the parameter of the M/M/1 queue state distribution) can be calculated from:

$$\rho_g = \frac{\sqrt{(E[n])^2 + 4E[n]} - E[n]}{2}, \tag{10}$$

where $E[n] = E[n_V] = \lambda E[W_V]$ and $E[W_V]$ is done by Eq. (5).

In Figs. 5–8 the comparisons between queue state characteristics obtained by the simulation and by approximation of the M/M/1 queue state distribution is shown. These results correspond with the exemplary system with vacations when $T_A/T_V = 10h_v/30h_v$ with the load of $\rho = 0.6$ and $\rho = 0.9$.

One can observe that the approximation by M/M/1 queue state distribution gives larger values for the tail of the distribution. It is important since we want to dimension buffer size for rather low values of losses, e.g., 10^{-3} or less.

After some algebra, the final formula for buffer size dimensioning, is

$$B = \left\lceil \frac{\ln(P_{loss})}{\ln(\rho_g)} - 1 \right\rceil, \tag{11}$$

where $\lceil x \rceil$ denotes the minimum integral value greater or equal x.

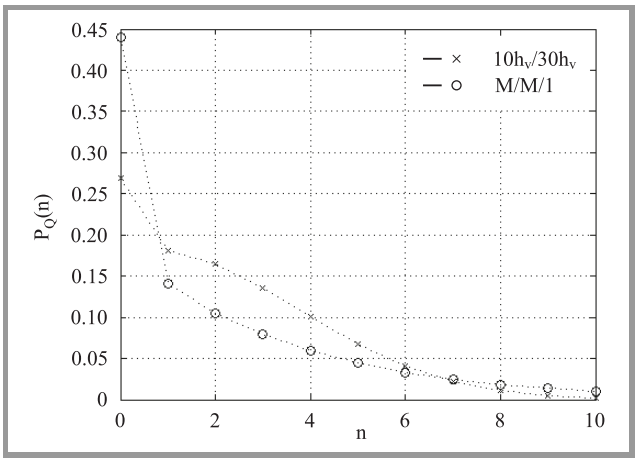


Fig. 5. Queue state distribution obtained from simulation compared with M/M/1 queue state distribution $(n = 0 \ldots 10)$, $\rho = 0.6$.

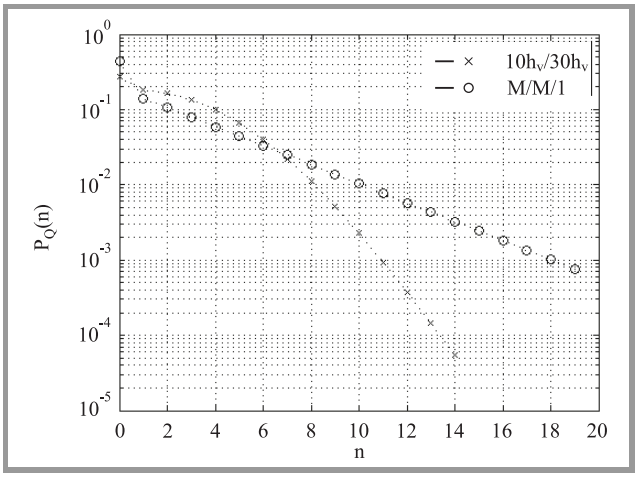


Fig. 6. Queue state distribution obtained from simulation compared with M/M/1 queue state distribution $(n = 0 \ldots 20)$, $\rho = 0.6$.

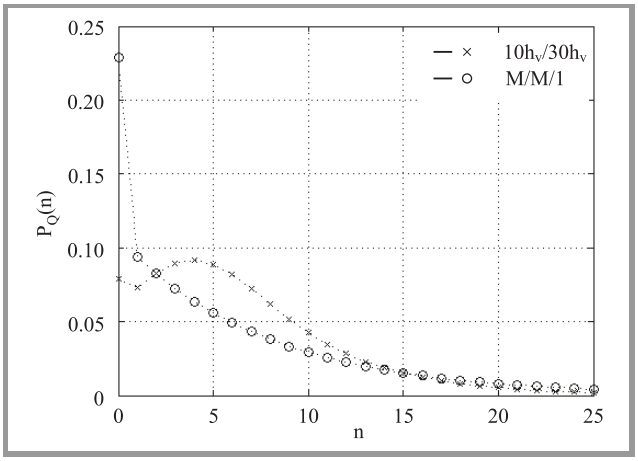


Fig. 7. Queue state distribution obtained from simulation compared with M/M/1 queue state distribution $(n = 0 \ldots 25)$, $\rho = 0.9$.

For comparison, the formula to dimension buffer size in the case of REM (Rate Envelope Multiplexing) multiplexing [6] is

$$\rho = \frac{2B}{2B - ln(P_{loss})}, \tag{12}$$

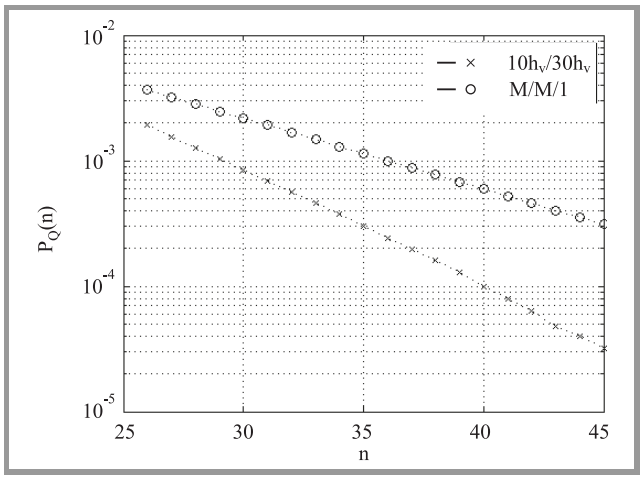


Fig. 8. Queue state distribution obtained from simulation compared with M/M/1 queue state distribution ($n = 26 \ldots 45$), $\rho = 0.9$.

and it can be transformed to

$$B = \left\lceil \frac{ln(P_{loss})}{2 - \frac{2}{\rho}} \right\rceil . \tag{13}$$

3.3. Results

In this section results for buffer dimensioning in the system with vacations REM multiplexing are presented. In Table 3 we show the results (B) of required buffer size for the system without vacations and REM multiplexing. The buffer size $B_{opt.}$ is obtained by simulation and $B_{over.}$ – indicates the relative error.

Table 3
Comparison of buffer size for the M/D/1 system and $P_{loss} = 10^{-3}$

ρ	$B_{opt.}$	B	$B_{over.}$ [%]
0.6	6	6	0
0.8	12	14	17
0.9	22	32	45
0.95	39	65	67

$B_{opt.}$ – the buffer size that provides the loss probability on the 10^{-3} level, result from the simulation
B – the buffer size calculated from Eq. (11)
$B_{over.}$ – percentage oversize of the buffer B

Table 4 presents the results of the required buffer size for the selected systems with vacations assuming $P_{loss} = 10^{-3}$. The reported results say that presented approach gives always the over estimation of the required buffer size. This overestimation is about 100%. Thus, the results are rather positive taking into account that the method is based on the approximation of the mean waiting time value only.

Table 4
Measured loss probability in the system with vacations

ρ	$E[n]$ sim.	$E[n]$ anal.	ρ_g	$B_{opt.}$	B	$B_{over.}$ [%]	P_{loss}
$T_A/T_V = 4h_v/4h_v$							
0.6	0.92	0.92	0.6	8	13	63	4.30E-06
0.8	2.24	2.22	0.75	14	24	71	7.90E-06
0.9	4.76	4.75	0.85	24	42	75	1.87E-05
0.95	9.78	9.78	0.91	40	73	83	3.03E-05
$T_A/T_V = 2h_v/6h_v$							
ρ	$E[n]$ sim.	$E[n]$ anal.	ρ_g	$B_{opt.}$	B	$B_{over.}$ [%]	P_{loss}
0.6	0.86	0.91	0.59	7	13	86	3.20E-06
0.8	2.12	2.21	0.74	13	22	69	1.63E-05
0.9	4.64	4.74	0.85	23	42	83	1.40E-05
0.95	9.27	9.75	0.91	39	73	87	2.24E-05
$T_A/T_V = 20h_v/20h_v$							
ρ	$E[n]$ sim.	$E[n]$ anal.	ρ_g	$B_{opt.}$	B	$B_{over.}$ [%]	P_{loss}
0.6	2.49	2.1	0.74	13	22	69	0.00E+00
0.8	4.38	3.8	0.82	19	34	79	6.00E-07
0.9	7.18	6.53	0.88	29	54	86	5.10E-06
0.95	12.35	11.64	0.93	45	95	111	5.20E-06
$T_A/T_V = 10h_v/30h_v$							
ρ	$E[n]$ sim.	$E[n]$ anal.	ρ_g	$B_{opt.}$	B	$B_{over.}$ [%]	P_{loss}
0.6	2.22	2.25	0.75	12	24	100	0.00E+00
0.8	3.87	4	0.83	17	37	118	0.00E+00
0.9	6.34	6.75	0.88	27	54	100	4.00E-06
0.95	12.06	11.88	0.93	43	95	121	3.40E-06

ρ_g – parameter of M/M/1 queue state distribution calculated from Eq. (10)
P_{loss} – measured loss probability for the buffer B (95% confidence intervals are on 10^{-7} level)

4. Summary

In the paper the analysis of the system with vacations fed by Poissonian stream was presented, constant service times and constant length of active and vacation periods. For this system the analytical approximate formulas for the mean waiting times and the buffer dimensioning was shown. The analytical results were compared with the simulation. The accuracy of the approximation is satisfactory.

The described methods were used to dimension virtual links in the IIP System build by the virtualization of the network infrastructure.

References

[1] W. Burakowski *et al.*, "Virtualized network infrastructure supporting co-existence of Parallel Internets", in *Proc. 13th ACIS Int. Conf. Softw. Engin., Artif. Intell., Netw. Parallel/Distrib. Comput. SNPD 2012*, Kyoto, Japan, 2012.

[2] S. S. Lam, "Delay analysis of a Time Division Multiple Access (TDMA) channel", *IEEE Trans. Commun.*, vol. 25, pp. 1489–1494, 1977.

[3] K. T. Ko and B. Davis, "Delay analysis for a TDMA channel with contiguous output and Poisson message arrival", *IEEE Trans. Commun.*, vol. COM-32, no. 6, pp. 707–709, 1984.

[4] I. Rubin, "Message delays in FDMA and TDMA communication channels", *IEEE Transactions Commun.*, vol. COM-27, pp. 769–777, 1979.

[5] M. Sosnowski and W. Burakowski, "Evaluation of mean waiting time in the system with vacations", Student Poster Session, *24th International Teletraffic Congress ITC 24*, Kraków, Poland, 2012.

[6] J. Roberts (Ed.), Information technologies and science: COST 224: *Performance evaluation and design of multiservice networks*, EUR 14152 en, 1992.

Maciej Sosnowski received his M.Sc. degree in Telecommunications from Warsaw University of Technology, Poland, in 2012. Since 2012 he has been Ph.D. student. He is a member of the TNT research group.

E-mail: M.Sosnowski3@itl.waw.pl
National Institute of Telecommunications
Szachowa st 1
04-894 Warsaw, Poland

E-mail: M.Sosnowski@tele.pw.edu.pl
Faculty of Electronics and Information Technology
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Wojciech Burakowski received his M.SC. Ph.D. and D.Sc. degrees in Telecommunications from Warsaw University of Technology in 1975, 1982 and 1992, respectively. Now he works as Full Professor at the Institute of Telecommunications, Warsaw University of Technology and at the National Institute of Telecommunications, Warsaw, as an R&D Director. He also leads the TNT research group (tnt.tele.pw.edu.pl). Since 1990 he has been involved in several COST and EU Framework Projects. He is a member of Telecommunication Section of the Polish Academy of Sciences and an expert in 7 FR Programme. He was a chairman and a member of many technical programme committees of national and international conferences. He is the author or co-author of about 200 papers published in books, international and national journals and conference proceedings and about 80 technical reports. He supervised 15 Ph.D. dissertations. His research areas include new networks techniques, ATM, IP, heterogeneous networks (fixed and wireless), network architecture, traffic engineering, simulation techniques, network mechanisms and algorithms, and recently Future Internet. Currently, he leads the strategic national project "Future Internet Engineering" (www.iip.net.pl).

E-mail: W.Burakowski@itl.waw.pl
National Institute of Telecommunications
Szachowa st 1
04-894 Warsaw, Poland

E-mail: wojtek@tele.pw.edu.pl
Faculty of Electronics and Information Technology
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Paper

Approximation of Message Inter-Arrival and Inter-Departure Time Distributions in IMS/NGN Architecture Using Phase-Type Distributions

Sylwester Kaczmarek and Maciej Sac

Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

Abstract—Currently it is assumed that requirements of the information society for delivering multimedia services will be satisfied by the Next Generation Network (NGN) architecture, which includes elements of the IP Multimedia Subsystem (IMS) solution. In order to guarantee Quality of Service (QoS), NGN has to be appropriately designed and dimensioned. Therefore, proper traffic models should be proposed and applied. This requires determination of queuing models adequate to message inter-arrival and inter-departure time distributions in the network. In the paper the above mentioned distributions in different points of a single domain of NGN are investigated, using a simulation model developed according to the latest standards and research. Relations between network parameters and obtained message inter-arrival as well as inter-departure time distributions are indicated. Moreover, possibility of approximating the above mentioned distributions using phase-type distributions is investigated, which can be helpful in identifying proper queuing models and constructing an analytical model suitable for NGN.

Keywords—call processing performance, IMS, message inter-arrival time distributions, message inter-departure time distributions, NGN, phase-type distributions, traffic model.

1. Introduction

Next Generation Network (NGN) [1] is a proposition for a standardized, packet-based telecommunication network architecture that delivers multimedia services with guaranteed quality. NGN consist of two stratums: transport stratum containing elements specific for particular transport technology and transport independent service stratum, which includes IP Multimedia Subsystem (IMS) [2] elements and uses mainly SIP (Session Initiation Protocol) [3] as well as Diameter [4] communication protocols.

In order to operate properly, both NGN stratums must be correctly designed. For this reason proposition and application of appropriate traffic models that evaluate among others call processing performance metrics [5], [6] are required. These metrics were formerly known as Grade of Service (GoS) parameters and include Call Set-up Delay (CSD) as well as Call Disengagement Delay (CDD).

As a result of the performed review [7], [8], the authors found out that standards organizations do not consider traffic engineering in their work and a majority of current research does not provide traffic models fully compatible with IMS-based NGN architecture (also abbreviated in the next part of the paper as IMS/NGN). Therefore, we decided to propose our own simulation [9] as well as analytical [10] model and use it to evaluate mean CSD and mean CDD in a single IMS/NGN domain.

During our investigations [10] some differences between analytical and simulation results were obtained, which are noticeable for high load. This leads to the research concerning message inter-arrival and inter-departure time distributions in a single domain of IMS/NGN performed using the simulation model. This would allow improving the precision of the analytical model by replacing simple M/G/1 queuing systems approximating the operation of servers and links with queuing models more adequate for the characteristics of IMS/NGN.

The results of this research were initially presented at a conference [11]. After that, they were supplemented by approximations of obtained message inter-arrival and inter-departure time histograms using phase-type distributions [12]–[16] and assessment of these approximations. All these aspects are presented in this paper, which is organized as follows. In Section 2 a description of the IMS/NGN network model and call scenario is provided. The simulation model used to measure message inter-arrival and inter-departure time distributions is presented along with the assumed measurements methodology. Histograms of message inter-arrival and inter-departure time obtained in different points of the network are described and commented in Section 3. Use of phase-type distributions to approximate the above mentioned histograms is examined in Section 4. Summary and future work are presented in Section 5.

2. Traffic Model of IMS/NGN

Network model [17], [18] and basic call scenario with two-stage resource reservation [18]–[21] in a single domain of ITU-T NGN architecture (the most advanced of all available NGN solutions [7], [8], [10]) are presented in Fig. 1 and Fig. 2 respectively. The model and the scenario are based

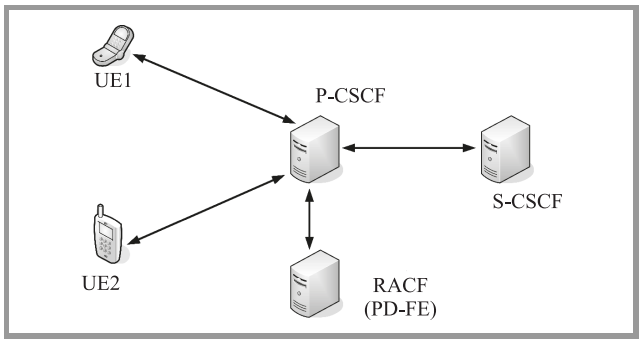


Fig. 1. Model of a single domain of IMS/NGN [17], [18].

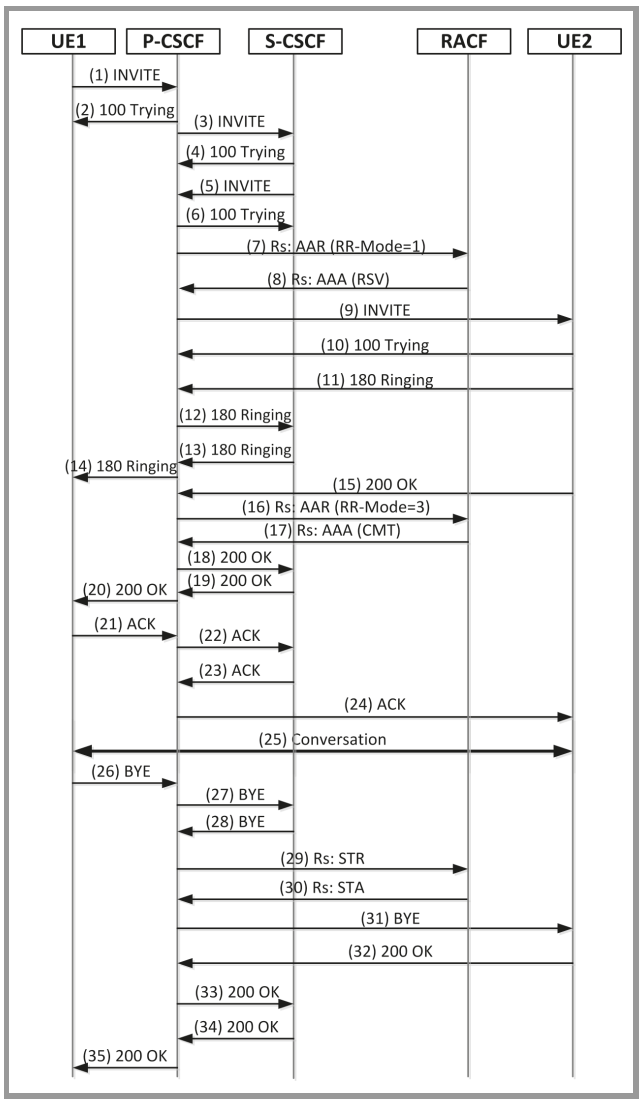


Fig. 2. Call set-up (messages 1–24) and call disengagement (messages 26–35) scenario in a single domain of IMS/NGN.

on the assumption that standard voice calls are requested by users and thus application servers are not used. Moreover, we assume that codec sets in User Equipments (UEs) are compatible and no audio announcements are played during the call.

Call set-up and disengagement requests are sent by UEs registered in the domain to the P-CSCF (Proxy – Call Session Control Function) server, which forwards them to the S-CSCF (Serving – Call Session Control Function) element, the main server handling all calls. The RACF (Resource and Admission Control Functions) unit representing the transport stratum allocates resources for a new call and releases resources associated with a disengaged call.

All elements of the network communicate using SIP protocol [3]. An exception is the communication of P-CSCF with RACF, for which Diameter protocol [4] is applied. Due to limited space available in the paper more detailed information about the network model and call scenario are not provided. This information can be found in [8], [10].

Based on the presented network model (Fig. 1) and assumed signaling messages exchange (Fig. 2), a simulation model of a single domain of IMS/NGN was proposed [9], which precisely implements the operation (algorithms) of all network elements as well as call set-up and disengagement scenarios. The aim of the model is to evaluate mean Call Set-up Delay E(CSD) and mean Call Disengagement Delay E(CDD) [5], [6]. Additionally, during simulations times of message arrival and departure at all network elements and links can be gathered. This allows calculation of histograms estimating message inter-arrival and inter-departure time distributions.

Logical structure of the simulation model is presented in Fig. 3. Details regarding the implementation of the model in the OMNeT++ [22] simulation environment are out of the scope of this paper and can be found in [9]. CSCF servers include Central Processing Units (CPUs), which are responsible for handling messages incoming from

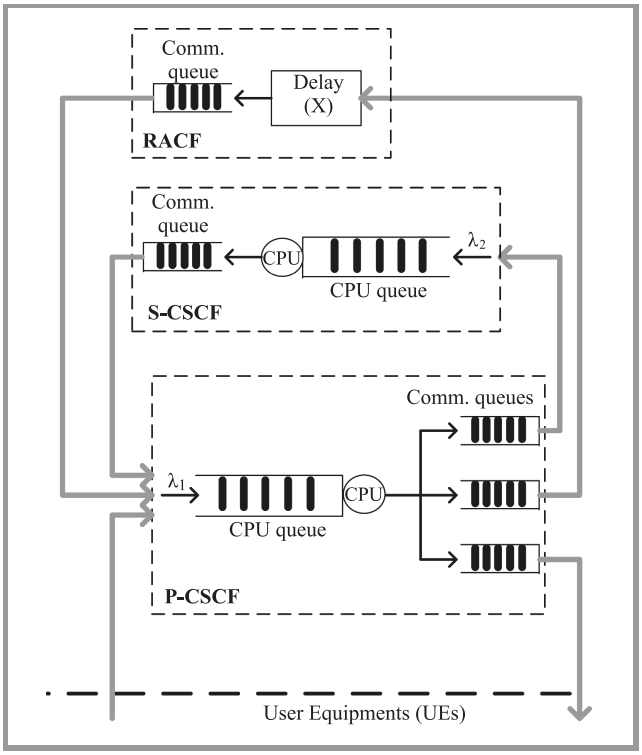


Fig. 3. Logical structure of the proposed traffic model.

other elements according to the assumed call scenario (Fig. 2). When CPUs are busy incoming messages are stored in CPU queues. Other network elements in the model respond with a particular delay (RACF responsible for resource allocation and release as well as UEs representing many user terminals). Each element of the model includes communication queues (one for each outgoing link), which buffer messages before sending them through busy links.

For proper operation of the model the following assumptions are taken [9]:

- message loss probability is negligible and thus there are no message retransmissions and each request is properly handled,
- UE1 generates aggregated call set-up requests (SIP INVITE messages) from many terminals with exponential intervals and given intensity, λ_{INV},
- call duration time is determined by an exponential distribution with definable mean (default 180 s),
- times of processing SIP INVITE messages by P-CSCF and S-CSCF are given by random variables T_{INV1} and T_{INV2} correspondingly,
- a_k factors (k = 1, 2, ... 8) determine times of processing other SIP and Diameter messages by CSCF servers

$$\begin{gathered} T_{TRi} = a_1 \cdot T_{INVi},\ T_{RINGi} = a_2 \cdot T_{INVi}, \\ T_{OKi} = a_3 \cdot T_{INVi},\ T_{ACKi} = a_4 \cdot T_{INVi}, \\ T_{BYEi} = a_5 \cdot T_{INVi},\ T_{OKBYEi} = a_6 \cdot T_{INVi}, \\ T_{AAAi} = a_7 \cdot T_{INVi},\ T_{STAi} = a_8 \cdot T_{INVi}, \\ i = 1 \text{ for P-CSCF, } 2 \text{ for S-CSCF}, \end{gathered} \tag{1}$$

- time of processing messages by RACF is described by a random variable, T_X,
- UE1 and UE2 represent many user terminals processing messages in nonzero time; SIP INVITE message processing time in UE1 and UE2 is described by a given distribution (default: uniform distribution with values from 1 to 5 ms); times of processing other messages are related to this time as in Eq. (1),
- communication times between elements of the network depend on definable lengths of optical links, bandwidth and lengths of transmitted messages,
- network elements communicate over dedicated interfaces; UE1 and UE2 represent many terminals connected to P-CSCF through a switch/router, communication times between the switches/routers and particular terminals are included in message processing times of UE1 and UE2,
- simulation is finished when the first of the following conditions occurs: total simulation time is exceeded, maximum number of generated calls is exceeded, confidence intervals for mean CSD and mean CDD are not greater than assumed maximum values.

The described simulation model was used to obtain histograms estimating message inter-arrival and inter-departure time distributions. For this reason message arrival and departure times were measured at the following points of all network elements (Fig. 4):

- CPU_i – the time of message arrival (the whole message, all bits are received) to the input of the P-CSCF or S-CSCF CPU queue,
- CPU_o – the time of message departure from the output of the P-CSCF or S-CSCF CPU,
- L_{ik} – the time of message arrival to the input of link communication queue for k-th link (if the link is not busy, then this is the time of sending the first bit of the message),
- L_{ok} – the time of sending the last bit of the message through k-th link.

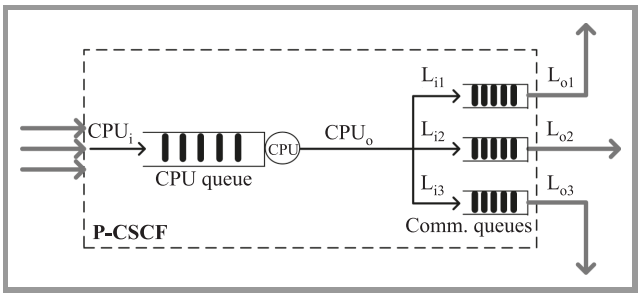


Fig. 4. Points of message arrival and departure times measurements based on the example of the P-CSCF server.

Acquired results were further processed in the MATLAB environment [23] to obtain histograms of message inter-arrival and inter-departure time at different points of the network. Additionally, some statistical values regarding the retrieved data were computed, including mean value and variance of message inter-arrival and inter-departure time. Also values of the c^2 variation coefficient were calculated, which is the ratio of variance of message inter-arrival or inter-departure time to squared mean value of this time.

3. Results

The simulation model described in Section 2 was used to investigate message inter-arrival time distributions at CPU_i and L_{ik} points as well as message inter-departure time distributions at CPU_o and L_{ok} points of all elements of the IMS/NGN domain (Figs. 1 and 4). During the investigations data sets presented in Table 1 and message lengths

presented in Table 2 were used. Additionally, the following assumptions were made:

- warm-up period 1500 s,
- 5 measurement periods,
- 0.95 confidence level,
- a_1= 0.2, a_2= 0.2, a_3= 0.6, a_4= 0.3, a_5= 0.6, a_6= 0.3, a_7= 0.6, a_8= 0.6,
- T_{INV1}, T_{INV2} and T_X input parameters are taken as constant values representing the maximum INVITE processing time by P-CSCF, the maximum INVITE processing time by S-CSCF, and the maximum message processing time by RACF respectively,
- simulation is finished when confidence intervals for E(CSD) and E(CDD) do not exceed 5% of mean Call Set-up Delay and mean Call Disengagement Delay or when total simulation time exceeds 10000 s; with such stop conditions at least 10000 message inter-arrival or inter-departure times were obtained during each simulation.

Table 1
Input data sets

Data set	λ_{INV} [1/s]	T_{INV1} [ms]	T_{INV2} [ms]	T_X [ms]	Links
1	100, 190, 220	0.5	0.5	3	No links
2	100, 190, 220	0.5	0.5	3	300 km 10 Mbit/s
3	100, 190, 220	0.5	0.5	3	300 km 100 Mbit/s

Table 2
Message lengths [24]

Message	Length [bytes]
SIP INVITE	930
SIP 100 Trying	450
SIP 180 Ringing	450
SIP 200 OK (answer to INVITE)	990
SIP ACK	630
SIP BYE	510
SIP 200 OK (answer to BYE)	500
Diameter messages	750

Although measurements in the simulation environment were performed at all available CPU_i, L_{ik}, CPU_o, L_{ok} points and for all data sets (Table 1), due to limited space we demonstrate only selected results. It is important to mention that under tested conditions offered loads to the P-CSCF CPU (ρ_1) and S-CSCF CPU (ρ_2) were as follows:

- ρ_1 = 0.41 and ρ_2 = 0.16 for λ_{INV} = 100 [1/s],
- ρ_1 = 0.78 and ρ_2 = 0.30 for λ_{INV} = 190 [1/s],
- ρ_1 = 0.90 and ρ_2 = 0.35 for λ_{INV} = 220 [1/s].

Performed research (Figs. 5–12) demonstrated that message inter-arrival and inter-departure time distributions in a single domain of IMS/NGN quite significantly differed

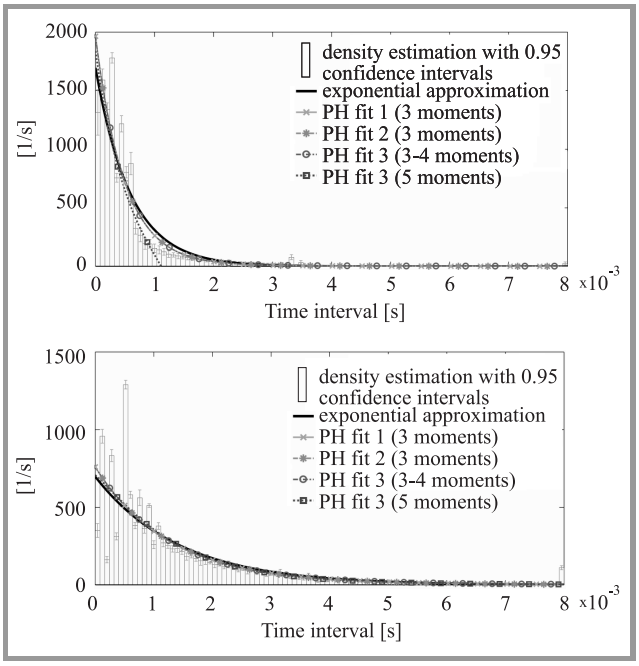


Fig. 5. Histograms of message inter-arrival times at the P-CSCF (top, c^2 = 2.19) and S-CSCF (bottom, c^2 = 1.33) CPU_i (data set 1, λ_{INV} = 100 [1/s]).

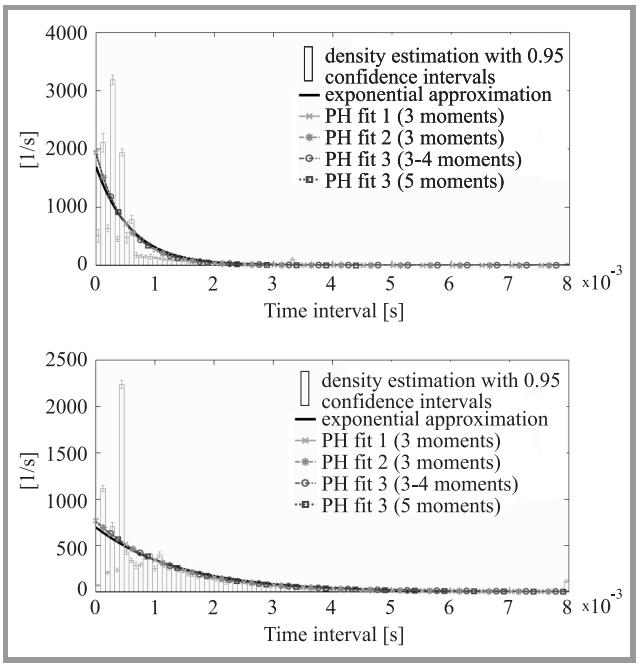


Fig. 6. Histograms of message inter-departure times at the P-CSCF (top, c^2 = 2.16) and S-CSCF (bottom, c^2 = 1.36) CPU_o (data set 1, λ_{INV} = 100 [1/s]).

from exponential distributions and depended on many parameters (in this section only conformity of the obtained histograms and their exponential approximations is discussed; all approximations will be considered and evaluated mathematically in the next section). This fact can be also confirmed by analyzing the c^2 coefficient, which for the most obtained data is far from the value 1.

Message inter-arrival and inter-departure time distributions are dependent among others on the offered load to CSCF servers CPUs. For low loads (low SIP INVITE message intensities λ_{INV}, Figs. 5–6) obtained histograms are to a certain extent similar to exponential distributions, especially at CPU$_i$ points (Fig. 5). Higher load results in more multimodal histograms with lower c^2 values, which is visible

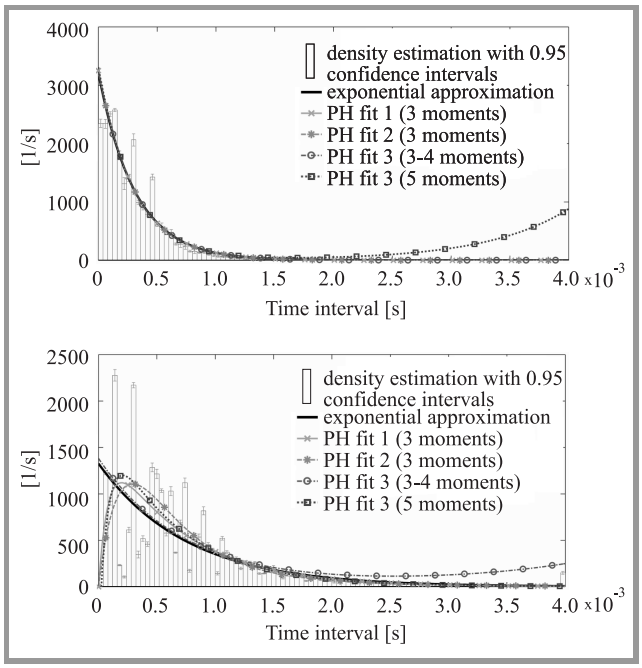


Fig. 7. Histograms of message inter-arrival times at the P-CSCF (top, c^2 = 1.12) and S-CSCF (bottom, c^2 = 0.87) CPU$_i$ (data set 1, λ_{INV} = 190 [1/s]).

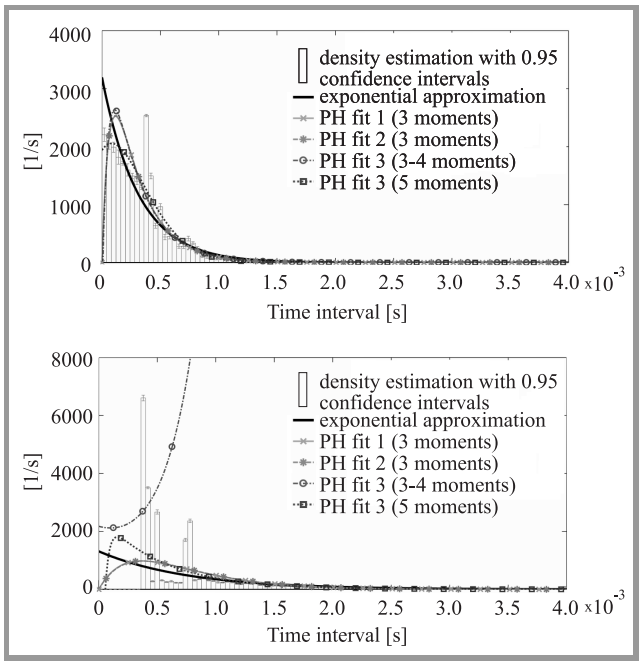


Fig. 8. Histograms of message inter-arrival times at the P-CSCF (top, c^2 = 0.69) and S-CSCF (bottom, c^2 = 0.59) CPU$_i$ (data set 2, λ_{INV} = 190 [1/s]).

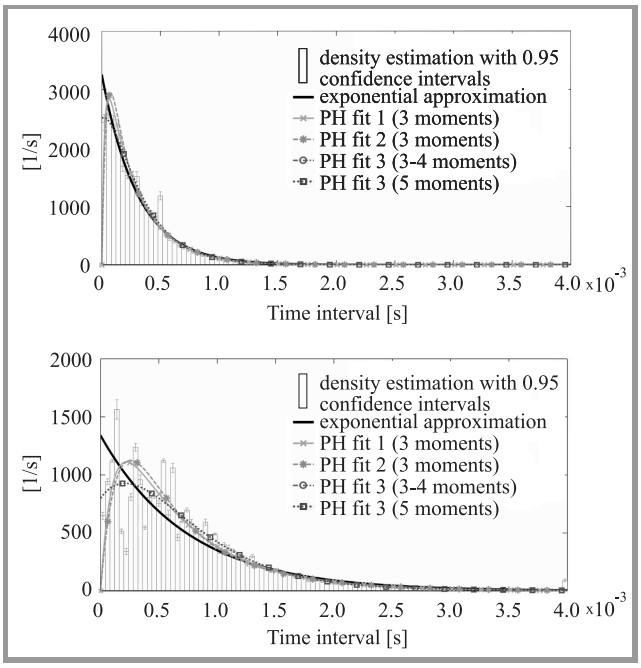


Fig. 9. Histograms of message inter-arrival times at the P-CSCF (top, c^2 = 0.90) and S-CSCF (bottom, c^2 = 0.81) CPU$_i$ (data set 3, λ_{INV} = 190 [1/s]).

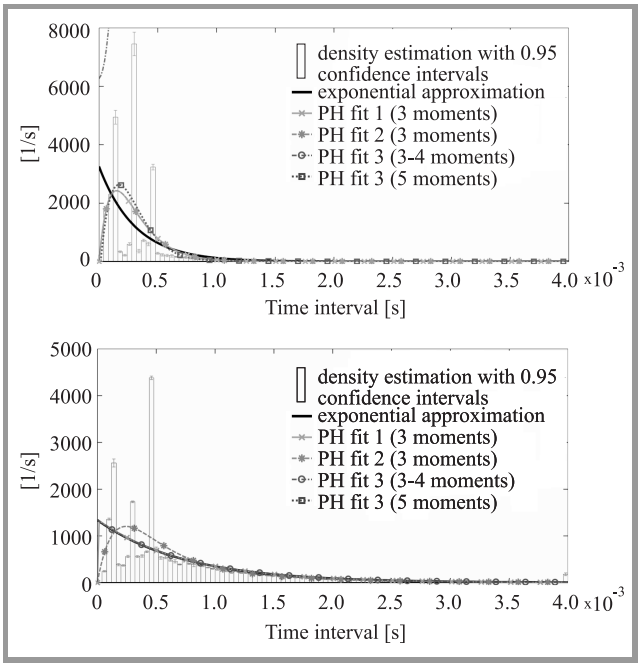


Fig. 10. Histograms of message inter-departure times at the P-CSCF (top, c^2 = 0.58) and S-CSCF (bottom, c^2 = 1.00) CPU$_o$ (data set 3, λ_{INV} = 190 [1/s]).

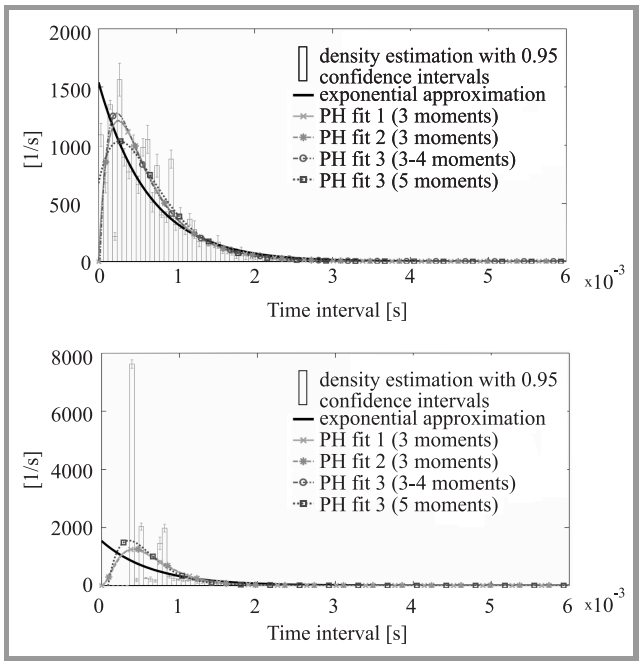


Fig. 11. Histogram of message inter-arrival times at the input of the P-CSCF→S-CSCF link (top, c^2 = 0.67) and histogram of message inter-departure times at the output of the P-CSCF→S-CSCF link (bottom, c^2 = 0.42) (data set 2, λ_{INV} = 220 [1/s]).

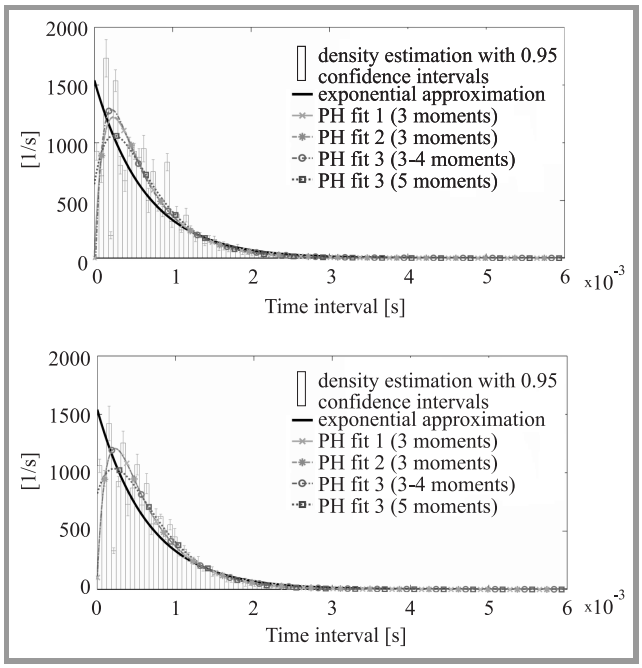


Fig. 12. Histogram of message inter-arrival times at the input of the P-CSCF→S-CSCF link (top, c^2 = 0.69) and histogram of message inter-departure times at the output of the P-CSCF→S-CSCF link (bottom, c^2 = 0.71) (data set 3, λ_{INV} = 220 [1/s]).

during analysis of the presented results (Figs. 7–10), particularly at CPU_o points (Fig. 10).

Such character of message inter-departure time distributions at the outputs of CSCF servers CPUs (Fig. 10) can be explained by the fact that under high load CPU queues are in most cases nonempty and one message is processed just after handling another. Therefore, message inter-departure times are usually very close to message processing times (Eq. (1)), which are a finite set of values. This results in multimodal message inter-departure time distributions.

Due to different set of messages processed by P-CSCF and S-CSCF these two servers have, however, slightly different inter-departure time distributions at CPU_o points (Fig. 10). Although in both elements these distributions are multimodal, for S-CSCF server there is in most cases one dominant peak at 0.5 ms and several smaller peaks, while for P-CSCF unit the position of the dominant peak is more dependent on the input parameters.

The investigated histograms are also influenced by the parameters of the links used to connect network elements, which involve application of communication queues (Fig. 3). This fact can be observed by comparing results presented in Fig. 7, obtained based on the assumption that all network elements are in one place and thus communication queues are not involved, to results demonstrated in Figs. 8–9, where links with particular length and bandwidth are used. In order to simplify simulations, it is assumed that all links have the same length and bandwidth.

The influence of communication links on message inter-arrival and inter-departure time distributions is dependent on the amount of offered load to the links, which is related to the available bandwidth. For low bandwidth (high offered load, Fig. 8) communication links may limit the minimal interval between the messages arriving to the CSCF server, which cannot be smaller than the time of sending the shortest message. This is also clearly visible in Fig. 11, where histograms at the input and output of the link between P-CSCF and S-CSCF are presented.

Links with relatively high available bandwidth (low offered load, Fig. 9 and 12) do not have a strong impact on message inter-arrival and inter-departure time distributions at CSCF servers. It can be noticed that for such links message inter-arrival time distributions at CPU_i points are slightly closer to exponential (less multimodal, Fig. 9), even comparing to the data set 1 (Fig. 7), where all network elements are connected directly to each other. It is important that the same experiments were performed for link bandwidths of 100 Mbit/s and 1 Gbit/s, however, no visible differences were noticed. Therefore, results obtained for 1 Gbit/s link bandwidth are not presented in the paper.

4. Approximations of Obtained Histograms Using Phase-Type Distributions

This section is dedicated to approximations of the message inter-arrival and inter-departure time histograms in a single domain of IMS/NGN using phase-type distributions [12]–[16] (Figs. 5–12). This term refers to the set of probability distributions that result from a system of one or more inter-related Poisson processes occurring in sequence,

or phases. Special cases of continuous phase-type distributions are [12]–[16], [25]:

- degenerate distribution (point mass at zero or the empty phase-type distribution) – 0 phases,
- exponential distribution - 1 phase,
- Erlang distribution – 2 or more identical phases in sequence,
- deterministic distribution (or constant) – the case of an Erlang distribution with infinite number of phases,
- Coxian distribution – 2 or more phases in sequence with a probability of reaching the terminating state after each phase,
- Hyperexponential distribution (also called a mixture of exponential) – 2 or more non-identical parallel phases, each of which has its own probability of occurring,
- Hypoexponential distribution – 2 or more (not necessarily identical) phases in sequence, a generalization of an Erlang distribution (in which phases are identical).

A very important feature of the set of phase-type distributions is that it is dense in the field of all positive-valued distributions [12]–[16], [25]. Therefore, phase-type distributions can represent or approximate (with any accuracy) any positive valued distribution.

Several algorithms for fitting different subsets of phase-type distributions to experimental data with respect to specified number of first moments have been proposed [12]–[16], [25]. The following algorithms are considered in this paper (Figs. 5–12):

- PH fit 1 [12], [26] – fitting acyclic Erlang-Coxian phase-type distributions with respect to 3 moments of experimental data,
- PH fit 2 [13], [27] – fitting minimal order acyclic phase-type distributions with respect to 3 moments of experimental data,
- PH fit 3 [14], [15], [27] – fitting phase-type distributions with respect to any number of moments of experimental data; in the paper we consider two cases: 3–4 moments (resultant phase-type distributions are the same for 3 and 4 moments) as well as 5 moments.

Apart from the above mentioned algorithms, we also approximated the obtained histograms using an exponential distribution (a special case of phase-type distributions) with λ parameter taken as the inverse of mean interval between messages. All calculations were performed in the MATLAB [23] environment.

Examples of fitting phase-type distributions to the obtained histograms are presented in Figs. 5–12. As can be observed, fitted distributions are much more smoother than the message inter-arrival and inter-departure time histograms. It is also very important that the PH fit 3 algorithm is sensitive to the input data (moments of intervals between messages) and sometimes produces results which are not acceptable (i.e., PH fit 3 with respect to 3–4 moments for S-CSCF CPU_i in Fig. 8). Such results were discarded during our research. This problem does not occur for other approximations (PH fit 1, PH fit 2, exponential).

For all histograms of intervals between messages at inputs and outputs of all network elements (Fig. 1 and 4) we applied phase-type distributions fitting, which was described earlier in this section. A more extensive set of call set-up request intensities was considered comparing to Table 1 (9 values ranging from 20 to 225 [1/s]), which resulted in 2700 fitted distributions. As mentioned before, some number of the PH fit 3 results were unacceptable and had to be rejected. For all accepted results chi-square tests [28] were performed, in order to assess goodness of fitting phase-type distributions to the obtained histograms. The following test parameters were assumed:

- message inter-arrival or inter-departure times were divided into 50 bins; bins were concatenated when necessary, to guarantee theoretical (expected) frequency not less than 5,
- significance level 0.05,
- number of degrees of freedom included the number of parameters determined for each fitted phase-type distribution.

The performed chi-square tests indicated that in the majority of cases histograms and fitted phase-type distributions significantly differ from each other (with respect to the assumed significance level). Therefore, detailed test results are not presented in the paper. Hypotheses that histograms follow particular phase-type distributions were not rejected only for some distributions at very low call set-up request intensities of 20 [1/s] and only at some measurement points in the network (inputs and outputs of the P-CSCF→RACF, P-CSCF→UE2 and RACF→P-CSCF links). Only at the input and output of the UE1→P-CSCF link there was a good conformity of histograms and fitted distributions for the range of 20–160 call set-up requests per second.

In order to check which of the applied approximations is the closest to the histograms obtained in particular measurement points, the following quality measure was assumed: mean difference between chi-square test statistics and critical chi-square values for the calculated number of degrees of freedom and assumed significance level. The averaging included all results for a particular approximation and measurement point (all call set-up request intensities and data sets).

Results of these investigations are presented in Table 3. It can be noticed that generally the best results are given by the PH fit 3 algorithm (variant with 5 moments fitted).

Table 3
Quality of tested approximations (numbers of moments for particular approximartions are given in brackets)

Point of measurement	Best approximation	...	...	...	Worst approximation
P-CSCF CPU_i	PH fit 1 (3)	PH fit 3 (3–4)	PH fit 2 (3)	Exponential	PH fit 3 (5)
P-CSCF CPU_o	PH fit 2 (3)	PH fit 1 (3)	PH fit 3 (5)	Exponential	PH fit 3 (3–4)
P-CSCF→RACF link in	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (5)	PH fit 3 (3–4)
P-CSCF→RACF link out	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (5)	PH fit 3 (3–4)
P-CSCF→S-CSCF link in	PH fit 1 (3)	PH fit 3 (5)	PH fit 2 (3)	Exponential	PH fit 3 (3–4)
P-CSCF→S-CSCF link out	PH fit 3 (5)	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (3–4)
P-CSCF→UE1 link in	PH fit 3 (5)	PH fit 1 (3)	PH fit 2 (3)	PH fit 3 (3–4)	Exponential
P-CSCF→UE1 link out	PH fit 3 (3–4)	PH fit 3 (5)	PH fit 2 (3)	PH fit 1 (3)	Exponential
P-CSCF→UE2 link in	PH fit 1 (3)	PH fit 2 (3)	Exponential	PH fit 3 (5)	PH fit 3 (3–4)
P-CSCF→UE2 link out	PH fit 1 (3)	PH fit 2 (3)	Exponential	PH fit 3 (3–4)	PH fit 3 (5)
RACF→P-CSCF link in	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (5)	PH fit 3 (3–4)
RACF→P-CSCF link out	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (5)	PH fit 3 (3–4)
S-CSCF CPU_i	PH fit 3 (5)	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (3–4)
S-CSCF CPU_o	PH fit 3 (5)	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (3–4)
S-CSCF→P-CSCF link in	PH fit 3 (5)	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (3–4)
S-CSCF→P-CSCF link out	PH fit 3 (5)	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (3–4)
UE1→P-CSCF link in	PH fit 1 (3)	Exponential	PH fit 2 (3)	PH fit 3 (3–4)	PH fit 3 (5)
UE1→P-CSCF link out	PH fit 2 (3)	PH fit 1 (3)	Exponential	PH fit 3 (5)	PH fit 3 (3–4)
UE2→P-CSCF link in	PH fit 3 (5)	Exponential	PH fit 1 (3)	PH fit 2 (3)	PH fit 3 (3–4)
UE2→P-CSCF link out	PH fit 3 (5)	PH fit 1 (3)	Exponential	PH fit 2 (3)	PH fit 3 (3–4)

Only slightly worse are the PH fit 1 and PH fit 2 algorithms, which operate on 3 moments of experimental data. Exponential approximations of the obtained histograms offer even poorer quality, which is, however, better than that of the PH fit 3 algorithm (3 moments).

5. Conclusions and Future Work

The aim of the work presented in the paper was to investigate message inter-arrival and inter-departure time distributions in a single domain of IMS/NGN architecture. The investigations also concerned a possibility of approximation of the above mentioned distributions using phase-type distributions. For these reasons the developed simulation model, which conforms to the latest standards and research, was used to gather data in different points of the network. Obtained results indicate that message inter-arrival and inter-departure time distributions in the system are influenced by many parameters (including offered load to CSCF servers and links) and are not exponential. This is especially visible for higher loads, for which histograms of time intervals between messages are multimodal, particularly at the output of CSCF servers CPUs.

To all obtained inter-arrival and inter-departure time histograms several phase-type distributions were fitted using available algorithms. Performed chi-square tests demonstrated that the acquired histograms in most cases do not follow the fitted distributions. Therefore, a metric, to examine which distributions are the most similar to the histograms was proposed. The best phase-type approximations are generally PH fit 3, based on 5 moments of intervals between messages. Only slightly worse results can be achieved by applying the PH fit 1 and PH fit 2 algorithms using 3 moments of experimental data.

The research described in this paper demonstrates that the problem of constructing an accurate analytical model of IMS/NGN is very complicated. According to the presented results, the obtained message inter-arrival as well as inter-departure time distributions and their exponential approximations generally significantly differ from each other. However, our experience indicates that in many cases using simple M/G/1 queues (with exponential inter-arrival times) to describe the operation of IMS/NGN servers and links gives satisfactory results when we consider the response of the whole system. Call processing performance (mean Call Set-up Delay as well as mean Call Disengagement Delay) results achieved with M/G/1 queues are very close to simulations and are comparable to the results obtained using commonly known G/G/1 approximations [10], [29].

Taking all mentioned facts into consideration, the authors are going to continue work on determination of proper queuing models for CSCF servers CPUs and optical links, in order to achieve better conformity of calculations and simulations for the whole investigated IMS/NGN architec-

ture. Apart from that, development of the traffic model is being planned in order to carry out research in a multi-domain IMS/NGN architecture, including also the elements specific for MPLS, Ethernet and FSA transport technologies [30]–[32].

Acknowledgements

This research work was partially supported by the system project "InnoDoktorant – Scholarships for PhD students, Vth edition" co-financed by the European Union in the frame of the European Social Fund.

References

[1] "General overview of NGN", ITU-T Rec. Y.2001, Dec. 2004.

[2] "IP Multimedia Subsystem (IMS); Stage 2 (Release 11)", 3GPP TS 23.228 v11.0.0, Mar. 2011.

[3] J. Rosenberg *et al.*, "SIP: Session Initiation Protocol", IETF RFC 3261, Jun. 2002.

[4] P. Calhoun *et al.*, "Diameter Base Protocol", IETF RFC 3588, Sept. 2003.

[5] "Call processing performance for voice service in hybrid IP networks", ITU-T Rec. Y.1530, Nov. 2007.

[6] "SIP-based call processing performance", ITU-T Rec. Y.1531, Nov. 2007.

[7] S. Kaczmarek and M. Sac, "Traffic modeling in IMS-based NGN networks", *Gdańsk University of Technology Faculty of ETI Annals*, vol. 1, no 9, pp. 457–464, 2011.

[8] S. Kaczmarek and M. Sac, "Zagadnienia inżynierii ruchu w sieciach NGN bazujących na IMS" ("Traffic engineering aspects in IMS-based NGN networks"), in *Biblioteka teleinformatyczna, t. 6. Internet 2011* (*Teleinformatics library, vol. 6. Internet 2011*), D. J. Bem *et al.*, Eds. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2012, pp. 63–115 (in Polish).

[9] S. Kaczmarek, M. Kaszuba and M. Sac, "Simulation model of IMS/NGN call processing performance", *Gdańsk University of Technology Faculty of ETI Annals*, vol. 20, pp. 25–36, 2012.

[10] S. Kaczmarek and M. Sac, "Traffic Model for Evaluation of Call Processing Performance Parameters in IMS-based NGN", in *Information Systems Architecture and Technology: Networks Design and Analysis*, A. Grzech *et al.*, Eds. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2012, pp. 85–100.

[11] S. Kaczmarek and M. Sac, "Message Inter-Arrival and Inter-Departure Time Distributions in IMS/NGN Architecture", in *Proc. 17th Polish Teletraffic Symp. PTS 2012*, Zakopane, Poland, 2012, pp. 37–43.

[12] T. Osogami and M. Harchol-Balter, "Closed form solutions for mapping general distributions to quasi-minimal PH distributions", *Perform. Eval.*, vol. 63, no. 6, pp. 524–55, 2006.

[13] A. Bobbio, A. Horvath and M. Telek, "Matching three moments with minimal acyclic phase type distributions", *Stoch. Mod.*, vol. 21, no. 2–3, pp. 303–326, 2005.

[14] M. Telek and G. Horvath, "A minimal representation of Markov arrival processes and a moments matching method", *Perform. Eval.*, vol. 64, no. 9–12, pp. 1153–1168, 2007.

[15] A. van de Liefvoort, "The moment problem for continuous distributions", Tech. rep., University of Missouri, WP-CM-1990-02, Kansas City, USA, 1990.

[16] S. Asmussen, O. Nerman and M. Olsson, "Fitting Phase-type distributions via the EM Algorithm", *Scandinavian J. Statist.*, vol. 23, no. 4, pp. 419–441, 1996.

[17] "Functional requirements and architecture of next generation networks", ITU-T Rec. Y.2012, Apr. 2010.

[18] "IMS for next generation networks", ITU-T Rec. Y.2021, Sept. 2006.

[19] "Resource and admission control functions in next generation networks", ITU-T Rec. Y.2111, Nov. 2008.

[20] "Resource control protocol no. 1, version 2 – Protocol at the Rs interface between service control entities and the policy decision physical entity", ITU-T Rec. Q.3301.1, Jun. 2010.

[21] M. Pirhadi, S. M. Safavi Hemami and A. Khademzadeh, "Resource and admission control architecture and QoS signaling scenarios in next generation networks", *World Appl. Sci. J. 7 (Special Issue of Computer & IT)*, pp. 87–97, 2009.

[22] OMNeT++ Network Simulation Framework [Online]. Available: http://www.omnetpp.org

[23] MATLAB – The Language of Technical Computing [Online]. Available: http://www.mathworks.com/products/matlab

[24] V. S. Abhayawardhana and R. Babbage, "A traffic model for the IP Multimedia Subsystem (IMS)", in *Proc. IEEE 65th Veh. Technol. Conf. VTC 2007-Spring*, Dublin, Ireland, 2007.

[25] T. Czachórski, "Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych" ("Queuing models in evaluation of effectiveness of computer networks and systems"). Gliwice: Pracownia Komputerowa Jacka Skalmierskiego, 1999 (in Polish).

[26] Moment Matching Algorithms [Online]. Available: http://www.cs.cmu.edu/ osogami/code/momentmatching/index.html

[27] BuTools Program Packages [Online]. Available: http://webspn.hit.bme.hu/ telek/tools/butools/butools.html

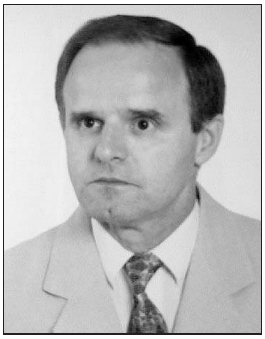
[28] G. W. Corder and D. I. Foreman, *Nonparametric Statistics for Non-Statisticians: A Step-by-Step Approach*. Wiley, 2009.

[29] S. Kaczmarek and M. Sac, "Analysis of IMS/NGN call processing performance using G/G/1 queuing systems approximations", *Przegl. Telekomun. i Wiadom. Telekomun.* (*Telecommun. Rev. & Telecommun. News*), no. 8–9, pp. 702–710, 2013.

[30] "Centralized RACF architecture for MPLS core networks", ITU-T Rec. Y.2175, Nov. 2008.

[31] "Ethernet QoS control for next generation networks", ITU-T Rec. Y.2113, Jan. 2009.

[32] "Requirements for the support of flow state aware transport technology in an NGN", ITU-T Rec. Y.2121, Jan. 2008.



Sylwester Kaczmarek received his M.Sc. in Electronics Engineering, Ph.D. and D.Sc. in Switching and Teletraffic Science from the Gdańsk University of Technology, Gdańsk, Poland, in 1972, 1981 and 1994, respectively. His research interests include: IP QoS and GMPLS networks, switching, QoS routing, teletraffic, multimedia services and quality of services. Currently, his research is focused on developing and applicability of VoIP and IMS/NGN technology. So far he has published more than 200 papers. Now he is the Head of Teleinformation Networks Department at GUT.

E-mail: kasyl@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Maciej Sac received his M.Sc. degree in Telecommunications from Gdańsk University of Technology in 2009. Since 2009 he has been a Ph.D. student at Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics. His research interests are focused on ensuring reliability and Quality of Service in IMS/NGN networks by the means of traffic engineering.

E-mail: Maciej.Sac@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Paper

Traffic Type Influence on Performance of OSPF QoS Routing

Michał Czarkowski, Sylwester Kaczmarek, and Maciej Wolff

Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

Abstract—Feasibility studies with QoS routing proved that the network traffic type has influence on routing performance. In this work influence of self-similar traffic for network with DiffServ architecture and OSPF QoS routing has been verified. Analysis has been done for three traffic classes. Multiplexed On-Off model was used for self-similar traffic generation. Comparison of simulation results was presented using both relative and non-relative measures for three traffic classes. Results were commented and analyzed. The basic conclusion is that performance for streaming and best-effort class for self-similar traffic is higher than performance for the same class with exponential traffic (Poisson). The other important conclusion is relation between performance differences and offered traffic amount.

Keywords—DiffServ, exponential traffic, network performance, OSPF routing, packets networks, QoS, self-similar traffic.

1. Introduction

Modern telecommunication networks are using many different technologies. The most important technology and the most developed at the time are packet networks. Unfortunately existing packet networks don't guarantee quality. That is why modern convergent technologies are challenging for telecommunication operators. Quality of Service (QoS) guarantee is necessary. One of the basic examples of QoS ensuring solution is DiffServ architecture with QoS routing.

Studies in packet networks proved that traffic in packet networks have self-similar character [1]. Unfortunately until now studies of self-similar traffic were focused on single services device [2], [3], [4] or devices connected in a chain. There was no research for networks with many routers and DiffServ architecture in real network structure.

In this paper performance of networks with QoS routing and DiffServ architecture for different network structures with self-similar traffic was analyzed. Network performance with exponential offered traffic and self-similar offered traffic was compared. Results for exponential offered traffic for OSPF routing were captured from existing work [5]. Results for self-similar traffic were obtained in this study. The simulation model is based on model from [5].

The paper is organized into six sections. Section 2 describes routing algorithm and realization of DiffServ architecture. Section 3 describes two traffic types: exponential offered traffic and self-similar offered traffic. In this part self-similar offered traffic model used for simulation model is described. Section 4 describes simulation model, its structure and features. Section 5 describes the simulation and presents the results. In this section also conclusions are presented and explained. Section 6 presents summary and description of next studies steps.

2. OSPF QoS Routing

The studied networks use OSPF routing algorithm and within this Dijkstra algorithm [6] to determine shortest route between source and destination router. This algorithm is sometimes called Shortest Path First (SPF). There are identical routes for all traffic classes in simulation model. Metric used for SPF algorithm implementation is metric from classical implementation of OSPF. This is product of constant number and inverse link capacity.

Simulation model fully implemented DiffServ architecture, where routers are divided into edge routers and core routers. Core routers handle and send packets with defined politics only. Edge routers define traffic class of packet and accept or discard traffic stream. Edge routers have also core routers functions. In this implementation of edge routers, decisions about acceptance or rejection of streams are based on actual network load. This algorithm is described in detail in [5].

Method of handling packets depends on the traffic class in DiffServ architecture. If edge router accepts packet, packet class is marked and information about it is saved in header. Next edge routers make decisions about traffic class and packet handling method on the basis of packet class information saved in header.

3. Traffic Type: Exponential and Self-Similar

Exponential offered traffic is short range dependent (SRD). This traffic is easy to simulate.

Self-similar offered traffic is long range dependent (LRD). Between events in this traffic there are dependencies in short and long time scale. Hurst coefficient [7] represents level of this dependency. Range of Hurst coefficient value for network traffic is between 0.5 and 1. Network traffic with Hurst coefficient equal to 0.5 is SRD traffic, and an example of this traffic realization is exponential offered traffic. Network traffic with Hurst coefficient greater than 0.5 and less than 1 is self-similar traffic [7].

Multiplexed On-Off model was used for self-similar offered traffic modeling. This model is multiplexing many two state streams. In first state, called On, packets are generated with constant time interval. Packets aren't generated in second state, which is called Off. On state time is determined through Pareto distribution, Off state time is determined through exponential distribution. This model is described in detail in [8].

4. Simulation Model

Simulation model is based on the model described in detail in [5]. It allows verifying performance of the networks with different QoS routing algorithm and different offered traffic types. Model was implemented using discrete event network simulator called Omnet++ [9]. The implementation has been provided using standard STL C++ libraries and functions. This model fully implements DiffServ architecture. As an addition to the model from work [5] self-similar offered traffic generator has been added. This self-similar traffic generator is implemented using the multiplexed On-Off streams.

Model consists of three basic network components: edge routers, core routers and central module. The central module component is used for data storage. It is combined with all network routers via virtual connections which are used for routing tables transfer. The global object shares also the interface which can be used for communication between the object and edge/core routers. The object stores also information about the network topology and all Link State Protocol information. Edge and core routers deliver the functions specified according to the DiffServ architecture. Both routers service systems are the same and specified by the DiffServ architecture. Service systems consist of two queuing policies Priority Queuing (PQ) and Weighted Fair Queuing (WFQ). Streaming traffic is attached to first queue of PQ and contains very short buffer just for few packets (REM model). This particular buffer should be no longer than 5 packets. Two other traffic classes (elastic, best effort) are directed to WFQ with ω_{AF} and ω_{BE} weight parameters respectively. The output from WFQ is directed to second input of PQ without additional buffering. Buffers length for elastic traffic should be not too long due to QoS constraint given to this class [10]. Best effort buffer is not set to a large value to omit resources waist. Packets are generated independently in edge router for all three traffic classes. The single generator of traffic class generated packets to all possible edge routers (all relations). Each generator is described by the time periods distribution between next generated packets, like uniform, exponential, Pareto, etc. Edge routers are at the same time traffic receivers. Hurst coefficient of self-similar traffic for three traffic classes can set independently in this model. Each edge router is connected with only one core router. The capacity of links connecting edge and core router is much larger than the capacity of links in the core (not to cause bottleneck here). The core router is similar to the edge router with the difference that it does not include the traffic generator block and traffic receiver block. Core routers do not generate packet but just process the packets and forward them to the output links according to the routing tables.

AC function to keeps QoS in this model is realized through acceptance or rejection stream in edge router. These operations are needed because of the required QoS and limit packets in network. First packet in each stream is initial packet. If edge router receives it, router calculate path for stream first based on SPF algorithm. In next step router verifies QoS parameters: delay (IPTD – IP Time Delay), delay variation (IPDV – IP Delay Variation), loss ratio (IPLR – IP Loss Ratio). These parameters are verified for end-to-end link based on the current network state for streaming class. In terms of capacity there is a check for all intermediate links in the path if all of them include the required bandwidth amount. The QoS parameters values are taken from [10]. If verified path meets above values then stream is accepted, and path is saved in route table and next in packet headers, else stream is rejected. Saving this information in header is needed for simulation process.

5. Results of Studies

5.1. Simulation Parameters and Scenarios

Simulation model has been applied for three structures with different connections density. The structures are Sun, NewYork and Norway [11]. Connections density is defined as number of links between routers divided by number of routers. NewYork structure is network with maximum connections density equal 3.06. Sun structure is network with least connections density equal 1.5. Connections density of Norway structure is between Sun and NewYork and is equal 1.89. In this paper results for Sun, Norway and NewYork are presented.

For each structure the simulation has been done with forty different traffic classes proportions. For each ten proportion: level of best-effort traffic class is constant (1–10, 11–20, 21–30 and 31–40), level of streaming traffic class is increase and level of elastic traffic class is decrease. These proportions are presented in Table 1. For example, for first proportion: 1% offered traffics are stream traffic, 19% traffics are elastic traffic and 80% traffics are best-effort traffic. The length packet for each traffic class is constant and for streaming traffic class is equal 160 bytes, for elastic traffic class is equal 500 bytes and for best-effort traffic class is equal 1500 bytes.

Buffer length for streaming traffic class is equal 5 packets (REM model), for elastic traffic is equal 10 packets, for best-effort traffic class is equal 50 packets. Input weight for handling of elastic class in WFQ is set to 0.4 and input weight of best effort class is set to 0.6.

The simulation time was set to 3600 s. For each traffic class proportion and each structure the simulations has been repeated six times, with exception Sun structure. For this structure simulation was done twelve times. These number of repetitions is required to get appropriate confidence intervals.

Table 1
Proportions of traffic

No. of proportion	Stream traffic	Elastic traffic	Best-effort traffic
1	0.01	0.19	0.8
2	0.03	0.17	
3	0.05	0.15	
4	0.07	0.13	
5	0.09	0.11	
6	0.11	0.09	
7	0.12	0.08	
8	0.13	0.07	
9	0.14	0.06	
10	0.15	0.05	
11	0.02	0.28	0.7
12	0.06	0.24	
13	0.1	0.2	
14	0.12	0.18	
15	0.14	0.16	
16	0.16	0.14	
17	0.18	0.12	
18	0.2	0.1	
19	0.24	0.06	
20	0.28	0.02	
21	0.05	0.35	0.6
22	0.08	0.32	
23	0.12	0.28	
24	0.14	0.26	
25	0.18	0.22	
26	0.24	0.16	
27	0.28	0.12	
28	0.32	0.08	
29	0.34	0.06	
30	0.35	0.05	
31	0.1	0.4	0.5
32	0.13	0.37	
33	0.16	0.34	
34	0.18	0.32	
35	0.2	0.3	
36	0.24	0.26	
37	0.28	0.22	
38	0.32	0.18	
39	0.38	0.12	
40	0.4	0.1	

The result of simulation is network performance for each structure and for each traffic proportion. The performance is the number of packet processed by network.

Hurst coefficient for stream, elastic and best effort is 0.9. This value is results of study technical publications on self-similar traffic. Analyze of the Hurst coefficient of stream traffic, VoIP traffic, is in the work [12]. The Hurst coefficient of elastic traffic, MPEG traffic, is presented in [13]. Study of the Hurst coefficient of best effort traffic is shown in [1].

5.2. Relative and Non Relative Measure

The simulation results are presented using two measures: non relative and relative. Non relative measure described amount of serviced packets in network for each traffic class. This amount in one network structure, for each proportion, for one traffic class shows in one figure. The relative measure is described by parameter

$$A_{rel} = \frac{A_{SS} - A_{exp}}{A_{exp}}. \tag{1}$$

In Eq. (1) A_{exp} is the mean amount services packet for exponential offered traffic for each traffic class for one structure, A_{SS} is the mean amount packet for self-similar offered traffic for each traffic class, for one structure. This measure can show relative differences between performance network with self-similar offered traffic and network with exponential offered traffic. Values of this measure, for one network structure, for each traffic class proportion, for one traffic class or combination of traffic class are presented in one figure.

5.3. Results

In Figs. 1–10 are presented results for Sun structure. In first five diagrams are presented results in non-relative measure. Next five presents results in relative measure. In Figs. 11–20 are presented results for Norway structure, and in Figs. 21–30 results for NewYork structure was shown in the same layout as for Sun structures.

First, results in non-relative measure are described. In Figs. 1–5 are presented results in amount packet serviced over network for traffic class: streaming, elastic, best-effort, aggregate streaming and elastic traffic and aggregate all traffic for Sun structure. For most traffic proportions for streaming, elastic and best-effort traffic confidence intervals of network performance for exponential offered traffic and self-similarity offered traffic are separable. Only for several proportions for elastic traffic confidence intervals overlap. For aggregate measures is similar, for the most traffic proportions the confidence intervals are separable.

For NewYork structure non-relative measures are presented in Figs. 21–25. For this structure, just as Sun one the most confidence intervals are separable. Only for elastic traffic almost all confidence intervals are overlap. The same results are for Norway structures, for which results presented in Figs. 11–15, only for elastic traffic almost all confidence intervals are overlap.

All other results based on relative measures. For Sun structures these results presented in Figs. 6–10, for NewYork in Figs. 26–30 and for Norway in Figs. 16–20.

First the results for Sun structures for streaming, elastic and best-effort traffic are described. Results for aggregate traffic are described as the second ones in this paper. For self-similar offered traffic performance is higher about 30% comparing to exponential offered traffic for the stream traffic class. For elastic traffic class confidence intervals are overlap and comparing results is impossible. For best-effort traffic class, performance for self-similar offered

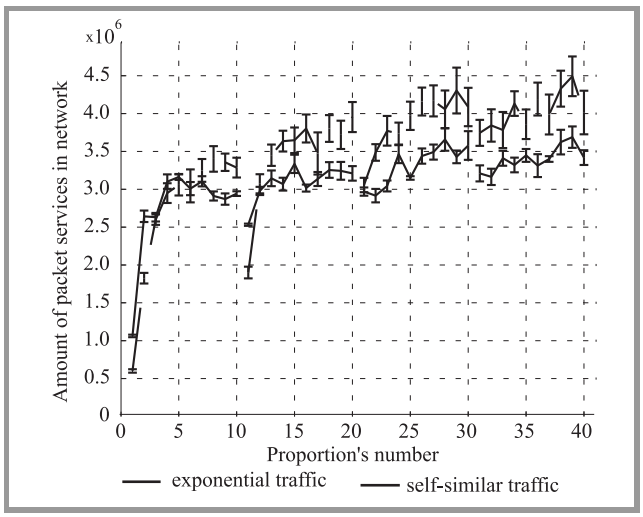


Fig. 1. Streaming class packet services for Sun network structure with exponential and self-similar offered traffic.

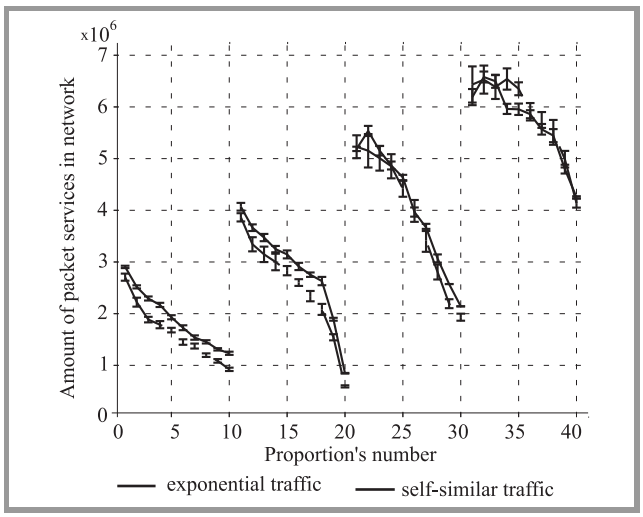


Fig. 2. Elastic class packet services for Sun network structure with exponential and self-similar offered traffic.

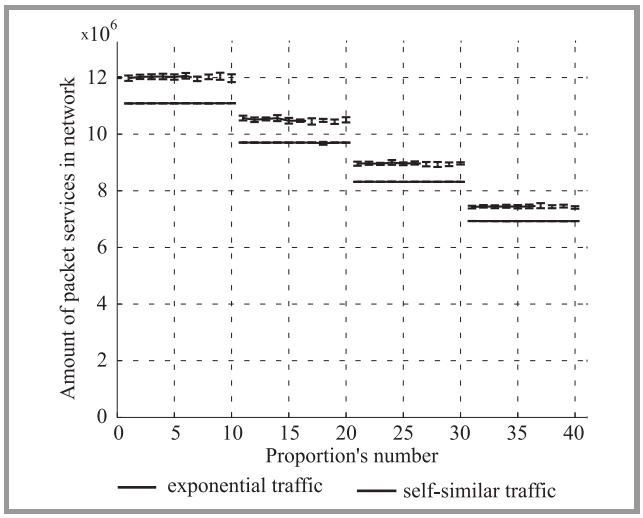


Fig. 3. Best-effort class packet services for Sun network structure with exponential and self-similar offered traffic.

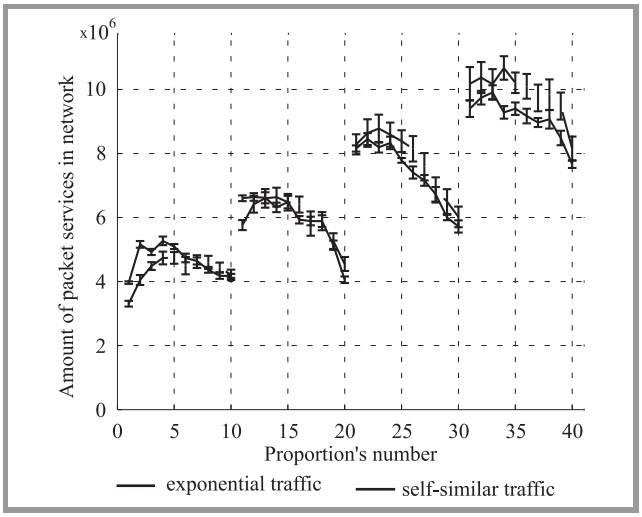


Fig. 4. Streaming and elastic class packet services for Sun network structure with exponential and self-similar offered traffic.

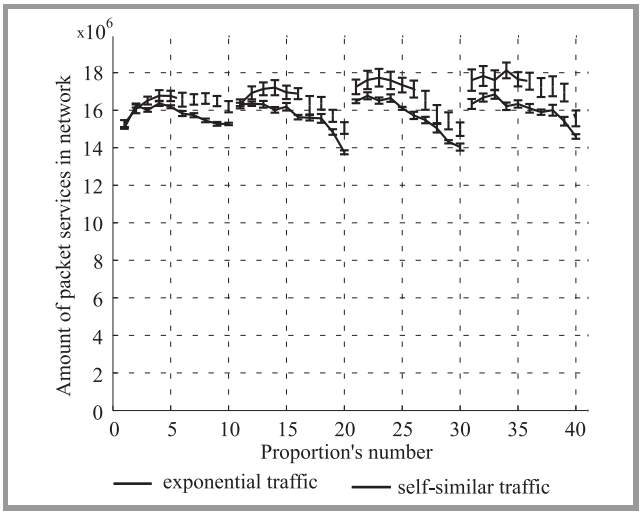


Fig. 5. All packet services for Sun network structure with exponential and self-similar offered traffic.

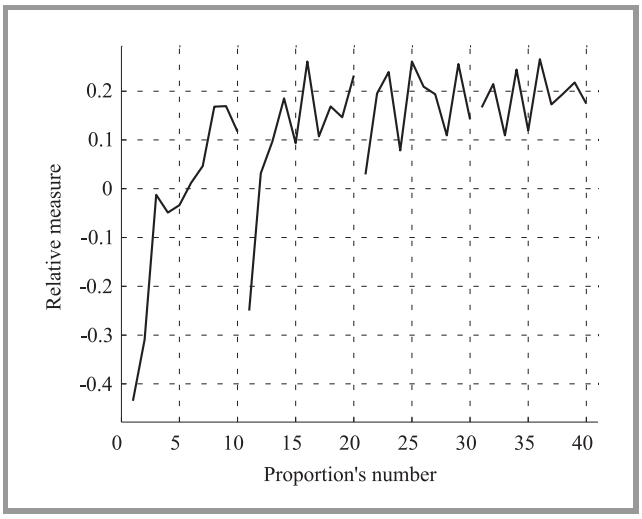


Fig. 6. Streaming class packet services for Sun network structure with exponential and self-similar offered traffic.

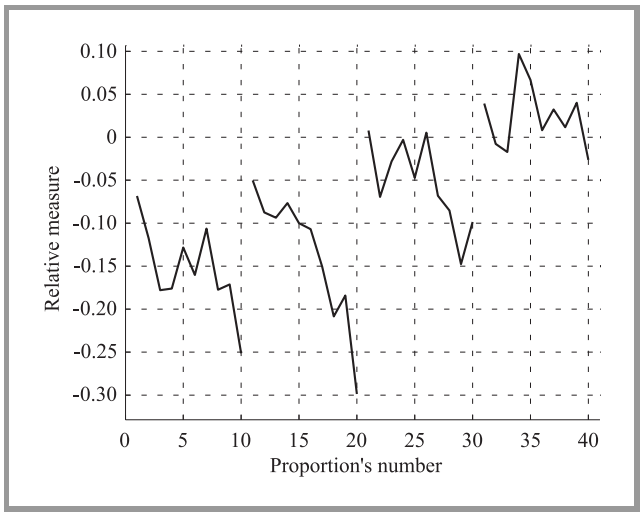


Fig. 7. Elastic class packet services for Sun network structure with exponential and self-similar offered traffic.

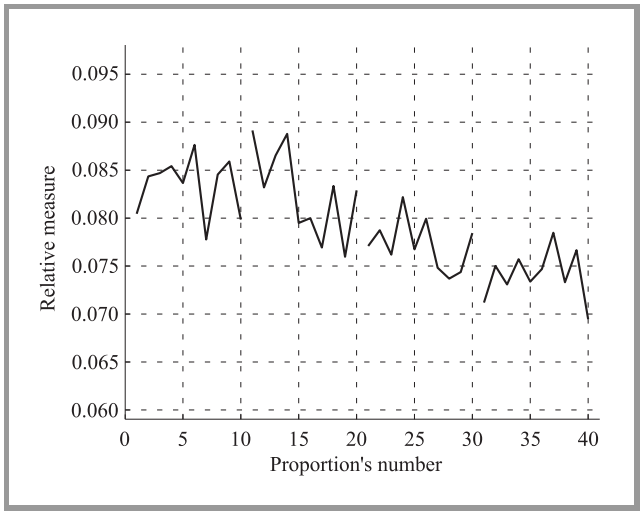


Fig. 8. Best-effort class packet services for Sun network structure with exponential and self-similar offered traffic.

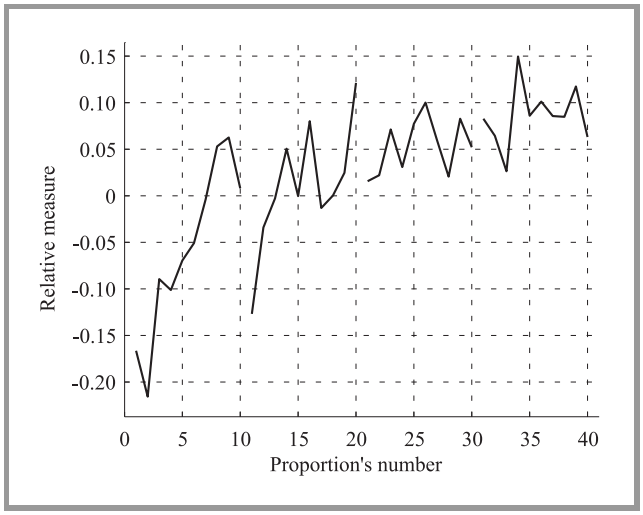


Fig. 9. Streaming and elastic class packet services for Sun network structure with exponential and self-similar offered traffic.

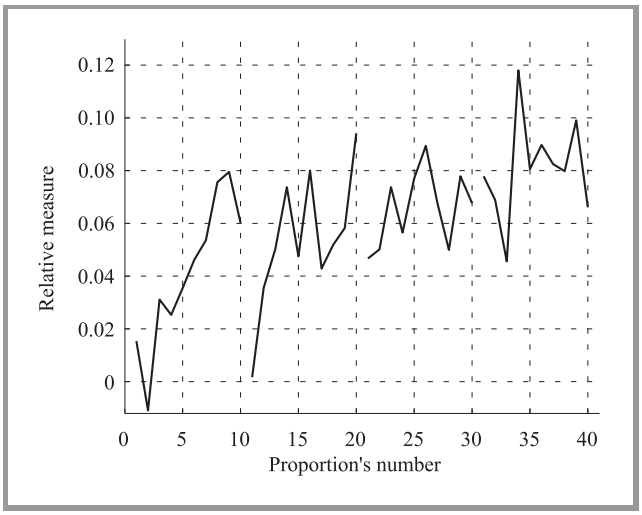


Fig. 10. All packet services for Sun network structure with exponential and self-similar offered traffic.

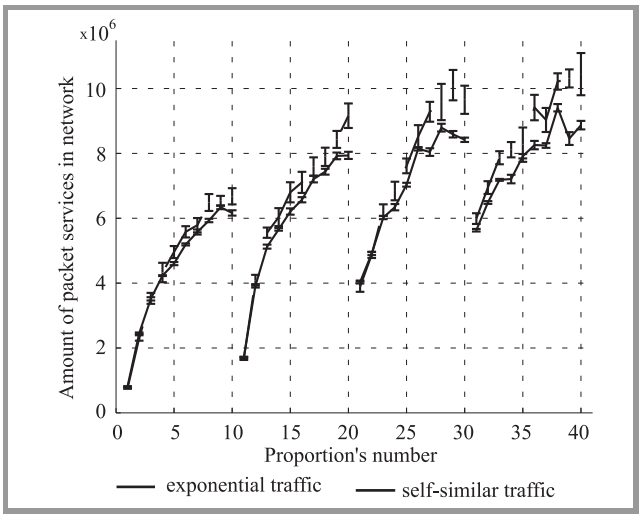


Fig. 11. Streaming class packet services for Norway network structure with exponential and self-similar offered traffic.

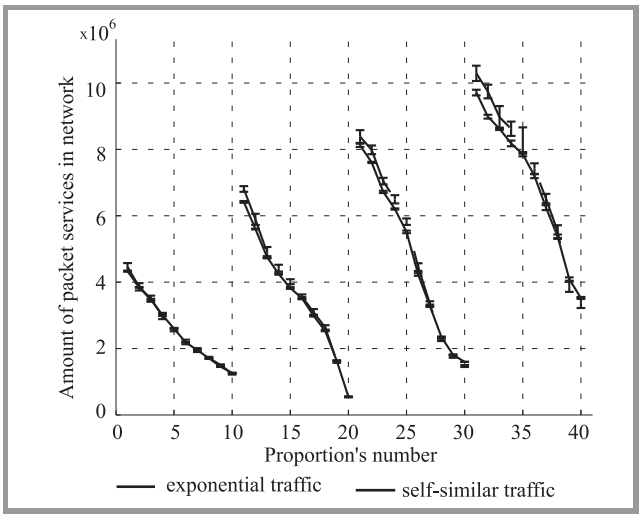


Fig. 12. Elastic class packet services for Norway network structure with exponential and self-similar offered traffic.

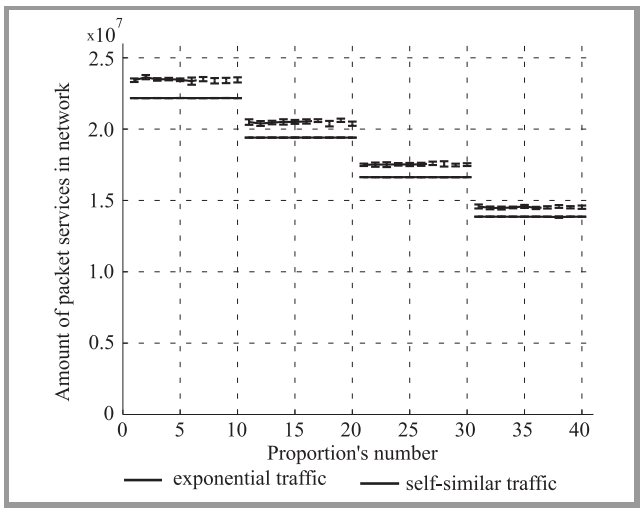


Fig. 13. Best-effort class packet services for Norway network structure with exponential and self-similar offered traffic.

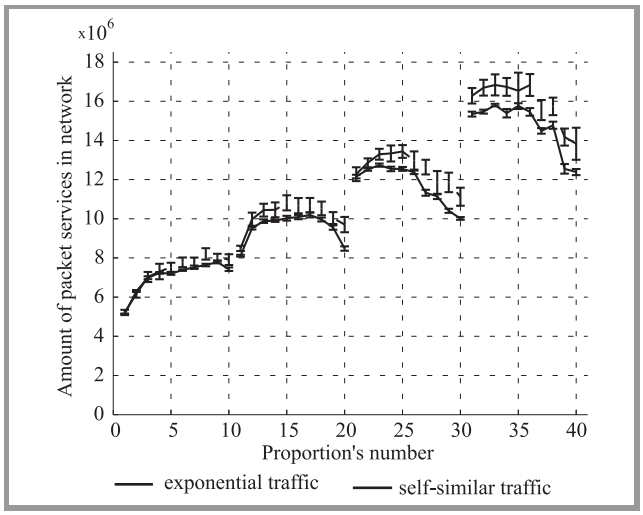


Fig. 14. Streaming and elastic class packet services for Norway network structure with exponential and self-similar offered traffic.

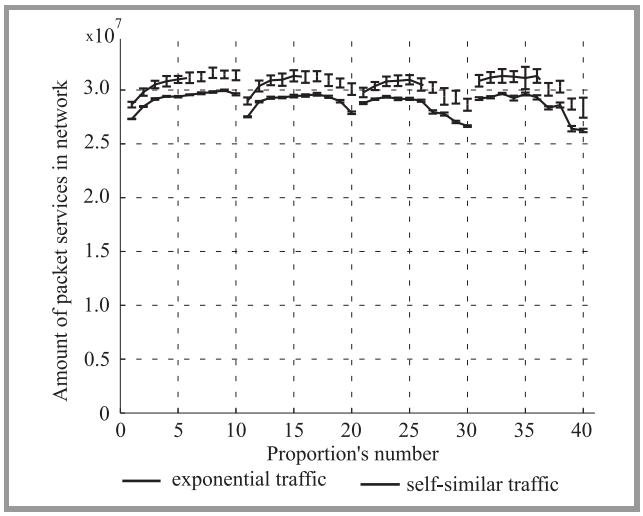


Fig. 15. All packet services for Norway network structure with exponential and self-similar offered traffic.

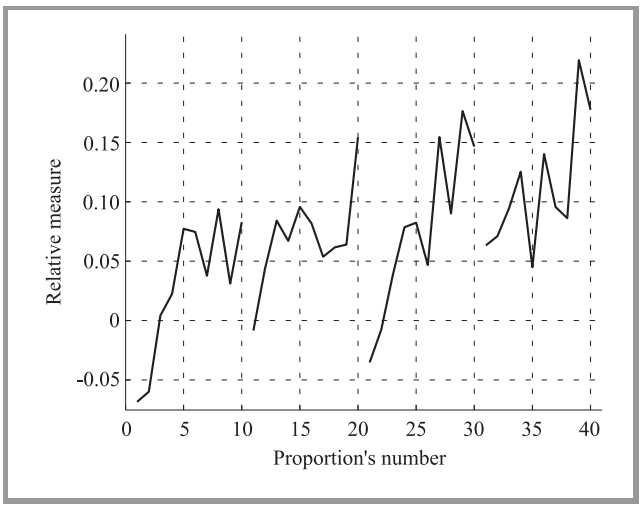


Fig. 16. Streaming class packet services for Norway network structure with exponential and self-similar offered traffic.

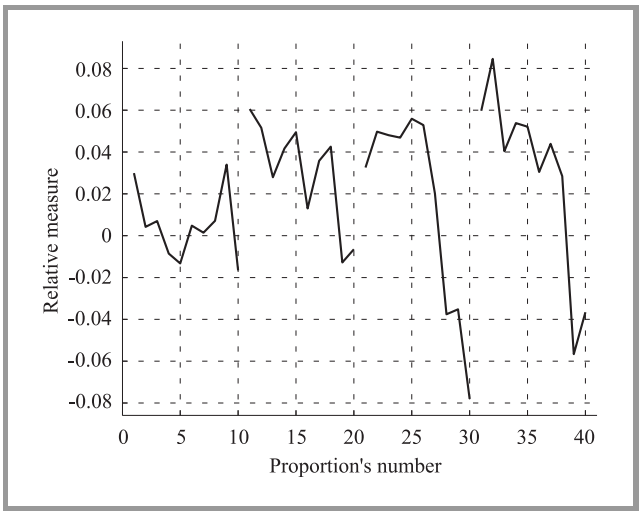


Fig. 17. Elastic class packet services for Norway network structure with exponential and self-similar offered traffic.

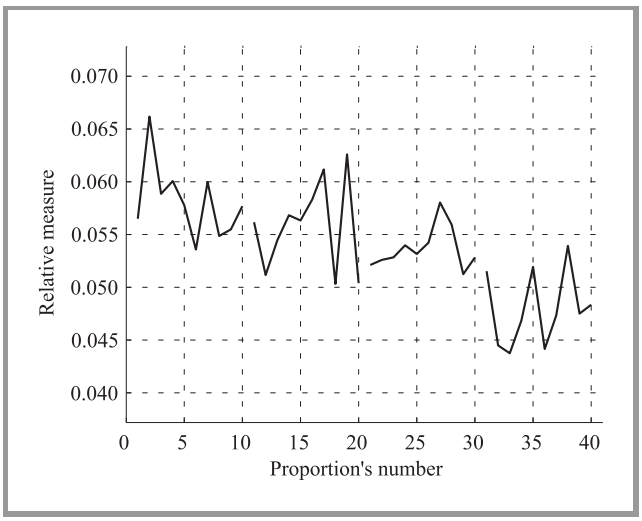


Fig. 18. Best-effort class packet services for Norway network structure with exponential and self-similar offered traffic.

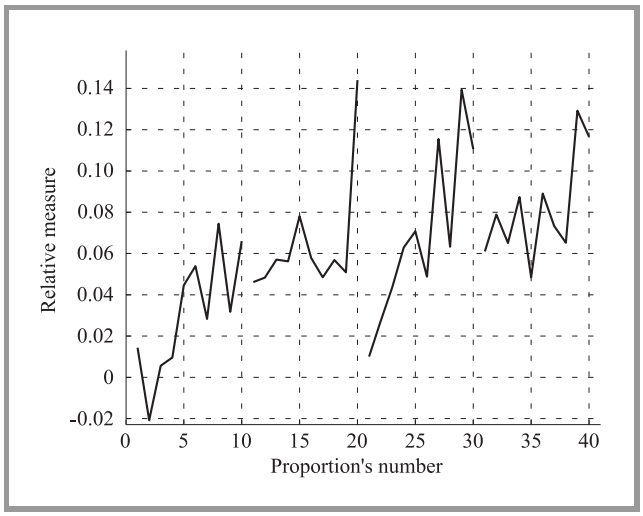


Fig. 19. Streaming and elastic class packet services for Norway network structure with exponential and self-similar offered traffic.

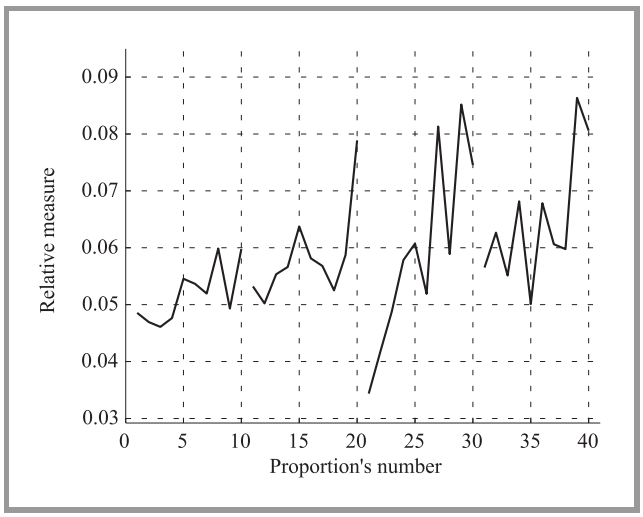


Fig. 20. All packet services for Norway network structure with exponential and self-similar offered traffic.

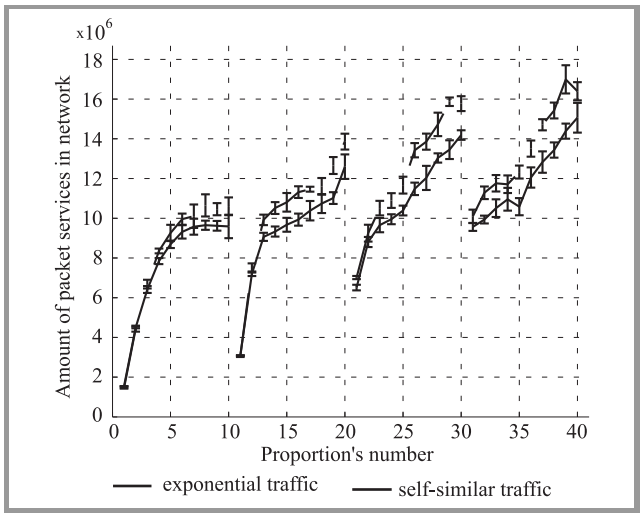


Fig. 21. Streaming class packet services for NewYork network structure with exponential and self-similar offered traffic.

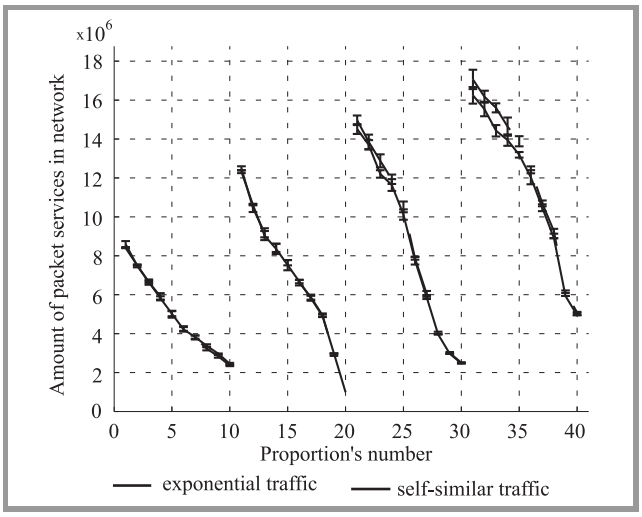


Fig. 22. Elastic class packet services for NewYork network structure with exponential and self-similar offered traffic.

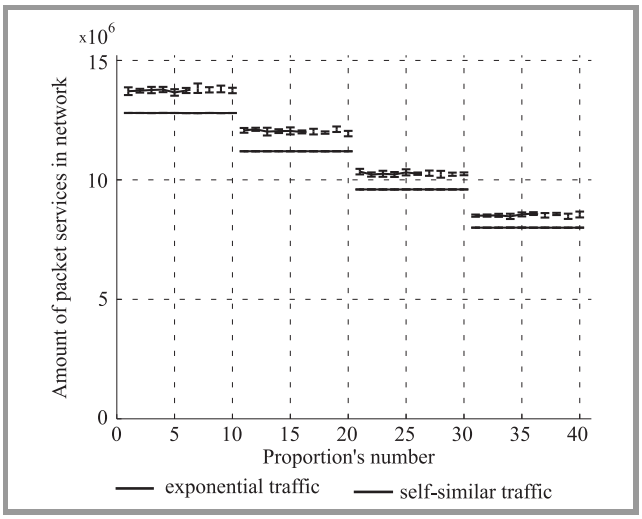


Fig. 23. Best-effort class packet services for NewYork network structure with exponential and self-similar offered traffic.

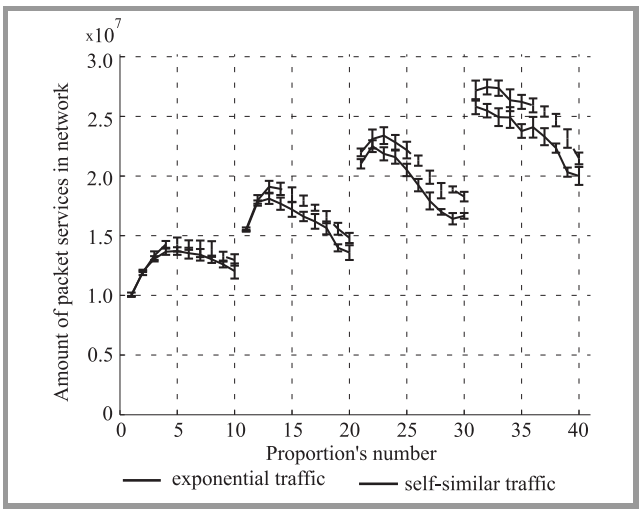


Fig. 24. Streaming and elastic class packet services for NewYork network structure with exponential and self-similar offered traffic.

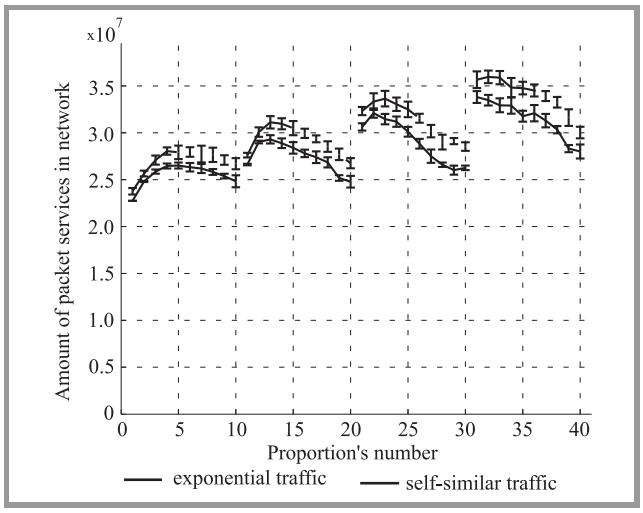


Fig. 25. All packet services for NewYork network structure with exponential and self-similar offered traffic.

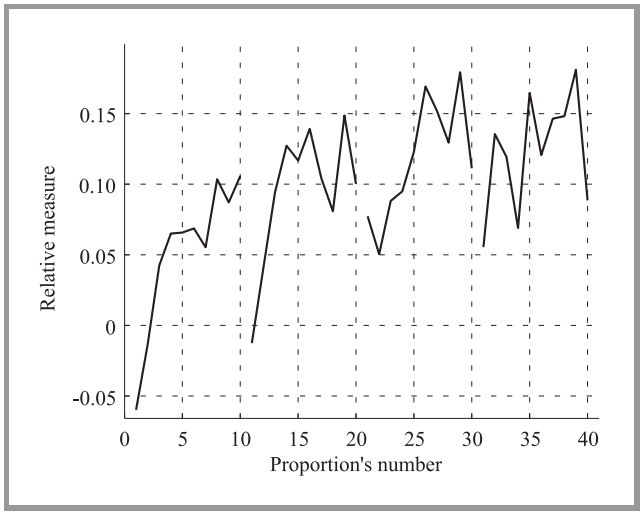


Fig. 26. Streaming class packet services for NewYork network structure with exponential and self-similar offered traffic.

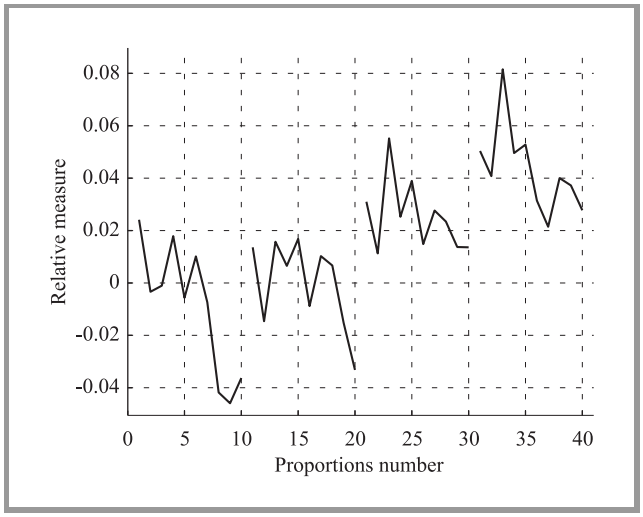


Fig. 27. Elastic class packet services for NewYork network structure with exponential and self-similar offered traffic.

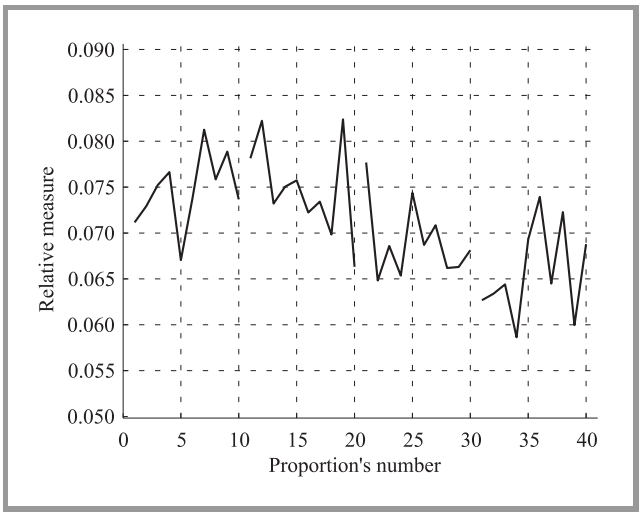


Fig. 28. Best-effort class packet services for NewYork network structure with exponential and self-similar offered traffic.

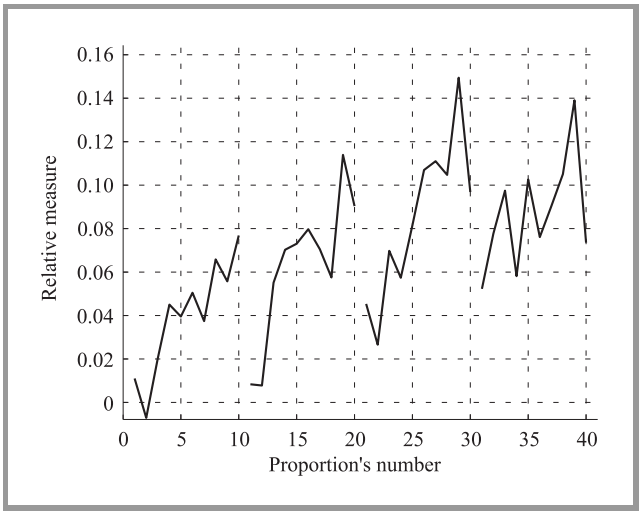


Fig. 29. Streaming and elastic class packet services for NewYork network structure with exponential and self-similar offered traffic.

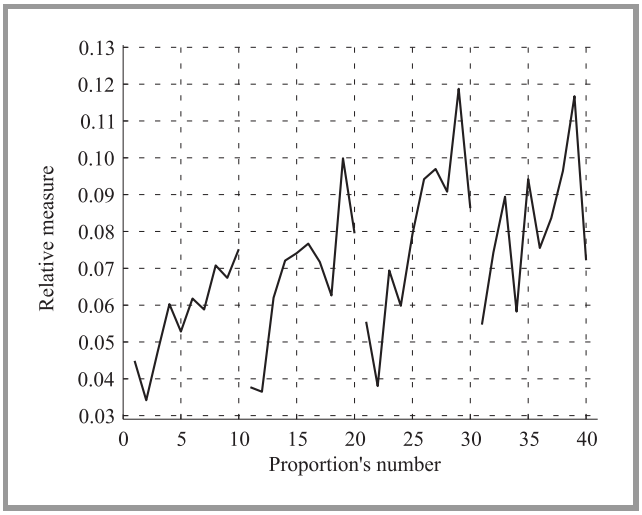


Fig. 30. All packet services for NewYork network structure with exponential and self-similar offered traffic.

traffic is higher about 7% in comparison to exponential offered traffic.

The same conclusions are for Norway and NewYork structures. Results for Norway structure for traffic class: streaming, elastic and best-effort, aggregate streaming and elastic and aggregate all traffic are presented in Figs. 11–15. Results for NewYork structure for traffic class: streaming, elastic and best-effort, aggregate streaming and elastic and aggregate all traffic are presented in Figs. 21–25. Major difference between results for NewYork or Norway and Sun structure is performance growth in percentage for streaming traffic class. Maximum performance is 22% higher for Norway structure and maximum growth performance is 18% for NewYork one. Result of this analysis is the proposal – difference of performance for streaming traffic class depends on connections density. If density grows then the network performance difference is lower. This statement is confirmed in relative measure. Similar results are for best-effort traffic class, but difference is less, and it is more visible in relative measure.

Now the results for aggregate traffic will be described. For NewYork and Norway structure with aggregate streaming and elastic traffic, more traffic is serviced by network with self-similar offered traffic than by network with exponential offered traffic. Maximum performance growth between network with self-similar traffic and network with exponential traffic is equal to 15% for NewYork structure and 14% for Norway structure. For Sun, which is the smallest one for some proportions there is lower network performance for self-similarity traffic than for exponential.

The results for aggregated traffic for all proportions, for all structures prove higher network performance for network with self-similar traffic than network with exponential one. Other kind of analysis is trend analysis of relative measure. Result of this analysis show impact of amount of offered traffic and the traffic character for network performance. Results for Sun structure are presented first. Difference of performance between network with self-similar offered traffic and network with exponential one, for streaming traffic class is higher while using higher offered traffic. The same results are for best-effort traffic class. The performance difference for elastic traffic class requires additional comment. If there is a visible gain on performance difference between network with self-similar offered traffic and network with exponential offered traffic, it is higher with growth amount elastic traffic class. In case of lower performance difference between networks with self-similar offered traffic and network with exponential offered traffic, this is decrease with decrease amount elastic traffic class. Equivalent conclusions are for Norway and NewYork structures and results for these structures are presented in Figs. 26–30 and Figs. 16–20.

Important result is trend analysis of relative measure for aggregate traffic and at the same time analysis of non-relative measures for streaming traffic. Larger amount of serviced packet for streaming traffic caused also gain within relative measure for aggregated traffic. Next conclusion is – if offered traffic had a self-similar character more traffic was serviced with increasing streaming offered traffic than for exponential character of offered traffic. A similar conclusion is for aggregated streaming and elastic traffic. Network can service more streaming and elastic traffic of self-similar traffic type than exponential traffic type with increasing streaming traffic amount in the network.

6. Summary

The main conclusion is that higher network performance was noticed for streaming and best effort traffic class for self-similar offered traffic type than for exponential offered traffic type. Important is also higher network performance for aggregate traffic for self-similar traffic type. Difference of performance for elastic and best-effort traffic class depends on connections density. If network density is growing the difference on network performance lowers. Other conclusion is the relation between different performance gain and increase of the offered traffic, but this relation is complex and may depend on buffers length and connection density. To fully confirm this thesis further research is required.

References

[1] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)", *IEEE/ACM Trans. Netw.*, vol. 2, pp. 1–15, 1994.

[2] X. Tan and Y. Zhuo, "Simulation based analysis of the performance of self similar traffic", in *Proc. 4th Int. Conf. Com. Sci. Edu. ICCSE 2009*, Nanning, China, 2009.

[3] H. S. Acharya, S. R. Dutta and R. Bhoi, "The Impact of self-similarity Network traffic on quality of services (QoS) of Telecommunication Network", *Int. J. IT Eng. Appl. Sci. Res. (IJIEASR)*, vol. 2, no. 2, 2013.

[4] Y. Koucheryavy, J. Harju and V. B. Iversen, "Multiservice IP network QoS parameters estimation in presence of self-similar traffic", in *Proc. 6th Int. Conf. Next Gen. Teletraf. Wired/Wirel. Adv. Netw. NEW2AN 2006*, St. Petersburg, Russia, 2006.

[5] M. Czarkowski and S. Kaczmarek, "Dynamic unattended measurement based routing algorithm for DiffServ architecture", in *Proc. 14th Int. Telecom. Netw. Strat.Plan. Symp. NETWORKS 2010*, Warsaw, Poland, 2010, pp. 1–6.

[6] J. T. Moy, *OSPF Anatomy of an Internet Routing Protocol*. Addison-Wesley, 2001.

[7] O. I. Sheluhin, M. S. Smolskiy, A. V. Osin, *Self-Similar Processes in Telecommunications*. Wiley, 2007.

[8] N. Likhanov, B. Tsybakov and N. Georganas, "Analysis of an ATM buffer with self-similar ("fractal") input traffic", in *Proc. 14th Ann. Joint Conf. IEEE Comp. Commun. Soc. – Bringing Information to People INFOCOM '95*, Boston, MA, USA, 1995, vol. 3, pp. 985–992.

[9] OMNeT++ [Online]. Available: http://www.omnetpp.org/

[10] "Network performance objectives for IP-based services", ITU-T Rec. Y.1504, Feb. 2006.

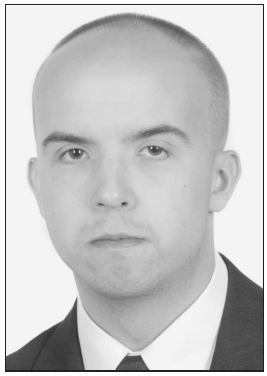
[11] sndlib [Online]. Available: http://sndlib.zib.de/

[12] T. D. Dang, B. Sonkoly, S. Molnar, "Fractal analysis and modeling of VoIP traffic", in *Proc. 11th Int. Telecom. Netw. Strat.Plan. Symp. NETWORKS 2004*, Vienna, Austria, 2004, pp. 123–130.

[13] J. Cano and P. Manzoni, "On the use and calculation of the Hurst parameter with MPEG videos data traffic", in *Proc. 26th Euromicro Conf.*, Maastricht, the Netherland, 2000, vol. 1, pp. 448–455.

Michał Czarkowski received M.Sc. and Ph.D. degree in Telecommunication Systems from Gdańsk University of Technology (GUT), Gdańsk, Poland, in 2004 and 2011, respectively. He is currently cooperating with GUT within QoS routing and effective routing algorithms. His interests focus also on QoS in packet networks.

E-mail: michal.czarkowski@intel.com
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland



Maciej Wolff received M.Sc. degree in Telecommunication Systems from Gdańsk University of Technology (GUT) Gdańsk, Poland 2012. His Master's thesis focused on the impact of traffic type for performance of networks. He began Ph.D. study in GUT in 2012.

E-mail: maciej.wolff@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Sylwester Kaczmarek – for biography, see this issue, p. 17.

Paper

Quality Aware Virtual Service Delivery System

Mariusz Fraś and Jan Kwiatkowski

Wrocław University of Technology, Wrocław, Poland

Abstract—The problem of providing support for quality of service (QoS) guarantees is studied in many areas of information technologies. In recent years the evolution of software architectures led to the rising prominence of the Service Oriented Architecture (SOA) concept. For Web-based systems there are three attributes that directly relate to everyday perception of the QoS for the end user: availability, usability, and performance. The paper focuses on performance issues of service delivery. The architecture of Virtual Service Delivery System (VSDS), a tool to serve requests for synchronized services is presented. It is proposed suitable monitoring technique used for estimation of values of service parameters and allocation of communication and execution resources by means of service distribution. The paper also presents results of experiments performed in real environment that show effectiveness of proposed solutions.

Keywords—quality of services, service virtualization, service request distribution.

1. Introduction

For Web-based systems there are three attributes that directly relate to everyday perception of the quality of service for the end user: availability, usability, and performance. The performance issues of information systems are very widely explored in different contexts. For SOA-based systems solutions concerning the quality of services have been generally developed in the context of Web services, usually proposing useful standards for quality of service mechanisms, such as WS-Policy [1] and WSLA [2]. To support the quality of service delivery some selection of service algorithms are also proposed. For example in the work [3] the service selection based on utility function on attributes assigned to services (such as price, availability, reliability and response time) has been proposed. Most of these works assume that values of service parameters does not change dynamically.

On the other hand the quality of services can be considered in the context of the quality of the resource utilization. Among the others the virtualization is already being used as a common and proven way to decrease the overall hardware needs and costs, however still the hardware utilization is around 20% and storage utilization does not go above 60% [4]. Using virtualization gives very promising results, but as stated in [5] it is still not enough. Virtualization stopped and is not pushing forward. Mission critical services are used as before due to the easier maintenance, controlling and monitoring. What is more, reduced budgets made it much more complicated for real virtualization adaptation since - especially at the beginning – costs of implementation are higher than those of keeping everything as is.

In the paper the Virtual Service Delivery System (VSDS), a tool for efficient allocation of communication and execution resources to serve requests for synchronous services and service monitoring during its execution is presented. The service requests are examined in accordance to the SOA request description model. The functional and non-functional requirements in conjunction with monitoring of execution of services and communication links performance data are used for requests distribution and for resource allocation. At the lower layer virtualization is used to control efficient resource allocation to satisfy service requests.

The paper is organized as follows. Section 2 briefly describes the main ideas used during designing and developing presented in the paper Virtual Service Delivery System. In the Section 3 the main service quality issues are discussed. The architecture and functionalities of the VSDS are presented in Section 4. Section 5 describes the ways how service monitoring and evaluation the values of service parameters is done by the Broker and Virtual Server Manager (VSM), two components of VSDS. In the next section results of the first experiments performed on the implemented system are presented. Finally, Section 7 outlines the work and discusses the further works.

2. The Concept of Quality Aware Service Delivery

The concept of effective and quality-aware infrastructure is based on the idea of Virtual Service Delivery System capable to handle client's requests taking into account service instance non-functional parameters.

The main components of the system are network service broker (further called Broker) and Virtual Server Manager. They are built as a component of a SOA. The main assumptions for operation of both modules are:

- the Broker delivers to clients the set of J services (so called atomic services) $as_j, j \in [1,J]$,
- the Broker knows execution systems $es_m, m \in [1,M]$, where real services (service instances) are available,
- the Broker monitors execution of client's requests and collects the monitoring data,

– the Broker acts as a service proxy – it hides real service instances, and distribute client's requests for services to proper instances according to some distribution policy,

– the VSM is responsible for creation of service instances,

– the VSM is responsible for service execution,

– the VSM offers the access to hypervisor actions that is independent on any used virtualization system by using *libvirt* toolkit,

– the VSM offers information about particular physical servers as well as running virtual service instances.

– the VSM is responsible for monitoring of executed services and execution environments (servers) including running of servers virtual machines.

The Broker implements the Virtual Service Layer (VSL). The VSL (Fig. 1) virtualizes real services available on service execution systems (servers). The VSM manages virtualized computational resources. Both layers are defined as the tuple $< ES, CL, AS, IS >$. $ES = \{es_1, \ldots, es_m, \ldots, es_M\}$ is the set of execution systems es_m, where: $m \in [1, M], M$ – the number of execution systems. The execution systems can be placed at different geographic locations. $CL = \{cl_1, \ldots, cl_m, \ldots, cl_M\}$ is the set of communication links cl_m from the Broker to execution systems. The Broker delivers the set of J atomic services $AS = \{as_1, \ldots, as_j, \ldots, as_J\}$. Each atomic service as_j available at the Broker is mapped to one or more known instances that form instance subset IS_j. Instances of given atomic service can be localized at different execution systems es_m. $IS = \{IS_1, \ldots, IS_j, \ldots, IS_J\}$ is the set of all instances of services, where: IS_j is the subset of instances of service as_j, $is_{j,m}$ is the m-th instance

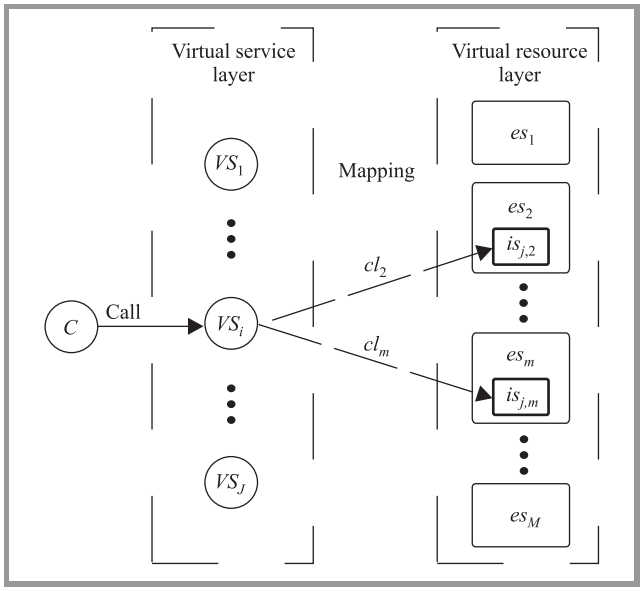


Fig. 1. The layers of Virtual Service Delivery System.

of j-th service as_j localized in given execution system and M_j is the number of instances of j-th service.

The real services are hidden from client point of view. The Broker advertises virtual services VS_j in accordance with SOA paradigm, and handles client's request for services. The client deals with virtual service (virtualized atomic service as_j) that can be executed at different locations.

The Broker collects essential data about service execution. It also monitors values of parameters of execution environment, i.e., communication links cl_m and execution systems es_m. The main advantage of virtualization of services is that according to values of service instance parameters some quality based policy of service delivery can be applied. The client of the system C calls the Broker for a service, and the Broker distribute the request to one, chosen service instance to ensure proper values of service quality parameters.

At the VRL the management of available execution systems at the lowest level is performed. VSM that implements VRL is responsible for efficient allocation of execution resources to services using virtualization techniques [6]. There are two aims of using VRL, which can, and in most cases would be, mutually exclusive. First of all the manager shall provision the instances of services with proper resources to ensure the fulfilment of requirements for requested service. Secondly it shall increase the utilization of the available resources, so that overall capacities are used to the highest possible degree. Managing the resources can be described in three distinct steps: provisioning of resources, adjusting and freeing the resources.

The largest difference between VSM and other similar solutions is coming from another targets standing behind our proposition. While most of other solutions are strictly devoted to manage the infrastructure, VSM is devoted to properly dispatch the requests, placing the virtualization management on the second place. Nonetheless one can point a number of similarities starting from common modular architecture with possibilities to customize the software easily. Furthermore just like other solutions *libvirt* is used to overcome the problem with communication with various hypervisors.

The role of VSM as a dispatcher means that some of the functionalities are redundant. Under such situation one may put offering Amazon compatible API, billing integration, number of control panels and so on. On the other hand the functionality is extended to understand the SOAP messages, identify which services are capable of performing them and finally running those services and dispatching the requests. Analogical software is found as an addition on top of OpenNebula and offers service orchestration and deployment or service management as a whole [7].

The instances of given atomic service are functionally the same and can differ only in the values of non-functional parameters $\psi(is_{j,m}) = \{\psi^1_{j,m}, \ldots, \psi^f_{j,m}, \ldots, \psi^F_{j,m}\}$, where $\psi^f_{j,m}$ is f-th non-functional parameter of m-th instance of j-th atomic service. Two kinds of service parameters may be distinguished: static parameters - constant in long period

of time (i.e., service price), and dynamic parameters – variable in short period of time, e.g. the completion time of execution of the service instance may be the case. From the client point of view, the very important service parameter is response time, which is usually quite variable parameter. In the network environment it consists actually of two components: data transfer time and execution time on the processing server (later called execution time for short).

3. Service Quality Issues

In order to assure proper quality of service delivery the three requirements are to be considered: effective and suitable service parameters monitoring and estimation, proper service request distribution according to current service parameters estimation and resource utilization, and service execution resources management.

The quality of network services depends on communication link properties and effectiveness of request processing on the server. Both affect the quality of each service instance separately and can be expressed by values of service instance non-functional parameters. To satisfy requested service parameters distribution of the request to proper service instance must be performed. The problem of service request distribution can be stated using criterion function Q

$$is_{j,m^*} \leftarrow arg_m(\psi^1_{j,m}, \ldots, \psi^f_{j,m}, \ldots, \psi^F_{j,m}) . \qquad (1)$$

It is the task to select such instance is_{j,m^*} to serve request for service as_j that criterion Q is satisfied. In the particular case it is the task of finding extreme of the criterion function.

To satisfy proper service instance selection the current values of each instance parameters should be known. This requires methods of estimation and/or forecasting of values of such parameters. The Broker uses two approaches: statistical methods based on time series analysis and method based on artificial intelligence approach – using fuzzy-neural network [8], [9] and monitoring of parameters characterizing execution environment. For both approaches the estimation of values of previous executions is required what is described in the subsequent sections.

On the basis of forecasted and/or monitored values of parameters several approaches to service distribution algorithms can be adopted. Generally, the fully controlled environment case and not fully controlled environment case can be distinguished. The first one refers to the use of dedicated links and VSMs in all execution systems. The second one is when there is no full control of communication links. It is the most common condition for delivery of service in the Internet according to SOA paradigm.

For the VSDS the best effort based algorithms for the uncontrolled environment are implemented by now. Two most commonly considered service parameters are used: data transfer time and completion time of service execution in the processing server. The selection of service instance for requested service as_j is performed according to criterion (2)

$$is_{j,m^*} \leftarrow arg_m min(T^{j,m}_{PROCESS} + T^{j,m}_{TRANSFER}) , \qquad (2)$$

where $T^{j,m}_{PROCESS}$ is execution time of service instance $is_{j,m}$ and $T^{j,m}_{TRANSFER}$ is data transfer time for fulfilling request for service instance $is_{j,m}$.

The distribution algorithms that takes under consideration the completion time of service execution require estimation and forecasting of values of these parameters. As mentioned above, the Broker uses time series analysis based forecasting or fuzzy-neural network based forecasting shortly described later.

4. The VSDS Components

The two main VSDS components, the Broker and Virtual Server Manager, are built as a components of a SOA-based system. The architecture of VSDS is very flexible and gives opportunity to compose the processes from services publicly or privately available.

The Broker handles SOAP requests for services. The virtual services provided by the Broker are described using WSDL (Web Service Definition Language) standard and are published in accordance to SOA paradigm.

The client's requests are analyzed and checked versus the information about possible places of execution as well as values of non-functional parameters of service execution at each location. In order to support evaluation of values of service instance parameters the Broker performs active monitoring of execution environment, i.e., values of parameters of communication links to execution systems (servers) and server state parameters. Execution system state monitoring is done with use of SOAP messages.

The above functionality is performed by the following modules of the Broker (Fig. 2):

- Controller – the main control unit performing service request distribution. It makes the decision on the basis of values of service instance parameters derived from Estimator/Predictor module;
- Service Monitor – monitors the execution of services at the TCP session level, and records the values of executed service parameters;
- Environment Monitor – makes active measurement of values of execution environment parameters – the server state and values of communication link parameters;
- Estimator/Predictor – the module which estimates values of essential parameters characterizing the service and instances with use of TCP session level data, and performs prediction of values of parameter on the basis of historic data and current values of environment parameters.

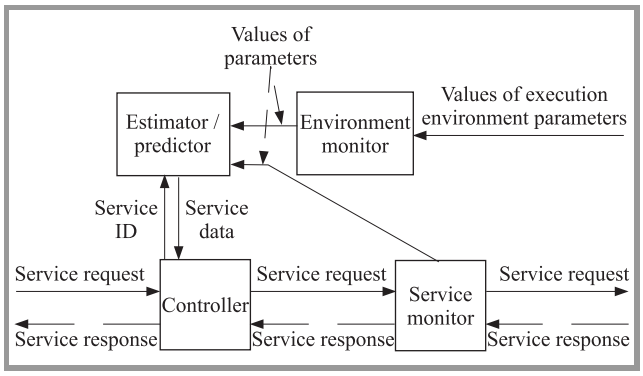


Fig. 2. The architecture of the Broker.

VSM offers two interfaces to interact with the virtualized environment. One is XML-RPC based that is used mainly for communication between internal VSM modules. More important is the possibility to direct SOAP calls to services to be handled by the VSM. Each and every request is then redirected to proper service instance based on the requirements it has. Proper instance is either found from the working and available ones or the new one is started to serve the request.

Such approach gives the possibility to manage the virtualization automatically with minimal manual interaction. The architecture of the VSM is presented in Fig. 3, as for now there is a number of independent modules offering the XML-RPC interfaces to interact with them.

- Manager – manages all other modules and routes the requests to the services,
- Virtualization Unit – offers the access to hypervisor actions, uses *libvirt* to execute commands what gives the independence from particular hypervisor,
- Database – module used to store monitoring data, images of available services (capsules) and information about available execution systems,
- Monitoring Unit – offers information about particular physical servers (execution system) as well as about available virtual service instances,
- Matchmaker – module responsible for the properly match the requirements of the request with capabilities of the environment and current state of it.

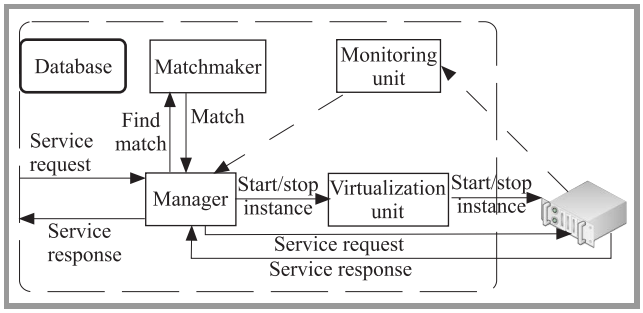


Fig. 3. The architecture of the Virtual Server Manager.

Each SOAP request coming to the system is directed to the Manager module which extracts requirements passed in the header section of the message to properly handle the request. Process of request handling starts with SOAP message coming from outside through the Broker that is an actor initiating the process service execution. This is in accordance to the general idea of placing VSM inside Service Oriented Architecture where Broker is common module for such purpose. The Manager module has a role of being the gateway to the system and hides all of the heavy lifting from outside world. It extracts the requirements passed in the SOAP message, in its header section. The requirements are extracted and converted to simple text form which is used by the Matchmaker module.

Handling the request can lead to one of three situations. There is a running service instance, which can perform it and it will be returned as the target to which the SOAP request shall be forwarded. There is no running service instance, but there is an image which satisfies the conditions. In such a case the image will be instantiated and it will be used as the one to perform the request. Last possibility is the lack of proper service and image in which case the error will be thrown and finally returned as a SOAP Fault message to the client.

As it was already mentioned, virtualization management is based on open source *libvirt* toolkit. It offers the virtualization API supporting number of the most popular hypervisors. From the point of view of this paper it is less important how technically the management is performed. It is more important to note what are the capabilities of the management and how it is understood here. The virtualization management does not mean simply to start or stop the virtual machine, it is much more complex problem. The complexity is coming first of all from the decision making problem. Firstly the correct service instance or service image should be found, by means of fulfilling specified in the service request requirements. In the case when the new instance of service has to be created the proper resources should be allocated and finally make processing as minimal footprint as possible.

Currently, using VSM the following features are available:

– mechanism for the creation and use of services – using the SOA paradigm and virtualization; this makes services independent from the available hardware architecture, and ensures the efficient use of hardware resources;

– method of delivery of services in a virtual machine environment – are taken into account the performance parameters of the virtual machine, service and equipment on which a virtual service instance is installed;

– tool architecture and its constituent modules is open, communication takes place via defined interfaces using XML-RPC for internal communication and the SOAP protocol for external communication.

5. Evaluation of Values of Service Parameters

5.1. Service Monitoring and Estimation in the Broker

The Broker performs request distribution based on the actual values of service instance parameters forecasted from the monitored and collected data of previous executions. The two basic service parameters, the data transfer time and completion time of service execution in the processing server, are obtained in two ways: with use of SOAP based cooperation between the Broker and the system which executes the service instance, and with use of the monitoring of TCP session which handles service request to the processing server.

In the first case the execution system must be able to interpret the specific additional data in Broker calls for service, and include additional specific data in service response. The execution systems controlled by VSM has such ability.

The Broker records the arrival time of each request for the service, the start time of call for service to the server which executes the service instance, and the time of end of processing of the service response from the server. The difference of the last two times establishes the total time of the request processing. The execution time of the service instance (processing time in the server) is delivered in the service response SOAP message. The service data transfer time is assumed as a difference between the total time of the request processing and service instance execution time. This time includes the time of resolving DNS address of processing server and all pre-transfer operations. However, these components of request intervals are measured by the Broker and can be excluded as described latter.

When cooperation between the Broker and the execution system is not possible, the values of essential service parameters are obtained with use of the analysis of the TCP session that handles the Broker's request for the service to the execution system.

The client request arrives at the moment t_{RA} (Fig. 4). The interval T_{DM} is the time of choosing the service instance (or server) that will process the request. Starting from this point the Broker measures the following time intervals of TCP session of call to processing server:

- the time of resolving DNS name address T_{DNS},
- the time of establishing TCP connection (TCP Connect time) T_{TCPC},
- the time to receive the first byte of transferred data from the server executing the service T_{FBYTE},
- the total time of the request processing $T_{SUM} = T_{FBYTE} + T_{DTRANS}$.

The Broker also records the number of sent bytes B_S and the number of received bytes B_R during the session.

It is assumed, that services are delivered using SOAP standard and the server responds after receiving all necessary

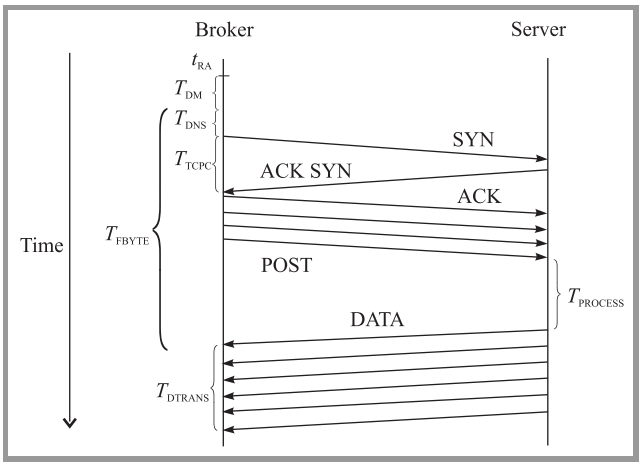


Fig. 4. The TCP session of handled client request.

data from the Broker. If we assume that transfer rate to and from the server are similar (what can be not true in general case), the service execution time $T_{PROCESS}$ can be evaluated with Eq. (3):

$$T_{PROCESS} = T_{FBYTE} - T_{DNS} - \frac{B_S}{B_R} \cdot (T_{SUM} + -T_{FBYTE}) - 2 \cdot T_{TCPC}. \quad (3)$$

Very often the request for service does not transmit other data in addition to those that fully identifies the service, so the time of this transmission can be neglected. Because the processing servers are registered in the broker earlier their IP addresses can be known, and the T_{DNS} time can be usually also neglected. In such case the service execution time $T_{PROCESS}$ is calculated according to Eq. (4):

$$T_{PROCESS} = T_{FBYTE} - 2 \cdot T_{TCPC}. \quad (4)$$

The data transfer time is the difference between the total time of the request processing and service execution time $T_{PROCESS}$. The service delivery time (for the client which requests the service from the Broker) includes also the decision making time T_{DM}, which according to distribution algorithm may be neglected or not.

In more general case some pre-transfer operations (i.e., SSL connect/handshake) must be also taken into account. The Broker can measure such operations too. It must be noted, that in case of lack of cooperation between the Broker and the execution system the estimation procedure is possible only when data transfer time from the Broker to the server is negligible, or data transfer rates to and from the server can be compared, or the data transfer time to the server can be measured separately.

Forecasting of values of service execution time and data transfer time for incoming requests is also performed in First, with use of time series analysis, i.e.:

- moving average of recorded times of previous executions:

$$\hat{t}^n_{j,m} = \frac{1}{L} \sum_{k=n-1}^{k-L} w_k \cdot t_k,$$

– moving median of recorded times of previous executions:

$$\hat{t}^n_{j,m} = med(t_{k-1}, t_{k-2}, \ldots, t_{k-L}),$$

where: $\hat{t}^n_{j,m}$ – forecasted time for n-th request served by service instance $is_{j,m}, L$ – the length of the observation window, w_k – window function, t_k – the times of previous requests served by instance $is_{j,m}, n$ – the index of current request.

When cooperation with the server is possible, i.e., when VSM is applied, the evaluation of service transfer and execution times can be performed with use of the concept of fuzzy-neural controller built with use of 3-layered fuzzy-neural network [8]–[11].

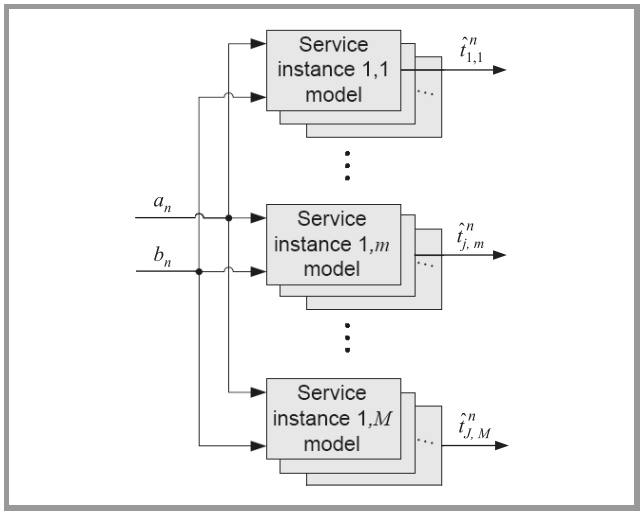


Fig. 5. Modeling service instances as fuzzy-neural controllers.

The fuzzy-neural controllers model each communication link and each service instances separately (Fig. 5). The adaptive model, described in detail [9] evaluate the output value (transfer or execution time) using two input values of parameters characterizing communication link or execution system. The input of the communication link model is, by now, link throughput of sample value of data downloaded from execution system, and link latency (namely TCP Connect Time), both derived with use of measurements and time series analysis. The input of service instance model can be any of monitored parameter by VSM or the counted number of being processed calls in the server.

5.2. Service Execution Monitoring in Execution System

The VSM is equipped with Monitoring Unit that is able to collect different parameters related to service execution and state of execution systems, depending on Broker request. Monitoring agent shall accompany with any currently active service. The frequency of measures or agreed values of attributes are present in the contract thus the agent shall simply check if the operations are done accordingly.

To ensure efficient resource utilization, incoming requests to VSM are attributed to execution classes. The functional and non-functional requirements are considered. The VSM exploits the combining the service orientation with automatic management using constraints attached to the requests what increases overall reliability, response time and constraints fulfilment, reducing the need for manual work in the same time.

Currently two different ways of performing monitoring can be in use. First solution based on using Munin, a tool, which is used to monitoring service execution and state of execution systems. Unfortunately this approach although highly efficient does not allow direct monitoring of virtual machines used for service instance execution. It's why it was decided to introduce an alternative way of monitoring. The second available solution based on using Xen-stat, which is an integral part of the package Xen virtualizer. Using Xen-stat allows to monitor the server and each virtual machine at intervals specified by the administrator. In particular, it is possible to monitor:

– CPU consumption by each virtual machine individually,

– CPU usage on the server,

– RAM memory usage of each virtual machine individually,

– RAM usage on a server.

For storing monitoring data simple MySQL database is used. Implemented database consists of three tables:

– measurements – contains information about the virtual machine load,

– capsules – contains data about virtual machines,

– servers – contains information about the server and its current load.

To collect and record information about the system load special module implemented in Perl is used. The module is run every minute and takes measurements at intervals set by the administrator. Data from the monitoring of virtual machines and servers available are displayed by Xentop command, and then the results are parsed and stored into a database. Then it is possible to visualized the results of the monitoring of resource usage by running virtual machines using Xen Graph tool. Monitoring data collected by the VSM in the local database are also available to the outside through accepted by the VRM SOAP messages.

Concluding current solution is very flexible because depending on the needs of monitoring the system usage and service execution gives the opportunity of using two different tools – Munin and Xen-stat.

6. Effectiveness Tests

In the preliminary experiments the selected proposed solutions were tested in real environment – in the Internet.

The broker has been implemented as fully operational tool in Java technology. It supports all described functionalities and serves its services in accordance with SOA standards. The experiments has been focused on testing usefulness of monitoring and evaluation of service instance execution time, however data transfer time was also tested.
The Broker served a number of clients requesting fixed set of network services from five servers located in different countries of Europe. There were established six test services. Each service was running on each server giving a total of 30 service instances. The services generated different amount of data to transfer from 50 to 200 kilobytes.

Service instances were set different values of non-functional parameters. On each server machine the services run in www server with established maximum number of parallel threads for serving clients requests. Each service instance was implemented in that way that have had minimal time of processing not including any server service handling overhead (e.g., queuing delay). The service differed in basic execution time with one another and the service instances of the same service differed in basic execution time depending on instance location. The times varied from 2 to 6 seconds.
The clients which requested services and the Broker were located at Wroclaw University of Technology campus. The servers that run service instances were located in five different countries:

- planetlab2.rd.tut.fi, (193.166.167.5), Finland,
- ple1.dmcs.p.lodz.pl, (212.51.218.235), Poland,
- planetlab1.unineuchatel.ch, (192.42.43.22), Switzerland,
- planetlab4.cs.st-andrews.ac.uk, (138.251.214.78), UK,
- planet1.unipr.it, (160.78.253.31), Italy.

The research scheme was the following:

- the clients requested all six services in a round-robin fashion, each client in a different order,
- the number of clients increased from 0 to 80 during 3 hour test,
- the requests were distributed by the Broker according to round-robin algorithm,
- it were measured essential moments of each TCP session handling requests for services, and recorded service instance execution times received in server responses,
- for each request the fuzzy-neural controller calculated forecasted value of service instance execution time and data transfer time,
- estimated and forecasted times were compared against measured ones.

All requests transmitted no additional data to the server. It was assumed that the true real values of service parameters were: service instance execution time measured in the server $T_{PROCESS-REAL}$, and the difference of the total time of request processing T_{SUM} and the service instance execution time measured in the server $T_{SUM} - T_{PROCESS-REAL}$, assumed as real value of data transfer.
The test showed that none of five servers were overloaded by requests for services. Figure 6 shows Mean Absolute Percentage Error (MAPE) calculated for estimation of service execution time $T_{PROCESS}$, using monitoring of TCP session only. The figure shows MAPE for all 30 instances grouped in the following manner: first six instances from first server (each of different service), next six instances from second server, and so on.

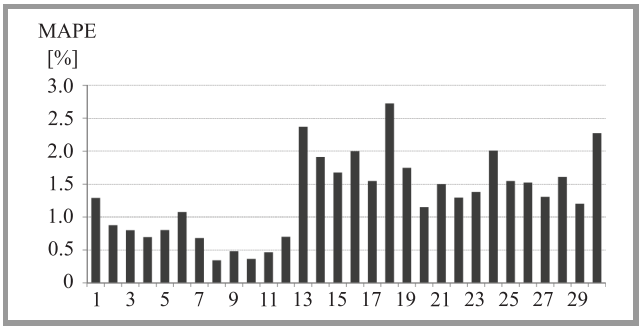


Fig. 6. The MAPE of $T_{PROCESS}$ time estimation for all service instances.

The total MAPE for all estimation is very good, and is equal 1,31%. It is interesting to see the visible difference of error level for particular servers. For server 2 (instances 7 to 12) the total MAPE is the smallest and is 0,51%. For server 3 (instances 13 to 18) the total MAPE is the largest and is 2,04%. This could be caused by different server queue thresholds (the number of requests processed in parallel), however should be thoroughly examined.
Figure 7 shows effectiveness of forecasting service instance execution times $T_{PROCESS}$ with use of fuzzy-neural controller. The total error of prediction MAPE is equal 1,32%. This is very good value. However, it must be noted that experiment was performed for stable server operation. In this case no obvious dependency of error level on particular server is visible.

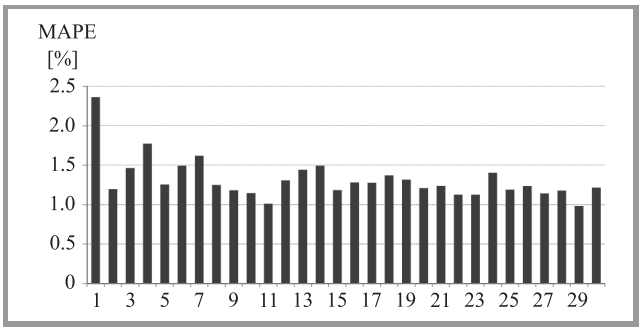


Fig. 7. The MAPE of fuzzy-neural forecasting of $T_{PROCESS}$ time for all service instances.

The forecasting of data transfer times were performed using moving average method and with use of fuzzy-neural controller. The forecasting errors were much greater than forecasting execution times. The total MAPE for fuzzy-neural controller was 13.5% (for weighted moving average even greater, about 19.3%).

It was due to the fact, that very small amount of data was transmitted and there were short transfer times – they varied from about 50 ms to hundreds of milliseconds. At the same time it was found that one of the link (to server 2 – service instances 7 to 12) was apparently problematic and significantly expanded total error as shown in Fig. 8. It must be noted that preliminary experiment was not focused on testing fuzzy-neural controller for data transfer time forecasting, and learning parameters of the controller were not tuned too. When this conditions will be met the better forecasting is expected.

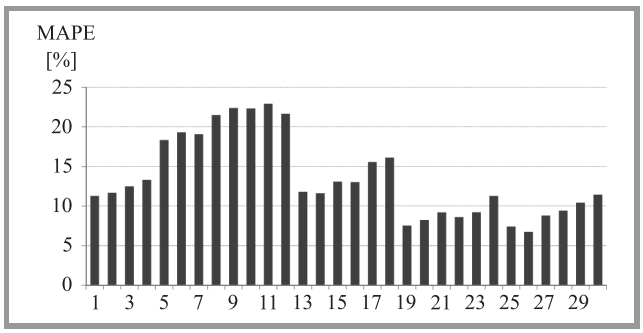


Fig. 8. The MAPE of fuzzy-neural forecasting of data transfer time for all service instances.

To test the performance of the VSM the test environment has been created. Most of the code has been written in Python (modules), sample services as well as sample requests are generated using Perl. Tests in current state has been limited to the CPU usage. The services are dummy and all they do is to utilize certain capacity of the CPU during certain amount of time. The solution to make this happen is somewhat not exact and limits the usage properly when set above 5%. It uses simple method to increase the CPU usage to 100% forking Perl processes and starting never ending loops. Such process would always consume all of the processing power so it is being limited using CPUlimit tool.

The number of CPU's to be stressed is fixed in the code, but it is not a problem to pass this number as an argument to the script. Time and CPU usage are limited in the directly in the dummy services implementation. It is done by first starting the script as a new process. It's ID is passed to the CPUlimit tool with desired usage, e.g., 50% and as the last step after desired number of seconds the process is killed to free the resources.

The first performed experiments using testing environment, in case when VSM is lack of automatic resource freeing are very promising. During these experiment called service is configured to consume 80% of CPU for 60 seconds what simulates some relatively exhausting operation. The requests are incoming almost in the same time and they require at least 30% CPU capacity to be free. The outcome of such a simple simulation is increasing number of instances of the service to handle sudden peak in requests, yet there is no mechanism to limit the number of the instances afterwards.

7. Conclusion

Quality of services in Service Oriented Architectures yields a number of issues which involves suitable monitoring and estimation of values of service parameters, distribution of service requests to selected execution systems running instances of services, forecasting values of service parameters and virtualization management. Automation of such process requires well designed architecture and procedures of service quality aware system.

The presented solution for solving all mentioned problems are still under study. First experiments are very promising. The effectiveness of service request distribution algorithms depends on precise evaluation of values of service instance parameters. The two basic ones, execution time and data transfer time, are the key to satisfy Quality of User Experience (QoE). The experiments showed that the evaluation of execution time is very good. However, the case of sending large amount of data from the client must be also tested. The evaluation of data transfer time is to be more explored and requires well prepared extensive experiments.

It is worth to note that presented solutions are implemented as fully operational tool ready for use in real environment that applies SOA standards and Simple Object Access Protocol (SOAP).

Acknowledgment

The research presented in this paper has been partially supported by the European Union within the European Regional Development Fund programs no. POIG.01.01.02-00-045/09-00 and POIG.01.03.01-00-008/08.

References

[1] D. Box *et al.*, "Web Services Policy Framework (WS-Policy)", 2003 [Online]. Available: http://public.dhe.ibm.com/software/

[2] A. Keller and H. Ludwig, "The WSLA framework: specifying and monitoring service level agreements for web services", *J. Netw. Sys. Manag.*, vol. 11, no. 1, 2003.

[3] L. Zeng, B. Benatallah, A. Ngu, M. Dumas, J. Kalagnanam, and H. Chang H, "QoS-aware middleware for Web services composition", *IEEE Trans. Softw. Engin.*, vol. 30, no. 5, 2004.

[4] P. Sargeant, "Data centre transformation: How mature is your it?", 2010 [Online]. Available: http://www.gartner.com/it/

[5] B. Snyder, "Server virtualization has stalled, despite the hype", InfoWorld, 2010 [Online]. Available: http://www.infoworld.com

[6] D. Rosenberg, "Analyst: Virtualization management key to success", 2010 [Online]. Available: http://news.cnet.com/8301-13846_3-10468343-62.html

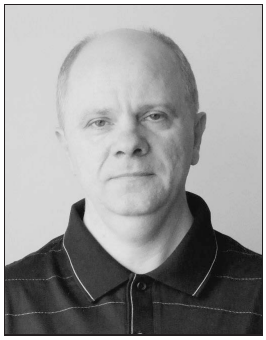
[7] P. Sempolinski and D. Thain, "A comparison and critique of Eucalyptus, OpenNebula and Nimbus", *C*, in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Technol. Sci. CloudCom 2010*, Indianapolis, USA, 2010.

[8] L. Borzemski, A. Zatwarnicka, and K. Zatwarnicki, "Global distribution of HTTP requests using the fuzzy-neural decision-making mechanism", in *Proc. 1st Int. Conf. Comp. Collective Intelligence*, Lecture Notes in AI, Springer, 2009.

[9] M. Fras, A. Zatwarnicka, and K. Zatwarnicki, "Fuzzy-neural controller in service request distribution broker for SOA-based systems", in *Proc. Int. Conf. Computer Networks 2010*, A. Kwiecien, P. Gaj, and P. Stera P., Eds. Berlin, Heidelberg: Springer, 2010.

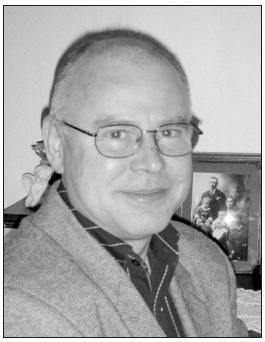
[10] L. C. Jain and N. M. Martin, *Fusion of Neural Networks, Fuzzy Sets, and Genetic Algorithms: Industrial Applications*. CRC Press LLC, London, 1999.

[11] E. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis", *IEEE Trans. Comp.*, vol. C-26, iss. 12, 1977.



Mariusz Fraś received M.Sc. in Electrical Engineering in 1989 and in Computer Science in 1991, both in Wrocław University of Technology. In 2004 he received Ph.D. in Computer Science in Institute of Informatics, Wrocław University of Technology. Since 2004 he works as an Assistant Professor at the Faculty of Computer Science and Management, Wrocław University of Technology. His area of scientific interest includes distributed processing and Internet services. His main area of interest is parallel and distributed processing in computer network environment, the quality of network services, and Internet research and the performance of Web services.

E-mail: mariusz.fras@pwr.wroc.pl
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland



Jan Kwiatkowski received M.Sc. and Ph.D. in Computer Science from the Institute of Technical Cybernetics, Wrocław University of Technology at 1977 and 1980, respectively. Since 1980 he works as an adjunct at the Faculty of Computer Science and Management, Wrocław University of Technology. In the years 1987–1998 he acted as a deputy director responsibly for education. From 2002 to 2004 under sabbatical leave, he worked as associate Visiting Professor at Math and Computer Science Department at the University of Missouri, St. Louis. Since 2007 he acts as Computer Science and Management Faculty Dean representative responsible for foreign students, currently as a Dean's Plenipotentiary for International Relations. His area of scientific interest includes software engineering and parallel processing. He is mainly interested in parallel and distributed software design process, performance evaluation of parallel programs and cluster/grid computing.

E-mail: jan.kwiatkowski@pwr.wroc.pl
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Paper

Comparison of Resource Control Systems in Multi-layer Virtual Networks

Bartłomiej Dabiński, Damian Petrecki, and Paweł Świątek

Institute of Computer Science, Wrocław University of Technology, Wrocław, Poland

Abstract—This paper describes the performance of various methods of QoS assurance for each connection in an environment composed of virtual networks and dedicate end-to-end connections inside them. The authors worked on the basis of research conducted with the use of the authorial network management system named Executed Management, which uses resources virtualization platforms VMware and Mininet for testing purposes. We briefly describe our system and techniques we used and some alternatives we tested and discarded because of their limitations. Functionality and performance of proposed solution to widespread implemented mechanisms as OpenFlow and MPLS are compared. Reasons for selecting well-known techniques to isolate networks and limit bandwidth on different levels of virtualization are considered. The purpose of this paper is to show out our studies and performance we achieved.

Keywords—Execution Management, MPLS, network virtualization performance, OpenFlow, parallel networks, virtual networks, virtual networks performance.

1. Introduction

The purpose of the project was to build a system that configures network in order to satisfy QoS requirements for large number of applications. The applications run in multiple virtual, isolated networks that share a single physical network. The system has to create connections inside these networks, with specified paths and guaranteed bandwidths, dynamically in response to applications' requests. It means that the system must be aware of network content. Only well proven and commonly implemented algorithms, protocols and techniques were to be used because the system should operate in the network built with generic equipment. This paper describes selected resources virtualization method and compares its performance to solutions providing similar capability.

Many techniques may have been applied in order for above mentioned goals to be achieved. First step of desribed work was to compare features of Multiprotocol Label Switching, IEEE 802.1ad (QinQ), IPv6-in-IPv6 tunneling and Provider Backbone Bridging. The authors decided to use VLANs to isolate virtual networks and to run dedicated virtual or physical machines for handling ISO OSI L3 networks inside these L2 networks. Then we decided to build the centralized system to manage network resources. The task of the system was to handle requests of applications and to create dedicated tunnels inside virtual networks in reply to the requests. Each connection between end nodes and each virtual network must have guaranteed bandwidth. A variety of traffic engineering methods were tested and the shaper of *tc* linux application was selected. The traffic of an end-to-end connection is limited in virtual interfaces (end-to-end tunnel entry points) by TBF classless queuing discipline. The traffic of virtual networks is limited in physical interfaces that send traffic into the network. u32 classifier filters packets belonging to a specific VLAN and the traffic is enqueued to the HTB classes to limit its bandwidth. Furthermore, the system ensures uninterrupted transmission without delay variation in case of tunnel path or bandwidth modification.

The laboratory network was built using VMware ESXi server (Debian hosts run in virtual machines), Mininet service [1], [2], Juniper EX4200 switch and two MikroTik RB800 devices. Mininet was designed as a network emulator for testing OpenFlow but we made some modification to make Mininet meet our needs. MikroTiks was used as a precise bandwidth and latency testing tools.

The authors experimented with the performance of the system both within the scope of delays in connection handling (creation, removal, modification) and within the scope of QoS ensuring, which means guaranteeing bandwidth requested by user's applications and preserving low packets flow latency in case of existing connections reconfiguration. In order to point advantages and disadvantages of proposed solution its functionality was compared to widespread implemented mechanisms like OpenFlow and Multiprotocol Layer Switching (MPLS).

In order to choose the best solution capability of MPLS, QinQ and Provider Backbone Bridging (PBB) along with suitable traffic shaping and policing techniques was analyzed, before we decided to modify the solution described in [3], [4] to meet our requirements. This solution uses VLANs and IPv6-in-IPv6 tunneling.

MPLS was dismissed due to the fact that most of its implementations in modern network equipment do not fully support IPv6. For deployment of MPLS and IPv6,

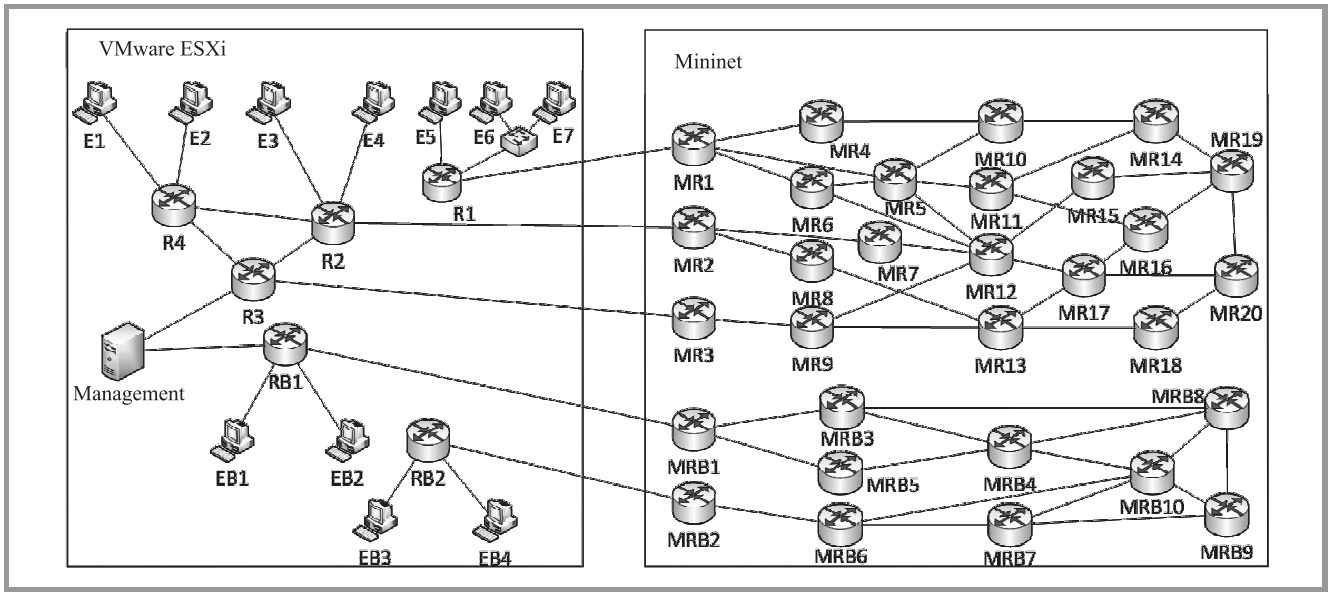


Fig. 1. Testing topology.

the protocol stack has usually to include following elements: MPLS, IPv4, VPLS and then IPv6. Furthermore, MPLS does not solve the problem of two applications, when each of them needs an end-to-end connection with the same pair of hosts using different QoS requirements. On the other hand, MPLS was a strong candidate because it offer simplest, fully automated configuration of the end-to-end connection. It does not require our system connect to and configure each node on the path separately.

QinQ seemed to be easy in implementation due to widespread support of this standard but it has several limitations, which are crucial in the context of this work. The most important disadvantage is a necessity to block communication of the host with the entire network, while the host is connected to another host. It is an implication of the requirement that end-to-end connection must be isolated and we cannot require end hosts to implement QinQ. Then, it is impossible to fulfill these conditions and handle separate, concurrent end-to-end connections for multiple applications on a single host.

Provider Backbone Bridging might perform well in the core network but it is Layer 2 protocol, so it is difficult to distinguish between multiple applications on a single host. Furthermore, PBB is a novel, advanced standard and unsupported by most of devices available for our research. We tried to used PBB to our purposes but it has almost all MPLS disadvantages and adds some more because of lack of support in general equipment.

Chosen Virtualisation Method for Isolated Parallel Networks

For creating parallel networks VLANs was used. End nodes were connected to specified VLANs and thus they can communicate only within definite part of the network. That end nodes belong to a single virtual parallel network unless they have multiple network interfaces. End nodes in different networks may have a common physical gateway but traffic intended for specified virtual networks have to be forwarded to proper virtual gateways. The end nodes can also be connected to external, unmanaged networks without QoS warranties (such as GSM). In above mentioned case we assume that first managed by us router on the connection path is the connection gateway and that node receives traffic with proper VLAN tagging. All physical gateways forward traffic to proper logical gateway by VLAN tagging performed on switch to which end nodes are directly, physical connected. The use of physical router for serving the virtual network is possible in the case when it has routed VLAN interfaces or deal with only one virtual network. It is strongly discouraged except the situation when it is not possible to tag packets that come to the router from end nodes.

The virtualization of the level 1 of the core network looks similar. The packets that belong to a particular network are sent from a logical gateway and are tagged by a physical node that hosts the logical router. In the next physical node analogous actions are performed. Based on tags, received traffic is forwarded to appropriate virtual router, which serves a particular network. In this way networks isolation is assured and delegation of virtual network management is possible. Network managing for external entities such as clients that would buy one virtual network can be delegated and have full control of virtual routers in theirs network.

There is also possibility to create a section of the logical network (core network, edge routers and end nodes) on a single physical device by intentional configuring virtual logical connections between virtual machines, with the use of resources virtualizer. As you can see in Fig. 1, we use

that to create testing purpose network with two physical hosts only.

When configuring the network, it is required to use a dynamic routing protocol (in our case it is OSPFv3) both between physical nodes and within virtual networks for logical nodes. It ensures full reachability of all the hosts and also communication for the management system and the nodes on the all virtualization levels.

The virtual networks of first level in the testing topology are built with the use of VMware ESXi virtualization platform, physical host with Mininet software, Juniper EX4200 switch and two MikroTik platforms RB800. ESXi served for virtualizing several access networks, end users' machines and virtual management server. Mininet was used in order to create two virtual backbone networks made of routers running Ubuntu operating system. The MikroTik devices were connected as physical end nodes for simulating users' computers and generating traffic for benchmark. The MikroTiks, through the Traffic Generator tool, which is built in RouterOS, collected statistics from transmitted/received data: packet loss rate and packet latency distribution. VLANs indicate which logical routers from access networks (ESXi) are allowed to communicate with specific logical routers in the core network (Mininet). In order to control Mininet host's incoming traffic, we had to modify Mininet scripts and create in this way properly defined virtual network with willful assignment of interfaces, addresses and connections. The Juniper EX4200 switch connects physically (Gigabit Ethernet) and logically (VLANs) the EX4200 machine and the host with Mininet. The testing topology consists of two parallel networks, whose only common point is a management node, which must be able to communicate with both virtual networks. The topology is presented in Fig. 1.

2. Network Virtualization

2.1. Chosen Method of End-to-end Connections Virtualization

End nodes operate in specific parallel network, so they can communicate with all other devices in this parallel network. At the time when they need to communicate, they do not start transmission directly but one node sends request to the Execution Management server. The management system analyzes QoS requirements for the new connection and available network resources and then it creates a proper IPv6-in-IPv6 tunnel. At the end of this process the application is informed that dedicated connection was established and it can start transmission. Virtual connections of second level are tunnels, which are dynamically created for each application use case. These tunnels are set on edge routers – the gateway routers for end nodes inside parallel networks. We do not impose any (non-standard for a generic IPv6 network node) requirements to end nodes owing to the approach in which traffic is directed to the tunnel by the mechanism implemented entirely in edge routers. This mechanism bases on packet filtering by following IPv6 and TCP headers fields: source address, destination address, source port, destination port. It is important that a computer, in case of having multiple active IPv6 addresses, uses for communication only the IPv6 address that was included earlier in the connection establishing request.

In order to preserve integrity of the network, the tunnel is configured on virtual dedicated interfaces, which are subinterfaces of loopbacks on edge routers. The addressing schema within tunnels is calculated with the use of 48-bit subnet divided into 126-bit mask subnets. The external addresses of the virtual interfaces use reserved pool of routed addresses. It is worth to mention that authors used the potential of IPv6 addressing which offers enough addresses to assign separate addresses for each parallel network. The external addresses of the tunnels are used to route the packets of its tunnel through the fixed path. Owing to this approach we are able to control tunnel path easily, by applying static routing based on destination IPv6 only. It is significant advantage since we can use any generic router in the core network and we do not cause high CPU utilization.

2.2. Limiting of Network Resources

The first part of network resources managing was to limit the bandwidth of entire parallel network. Except assigning resources for particular networks, it was necessary to reserve some bandwidth for management traffic and OSPFv3 packets. The mechanisms built into networking hardware to limit bandwidth of an interface and a software tool that implemented HTB algorithm was used, in which appropriate classes based on VLAN tags were created. Then bandwidth limits with these classes was associated.

It is advised for an administrator of an isolated network to use traffic shaping or policing mechanisms for isolating the class of traffic for signaling purposes. In order to achieve this, the mechanism that classifies all the traffic that is not an IPv6-in-IPv6 tunnel (by checking corresponding IPv6 header field) and guarantees bandwidth for this class is proposed.

The remaining traffic belongs to IPv6-in-IPv6 tunnels. Due to the use of dedicated virtual network interfaces in edge routers bandwidth of given tunnel can belimited simply by one classless *tc qdisc* discipline (e.g., by recommended Token Bucket Filter – TBF) configured on this virtual tunnel interface. This approach does not require traffic classifying. Centralized management of dedicated connections, in the networks which resources is known, guarantees that all the connections have sufficient resources and thus QoS is ensured. If a new connection would exceed the capacity of the network, the system simply rejects the connection and informs requesting application about the reason.

The variant possibilities of tunnel bandwidth limiting are briefly described in Section 5.

3. Management System

The management system, which was used for testing, was built especially for such purposes. It is made up of the Execution Management module, a database, and a supporting QoS module. The database stores i.a. the state of the network, which is data concerning all the nodes and connections on all the levels of virtualization. This is used to find an appropriate path for a new connection.

The system receives requests via a web application (that was built for an administrator for network managing purposes) or directly from applications via XML messages. For communication with the management system a dedicated traffic class with guaranteed bandwidth for signaling is used. In order to make an application work independent on the network management system, it is possible to use intermediary layer. An example of such deployment is described in [4], [5]. After receiving a request, the system verifies it and sends a query for an eligible path to the module that finds a path satisfying QoS. Then the system receives a reply and prepares scripts, which are subsequently sent parallel to network devices in order to configure proper services.

The connection path may be modified in the case of increased QoS requirements or when given connection have to be released because of a resources rearrangement. Such situation occurs when a new connection must not be established unless the resources already occupied by another connection are released. This connection may be routed another path that also satisfy its QoS. This process is fully automated and is imperceptible for the user. The process has following steps: creation of a new tunnel with new addresses and a new path, redirection of the traffic from the old tunnel to the new tunnel, removal of the old tunnel. The connection may also be removed in response to a removal request from the application, which used this connection.

More about Execution Management and cooperating modules can be found in [6].

4. Performance Evaluation

4.1. Execution Management Performance

The performance of the management system is an important factor influencing QoE because it determines the time of connection establishment. Therefore, it was crucial to comply with ITU-T recommendations [7].

The tests were performed to determine the behaviour of the system for a large number of queries and under different load (from 1 up to 100 requests). As expected, the system was stable and retained full data integrity, even in the case of multiple simultaneous requests for the creation, modification, and removal connections. Each series of tests were performed using 100 requests for a tunnel creation, 100 requests for removal and 50 requests for modification. Test results are averaged over the repetition of each series (1, 10 or 100 simultaneous requests) 50 times. Requests for testing were randomly generated in network with 40 end nodes to test system behaviour in large network. All time values showed in Figs. 2–4 are measured in miliseconds.

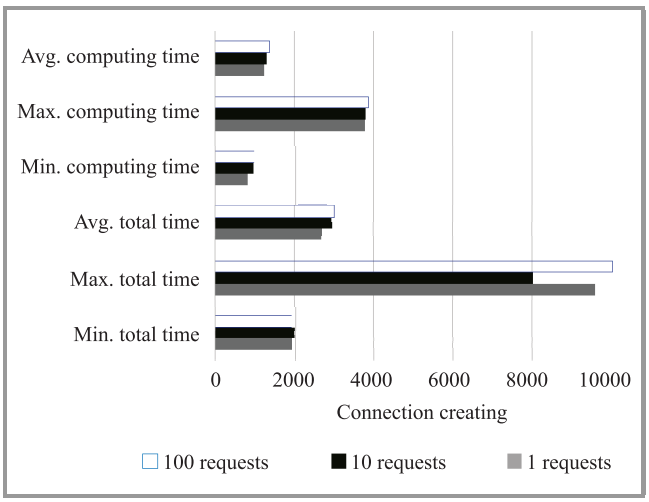


Fig. 2. Execution Management engine performance 1.

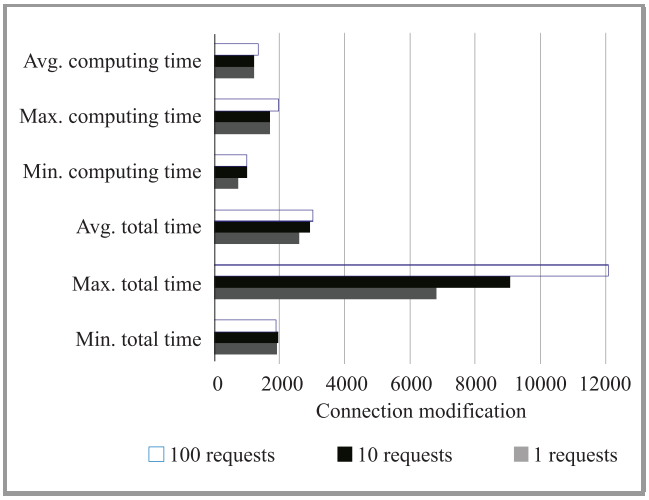


Fig. 3. Execution Management engine performance 2.

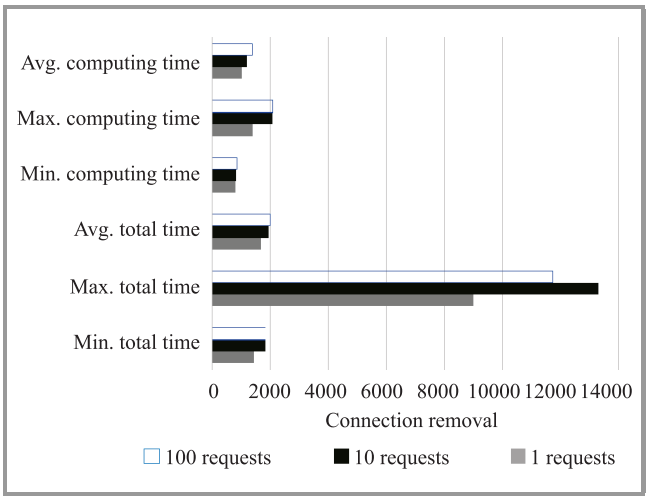


Fig. 4. Execution Management engine performance 3.

Approximately 80% of the processing time in the case of creation or modification, and up to 92% of the removal time is consumed by database operations. Albeit these numbers are significant, the result is much better when comparing to the previously used external database implementation. It probably can be further improved by creating our own library for handling database queries created by EM.

The total execution time depends primarily on the status of the network rather than on the performance of the virtualization tools. In some cases the times were very high due to temporary high load of Mininet host network operator or VMware Server. In such situations, it dramatically lengthened the waiting for SSH connection to the nodes.

Tests exclude situation when two requests require a connection via SSH to the same node. In this case, the waiting time for connection can be extended up almost double. This is due to the sequential SSH connections handling by a device. If there is a need for multiple connections to the same device in a single request (e.g., to configure two static routes or route and tunnel), all commands to be sent are combined in one. However, there is not a mechanism responsible for this in the case of multiple requests that have to be configured on the same node because it would starve scripts from earlier demands by continually appended subsequent commands.

There is not necessary to run the tunnel removals in parallel because the latency caused by removal process is not significant. Then each network is removed sequentially.

4.2. Performance of Network Mechanisms

This part of the chapter concerns the performance of implemented virtualized network environment. For testing MikroTik Traffic Generator tool was used. This is an advanced tool built into RouterOS that allows to evaluate performance of DUT (Device Under Test) or SUT (System Under Test). The tool can generate and send RAW packets over specific ports. It also collects latency and jitter values, Tx/Rx rates and counts lost packets.

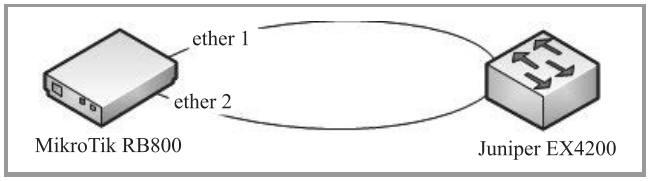


Fig. 5. Topology for testing MikroTik RB800 and Juniper EX4200 switch.

The goal of the first experiment was to examine the throughput of MikroTik RB800. We built the scenario as shown in Fig. 5. One stream of packet to examine unidirectional maximal throughput in half duplex (when network works stable) and two parallel streams of packets to examine bidirectional maximal throughput (full duplex) were used. The real maximal transmission rates were higher but connection was unstable. There were high packets losses and high latency. We arbitrary assumed that acceptable packet loss is 0.1%. Transmissions that have more packet loss were not considered. Results are presented in Table 1. The values are rounded down to nearest 5 Mbit/s. Each test lasted for 90 seconds.

Table 1
Results of performance tests of MT RB800 and Juniper EX4200

Direction	Packet size [bytes]	Throughput [Mbit/s]	Latency
ether1→ether2	1500	980	Min: 109 μs Avg: 6.5 ms Max: 11.5 ms
ether1→ether2 ether2→ether1	1500	760	Min: 32 μs Avg: 255 μs Max: 1.35 ms
ether1→ether2	100	240	Min: 23 μs Avg: 62 μs Max: 773 μs
ether1→ether2 ether2→ether1	100	100	Min: 21 μs avg: 64 μs Max: 883 μs

The subsequent test concerned the performance of the laboratory virtual networks. The network with two MikroTik RB800 platforms connected was used as shown in Fig. 6. The logical topology is presented in Fig. 7. Two RB800 devices are used since single RB800 could be a bottleneck in some cases. The results are presented in Table 2.

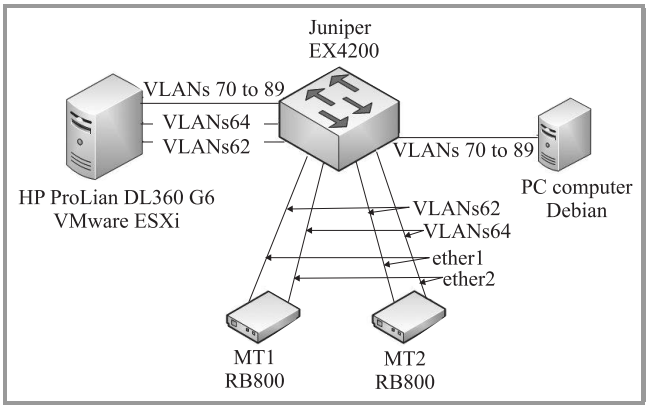


Fig. 6. Physical topology for tests.

The results show that the performance of examined virtual network is comparable to the maximal throughput of physical devices when transmitted packets are large (1500 bytes, which equals MTU size). The significant difference in the case of two-way traffic is caused by the fact that the traffic that flows in one direction passes the physical link between ESXi and Debian twice. Thus the real throughput in this link is doubled.

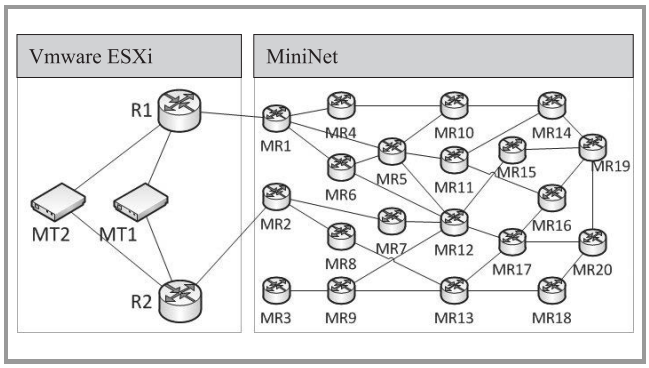


Fig. 7. Logical topology for virtual environment tests.

Table 2
Results of virtual networks performance tests

Paths	Packet size [bytes]	Through-put [Mbit/s]	Latency
R1→MR1→MR5→ MR12→MR7→ MR2→R2	1500	840	Min: 471 μs Avg: 1.9 ms Max: 15.2 ms
R1→MR1→MR5→ MR12→MR7→ MR2→R2 R2→MR2→ MR7→MR12→ MR5→MR1→R1	1500	420	Min: 397 μs Avg: 2.2 ms Max: 13.5 ms
R1→MR1→MR5→ MR12→MR7→ MR2→R2	100	40	Min: 107 μs Avg: 253 μs Max: 8.14 ms
R1→MR1→MR5→ MR12→MR7→ MR2→R2 R2→MR2→ MR7→MR12→ MR5→MR1→R1	100	40	Min: 110 μs Avg: 354 μs Max: 10.4 ms

In the case of small packets (100 bytes) the difference is much bigger. It is caused by the fact, that this testing scenario involves much higher packets per second rates and each packet has to be served by each virtual router (7 times) so the CPU performance of the virtualizers is the bottleneck.

The second parts of tests concern the IPv6-in-IPv6 tunnel and bandwidth limiting mechanism for this tunnel. The logical topology of this scenario is presented in Fig. 8. The bold lines indicate the tunnel path. The routers R1 and R2 are tunnel entry/exit points. Table 3 presents the test results. The maximal packets used for these tests were smaller comparing to the packets in previous tests because the MTU of a tunnel is decreased by the additional IPv6 header (40 bytes). Nevertheless, the presented throughput does not involve these additional 40 bytes of data. With default configuration of IPv6-in-IPv6 tunneling in Debian 6, the displayed MTU of a tunnel interface is 1460 but the real MTU is 1452 bytes. These missing 8 bytes was reserved for an encapsulation limit extension header, which was confusing since this extension header was not transmitted. We explicitly disabled encaplimit in order to transmit full 1460 bytes packets.

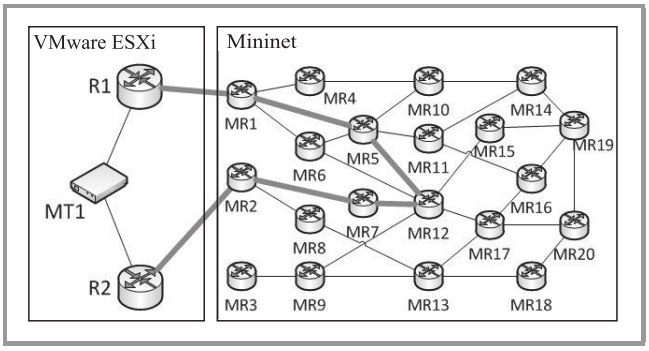


Fig. 8. Logical topology for tunnel testing

The *Transmission rate* column of the Table 3 means the fixed rate at which the MikroTik was sending data. *Tunnel bandwidth* means the value of tunnel bandwidth limit. *Throughput* is the rate at which packets were coming back to the MikroTik (after passing the tunnel).

The results presented in Table 3 are comparable to the results presented in Table 2. It means that implemented by us mechanisms does not introduce significant overhead to packet processing. The drop of the throughput is most noticeable in case of small packets and it is about 12.5%.

5. Bandwidth Limiting Methods

5.1. Limiting Bandwidth of Virtual Network

VLAN interfaces were chosen for implementation of parallel networks, which determined the methods of bandwidth limiting that might be used. It was important to limit bandwidth of virtual networks from outside of these networks. This implicates that virtual networks administrators do not need to take any action to limit bandwidth of their network. Because the limits are on different level of virtualization, they are invisible for the administrators and they cannot be exceed.

Ensuring bandwidths for VLANs was implemented by setting bandwidth limits for all the VLAN interfaces on the node. Specific implementation is equipment dependent but almost each carrier-class switch or router is capable to perform this task and such implementation is fairly easy.

5.2. Limiting Bandwidth of End-to-end Connection

Recall that all the functions concerning QoS are performed on edge routers, which are gateways for end users. It is true

Table 3
Results of tunnel performance tests

Paths	Packet size [bytes]	Transmission rate [Mbit/s]	Tunnel bandwidth [Mbit/s]	Throughput [Mbit/s]	Latency
R1→MR1→MR5→ MR12→ MR7→ MR2→R2	1460	950	790	825	Min: 1 s Avg: 1 s Max: 1.06 s
R1→MR1→MR5→ MR12→MR7→ MR2→R2	1460	790	790	790	Min: 494 μs Avg: 1.2 ms Max: 9.4 ms
R1→MR1→MR5→ MR12→MR7→ MR2→R2 R2→MR2→ MR7→ MR12→ MR5→MR1→R1	1460	950	410	412	Min: 1 s Avg: 1 s Max: 1.03 s
R1→MR1→MR5→ MR12→MR7→ MR2→R2 R2→MR2→ MR7→ MR12→MR5→MR1→R1	1460	410	410	410	Min: 353 μs Avg: 4.5 ms Max: 19.4 ms
R1→MR1→MR5→ MR12→MR7→ MR2→R2	100	400	35	35	Min: 1.01 s Avg: 1.01 s Max: 1.01 s
R1→MR1→MR5→ MR12→MR7→ MR2→R2	100	35	35	35	Min: 137 μs Avg: 514 μs Max: 7.6 ms
R1→MR1→MR5→ MR12→MR7→ MR2→R2 R2→MR2→ MR7→ MR12→ MR5→MR1→R1	100	200	35	35	Min: 1 s Avg: 1 s Max: 1.02 s
R1→MR1→MR5→ MR12→MR7→ MR2→R2 R2→MR2→ MR7→ MR12→ MR5→MR1→R1	100	35	35	35	Min: 149 μs Avg: 624 μs Max: 36.2 ms

in our case but the architecture of our system does not disenable the use of separate, hardware traffic shapers, which is the case in many professional applications. Guarantee of the bandwidth for end-to-end connections in described implementation is achieved by limiting the bandwidth for the IPv6-in-IPv6 tunnel interfaces. It is a mechanism similar to the above mentioned mechanism for limiting the bandwidth for VLANs: in both cases we limit the bandwidth of a virtual interface. Nonetheless, the actual implementation may vary significantly because handling VLANs is usually performed by switches and IPv6 tunneling by routers since most switches (even with L3 support) are not capable of IPv6-in-IPv6 tunneling.

We focused on implementation end-to-end connection bandwidth limit in Debian 6 OS. A *tc* tool for performing traffic control was chosen because this is a very efficient and common tool, almost each traffic control application in linux bases on *tc*. Indeed, *tc* is very powerful and has functions that meet the requirements.

We used Token Bucket Filter (TBF) classless queuing disciplines. It suits our requirements best because we do not need hierarchical structure offered by classful disciplines (we have only one class in one interface, without involving priorities). TBF is less CPU algorithm, which is important advantage for us because the tests showed that in some cases the performance of virtualizer CPU is bottleneck, because the host CPU is engaged in not only virtual CPUs virtualization but in virtual network adapters too. These advantages make TBF the recommended qdisc for limiting the bandwidth of the entire interface.

Here is an example configuration for the tunnel bandwidth limiting in Debian 6 for the tunnel interface named *tunnel*. The *rate* parameter is our bandwidth limit.

```
tc qdisc add dev tunnel root tbf
rate 2 Mbit latency 1000 ms burst 15000
```

Please note how *tc* calculates units:

$$1 \text{ Mbit} = 1000 \text{ Kbit} = 1000000 \text{ bps}$$

It is commonly confused that specified rate of 1 Mbit equals $1024 \cdot 1024$ bytes.

The *latency* parameter limits the buffer of the algorithm. It means that packets that would wait longer than 1000 ms are simply dropped. Instead of *latency*, it is possible to use *limit* parameter, which means the size of the buffer in bytes. These two parameters are mutually exclusive.

The *burst* parameter is the size of the bucket, in bytes. We increased this value from default 5000 to 15000 in order to let algorithm handle high traffic rates (above 100 Mbit/s). On the other hand, if buffer is too large, the algorithm's precision recede considerable. We chose the value 15000 by experiments.

The only problem concerning traffic control that we were not able to solve is the lack of preciseness when the rates are very high (especially higher than 500 Mbit/s). It is the result of the fact that traffic control is performed by CPU that operates with non-zero time slots and it follows with finite resolution of traffic control mechanisms. The lack of precision is up to 5% of declared rate while the value is 800–1000 Mbit/s. In real applications it does not seem to have serious implications because such large bandwidth rates are rarely reserved for a single end-to-end connection. Yet even in this situation, the QoS is still ensured if 5% of mechanism inaccuracy is calculated in bandwidth allocation plan. If one needs perfect precision with high transmission rates, it is advisable to use a high-class hardware traffic shaper.

6. Performance Comparison and Conclusions

6.1. Comparison of the Performance of the System with the Performance of Alternative Solutions

The authors did not conduct experiments but used the results delineated in [8] in the case of OpenFlow and in [9] in the case of MPLS. Regarding to benchmarking OpenFlow, two factors are to be considered. The former is the performance of a controller and the latter is the bandwidth of switches.

Table 4 presents the performance of different controllers running in one-thread mode on a highly efficient machine with 16-cores processor AMD, 40 GB RAM memory and Debian Wheezy operating system, which hosted simultaneously 16 switches with 100 unique MAC addresses each [8]. The controllers were limited to one-thread operation because the Mirage controller does not support operations on many cores simultaneously.

Table 4
Performance of OpenFlow controllers

Controller	Average throughput [Mbit/s]	Average latency [ms]
NOX fast	122.6	27.4
NOX	13.6	26.9
Maestro	13.9	9.8
Mirage UNIX	68.1	21.1
Mirage XEN	86.5	20.5

As shown in Table 4, the performance of different controllers may vary widely, up to 10 times. It means that the system in the worst case may be not able to handle a new flow that is directed to the controller, hence it will not ensure QoS. Execution Management does not have this issue because it informs an application whether the connection may be established. The application may start to transmit only when a dedicated tunnel with guaranteed bandwidth is active. Due to this proceeding, an end user waits a little bit longer to start for example VoIP conversation, but after the connection is established he has ensured sufficient bandwidth for a VoIP transmission with required quality.

The second, important dimension of the performance is the throughput of devices. In the case of OpenFlow use, it is required to covey to the controller instructions for dealing with a particular connection. It requires the middleware that would communicate with applications and the controller. In such case, the throughput is not constrained by switches. Due to the hardware handling of traffic on low implementation level, the performance of OpenFlow switches does not diverge from the performance of a generic hardware switch. Another issue is limiting resources. The newest OpenFlow version 1.1.0 enables the use of shapers that are built in switches, which means that the mechanism is capable of limiting bandwidth according to requirements even if there are large number of separate traffic flows.

Execution Management uses classles queuing disciplines of TBF type to limit bandwidth of tunnels. It works well for low bandwidth rates. When large bandwidth is configured (more than 500 Mbit/s) it is possible that mechanism would accept the transmission with higher rate than configured. The order of magnitude of the difference is several percent and it depends on configured bandwidth limit. The general rule is that the higher limit is set the bigger is inaccuracy, although slight anomalies may occur while experimenting with narrow range of bandwidth values. We increased the *burst* parameter of TBF discipline (using *tc qdisc* tool) to 10 times MTU size, which cause that the real maximal bandwidth is always equal or higher than configured value. It means that application always has its requested bandwidth and sometimes it may have slightly larger bandwidth than requested. This difference (5% of configured value) is included in QoS arrangement of all the available bandwidth of the network.

In the case of applying OpenFlow, the accuracy of bandwidth limiting depends on the accuracy of traffic control mechanisms implemented in particular device that was chosen by the controller, providing that on the path there is a device that implements traffic control mechanisms.

The second mechanism chosen for comparison is MPLS. In this case there is also lacking a module that allows application to communicate with a network management system in order to establish requested connections automatically. Furthermore, MPLS does not have measures to distinguish between different applications running on a single host. On the other hand, the performance tests shows advantage of MPLS over other solutions. Due to the hardware implementation of packets switching in network

nodes, the performance of MPLS based network is the same as the performance of routed network (with hardware routers).

For traffic management LDP and RSVP protocols might be used. In case of applying both protocols for path and bandwidth managing it is possible to achieve our goals (limiting bandwidth of end-to-end connections, determining the connection's path) but it may cause significant packet loss. According to research [9], the use of above mentioned protocols causes the packet loss of 0.03% to 0.17%. For comparison – our solution does not influence the packet loss rate. Only in the extreme situation, while switching large traffic (more than 100 Mbit/s) from one tunnel to another, the loss of up to 16 packets may occur. Such packet loss does not occur with each iteration (with most iterations no packet is lost). Even in this extreme situation, the packets loss in a second of tunnel path changing is no more than 0.02%.

6.2. Conclusion

For the performance tests of virtual networks with resources guarantees for dedicated connections or for entire isolated virtual networks, we used the Execution Management system for end-to-end connections establishment. We focused on existing solutions and techniques implemented in common network equipment. Owing to this fact, our solution can be deployed in almost any network, for example as point of reference in benchmarking. Except for testing cases, Execution Management is also suitable for business purposes. It allows selling multiple particular, limited network resources (access to virtual networks) of a single physical infrastructure.

The performance of our system is slightly lower comparing to low level hardware-based mechanisms like MPLS and OpenFlow. The advantages of our system are the simple architecture and capability of traffic engineering (in terms of path and bandwidth) in a heterogeneous environment made of generic equipment. The time needed for new connections establishing is not excessively high and rises only slightly in the case of handling large number of requests simultaneously. Moreover, the larger network is served, the higher is probability that multiple request of new connection will be handled faster, providing constant number of parallel requests.

It is worth to mention that Execution Management has several significant functions that are unavailable in competitive solutions, such as discrimination multiple applications on a single host, creation many independent end-to-end connections between a pair of hosts and full support for IPv6. Furthermore, the system cooperates with comfortable, graphical web client, which may be used by the administrator or shared with virtual networks' administrators.

The presented system does not provide excellent performance in the case of large number of connections with high QoS requirements. Nevertheless, the system despite its disadvantages meets the assumed requirements and offers functions that other systems are lacking. On this ground, the system is suitable for reference testing and for design, implementation and management of laboratory testbeds for prototype QoS-aware applications. Execution Management was for example utilized for initial evaluation of applcations designed in the Future Internet Engineering project such as: eDiab [10], SmartFit [11], Online Lab [12] and others, e.g., [13], [14]. In the future work the authors will compare the performance our solution to the performance of the IIP System [15]–[18].

Acknowledgements

The research presented in this paper has been partially supported by the European Union within the European Social Fund and the European Regional Development Fund program no. POIG.01.01.02-00-045/09.

References

[1] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: rapid prototyping for software-defined networks", in *Proc. 9th Ninth ACM Worksh. Hot Topics in Netw. HotNets-IX*, Monterey, CA, USA, 2010.

[2] C. Guo *et al.*, "SecondNet: a data center network virtualization architecture with bandwidth guarantees", in *Proc. 6th ACM Int. Conf. Co-NEXT 2010*, Philadelphia, PA, USA, 2010.

[3] K. Chudzik, J. Kwiatkowski, and K. Nowak, "Virtual networks with the IPv6 addressing in the Xen virtualization environment", in *Computer Networks*, A. Kwiecień, P. Gaj, snf P. Stera, Eds. Berlin, Heidelberg: Springer, 2012, pp. 161–170.

[4] B. Dabiński, D. Petrecki, and P. Świątek, "Performance of mechanisms of resources limiting in networks with multi-layered wirtualization", in *Proc. 17th Polish Teletraf. Symp. 2012*, Zakopane, Poland, 2012, pp. 91–98.

[5] D. Gawor, P. Klukowski, D. Petrecki, and P. Świątek, "Prototyp systemu zdalnego monitoringu parametrów przyżyciowych człowieka w sieci IPV6 z gwarancją QoS", in *Proc. ICT Young 2012*, Gdańsk, Poland, 2012, pp. 409–414 (in Polish).

[6] D. Petrecki, B. Dabiński, and P. Świątek, "Prototype of self-managing content aware network system focused on QoS assurance", in Information systems architecture and technology [electronic doc.]: networks design and analysis, A. Grzech *et al.*, Eds. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2012, pp. 101–110.

[7] ITU-T Rec. E.271 (10/1976).

[8] C. Rotsos *et al.*, "Cost, Performance & Flexibility in OpenFlow: Pick Three", 23.10.2012 [Online]. Available: http://www.cs.nott.ac.uk/rmm/papers/pdf/iccsdn12-mirageof.pdf

[9] G. Liu and X. Lin, "MPLS performance evaluation in backbone network", in *Proc. IEEE Int. Conf. Commun. ICC 2002*, New York, USA, 2002, vol. 2, pp. 1179–1183.

[10] J. Świątek, K. Brzostowski, and J. M. Tomczak, "Computer aided physician interview for remote control system of diabetes therapy", in *Adv. Analysis and Decision-Making for Complex and Uncertain Systems*, Baden-Baden, Germany, 2011, vol. 1, pp. 8–13.

[11] K. Brzostowski, J. Drapała, A. Grzech, and P. Świątek, "Adaptive decision support system for automatic physical effort plan generation – data-driven approach", *Cybernet. and Syst.*, vol. 44, no. 2–3, pp. 204–221, 2013.

[12] P. Świątek, K. Juszczyszyn, K. Brzostowski, J. Drapała, and A. Grzech, "Supporting Content, Context and User Awareness in Future Internet Applications", in *The Future Internet*, Lecture Notes in Computer Science 7281, pp. 154–165. Springer, 2012.

[13] P. Świątek, P. Stelmach, A. Prusiewicz, and K. Juszczyszyn, "Service composition in knowledge-based SOA systems", *New Generation Comput.*, vol. 30, no. 2, pp. 165–188, 2012.

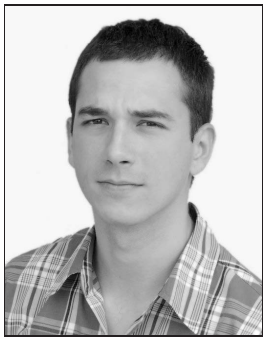
[14] A. Grzech, P. Świątek, and P. Rygielski, "Dynamic resources allocation for delivery of personalized services", in *Software Services for e-World*, W. Cellary and E. Estevez, Eds. IFIP Advances in Information and Communication Technology 341, pp. 17–28. Springer, 2010.

[15] H. Tarasiuk *et al.*, "Architecture and mechanisms of Parallel Internet IPv6 QoS", *Przegląd Telekom. + Wiadomości Telekom.*, vol. 84, no. 8/9, pp. 944–954 2011 (in Polish).

[16] A. Bęben *et. al.*, "Architektura sieci świadomych treści w systemie IIP" ("Architecture of content aware networks in the IIP system"), Przegląd Telekom. + Wiadomości Telekom., vol. 84, pp. 955–963, 2011 (in Polish).

[17] W. Burakowski *et al.*, "Provision of End-to-End QoS in Heterogeneous Multi-Domain Networks", *Ann. of Telecommunications*", Springer, vol. 63, pp. 559–577, 2008.

[18] H. Tarasiuk *et al.*, "Performance evaluation of signaling in the IP QoS system", *J. Telecommun. Inform. Technol.*, no. 3, pp. 12–20, 2011.



Paweł Świątek received his M.Sc. and Ph.D. degrees in Computer Science from Wrocław University of Technology, Poland, in 2005 and 2009, respectively. From 2009 he is with Institute of Computer Science, Wrocław University of Technology, where from 2010 he works as an Assistant Professor. His main scientific interests are focused on services optimization and personalization, optimization of service-based systems, resources allocation, QoS delivery in heterogeneous networks, mobility management in wireless networks and application of service science for e-health.

E-mail: pawel.swiatek@pwr.edu.pl
Institute of Computer Science
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Damian Petrecki received his M.Sc. degree in Computer Science from Wrocław University of Technology, Poland, in 2013. From 2010 to 2013 he was working at the Institute of Computer Science inter alia within Future Internet Engineering project. His scientific interests were focused on multi-layered network virtualization as a way to achieve QoS guaranties in multi-agent or centralized network topology as well as on modern car stabilization using decision-making system and environment awareness.

E-mail: damian.petrecki@gmail.com
Institute of Computer Science
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Bartłomiej Dabiński received the M.Sc. in Information and Communication Technologies from the Wrocław University of Technology in 2013. He has worked as a networking engineer for ISP companies since 2006. During 2011–2012, he worked at the Institute of Computer Science, Wrocław University of Technology. His professional and research interests includes performance analyzing of WISP class equipment, multi-layered virtualization of network resources and QoS mechanisms.

E-mail: bartlomiej@dabinski.pl
Institute of Computer Science
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Paper

Quality Management in 4G Wireless Networking Technology Allows to Attend High-Quality Users

Małgorzata Langer

Institute of Electronics, Lodz University of Technology, Lodz, Poland

Abstract—The 4G networks are all-IP based heterogeneous networks that allow users to use any system at anytime and anywhere, and support a variety of personalized, multimedia applications such as multimedia conferencing, video phones, video/movie-on-demand, education-on-demand, streaming media, multimedia messaging, etc. Personalized services should be supported by personal mobility capability, which concentrates on the movement of users instead of users' terminals. As the technology matures, traffic congestion increases, and competitive pressures mount, QoS and policy management will become more and more important. Also the users come with a number of new requirements on connectivity, and D2D networking technology must follow the idea of "Internet of things". Operators must make sure they are working with vendors that have a strong framework to supply end-to-end QoS and are capable of supporting evolving needs. The paper discusses qualitative and quantitative indicators of telecommunication services. Main challenges that need to be addressed by nowadays and future systems are shown, and within them the massive growth in traffic volume and in the number of connected devices seems to play the most key role.

Keywords—4G, device-to-device communication, LTE, policy management, QoS.

1. Introduction

With the introduction of the first generation (1G) mobile communication systems in the year 1980, "ordinary" people were able to communicate with others with voice while being on the move. Then with the second generation (2G) mobile communication systems, as CDMA (Code Division Multiple Access), GSM (Global Systems for Mobile Communications) in the early 90's, people were able to communicate not only with voice but also with text messaging. With the introduction of the third generation (3G) mobile communication systems, i.e., WCDMA (Wide-band Code Division Multiple Access), also known as UMTS, in the late 90's, people were able to communicate with more data-centric services and applications. 3G networks are based on wireless technology, which wins over its predecessors because of high speed transmission, advanced multimedia access and global roaming. This technology allows connecting the phone to the Internet, or other IP networks, to make a voice or video call or to transmit data. Although multimedia communication is possible with 3G systems, scalability and cost have been problems preventing wide deployment thus far. The 4G networks, i.e., LTE and WiMAX [1] support a variety of personalized, multimedia applications such as multimedia conferencing, video phones, video/movie-on-demand, education-on-demand, streaming media, multimedia messaging, etc. Personalized services should be supported by personal mobility capability, which concentrates on the movement of users instead of users' terminals (one will start watching a film on the 3" screen of his/her mobile, will continue from the exact point on 15" laptop screen and then will move to home TV set screen to see the final scenes), and which involves the provision of personal communications and personalized operating environments. D2D (Device-To-Device) communication forces to use LTE radio access not only for the access (network to terminal) link but also as a solution for wireless backhauling. The heterogeneous deployments of low power network nodes under the coverage of on overlaid layer of macro nodes will meet the idea of communicating machines in next several years.

In the 80's, five functional areas were identified, namely Fault management, Configuration management, Accounting management, Performance management, and Security management (FCAPS). These five functional areas were sufficient to cover most, if not all, of the issues related to the operations and management of the wired networks including the Internet and enterprise networks. With the introduction of wireless and mobile networks several additional areas, which could not be easily covered by FCAPS, had to be added. They are Mobility management, Customer management, and Terminal management.

The quality issues have stopped to be described as "best effort" only (though still it is "best effort" for low data rate in Internet). Supporting multimedia applications with different Quality of Service (QoS) requirements in the presence of diversified wireless access technologies (e.g., 3G cellular, IEEE 802.11 WLAN, Bluetooth) is one of the most challenging issues for fourth-generation (4G) wireless networks. In such network, depending on the bandwidth, mobility, and application requirements, users should be able to switch among the different access technologies in a seamless manner. Efficient radio resource management and CAC (Call Admission Control) strategies are key components in such a heterogeneous wireless system supporting multiple types of applications with different QoS requirements. 4G is a packet-based network. Since it should carry voice as well as Internet traffic it should be able to provide differ-

ent level of QoS. Other network level issues include Mobility Management, Congestion control, and QoS Guarantees. 4G systems are expected to provide real-time services – so, e.g., pre-computed delay bound is required for the service.

2. Migration of Technology – Differences between 3G and 4G

Up to the 3G technology (UMTS) the transmission has been (and is) realized in time domain. The main difference in 4G technology is frequency domain for transmitting. Orthogonal Frequency Division Multiplexing (OFDM) is a combination of TDMA (Time Division Multiplexing Access) and FDMA (Frequency Division Multiplexing Access) and has been known since the 50–60's from military. Analog processing made it extremely expensive. In 80's its complexity reduced due to digital signal processing. It was already a technology proposal for UMTS, but only recently some challenges for mobile communication, particularly for uplink synchronization have been solved. The OFDM becomes today's dominating communication technology and WiMAX, LTE, Flash OFDMA are using it now. The main benefits are:

– high spectral efficiency with simple receiver design,

– bandwidth divided into many narrow tones – in theory fully orthogonal one to each other,

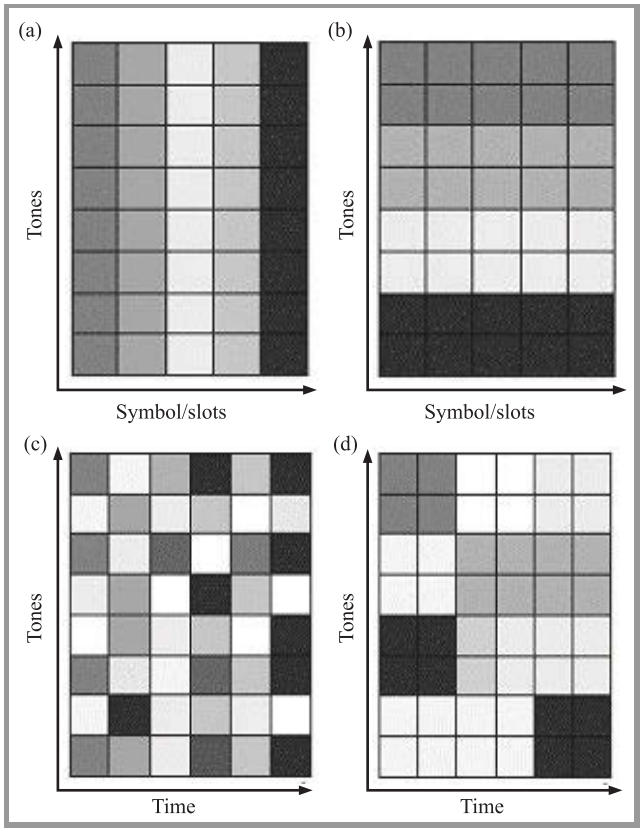


Fig. 1. Multiple access – resource allocation: (a) pure TDMA, (b) pure FDMA, (c) distributed OFDMA, (d) localized OFDMA.

– high-rate data distributed onto many low-rate channels,

– flexible bandwidth allocation.

Figure 1 shows the example of resource allocation in TDMA, FDMA and two types of OFDM: the distributed OFDMA (as in WiMAX), and the localized OFDMA (as in LTE). It helps in better understanding the issue of the technology basics. The deep technical description of them is not the task of this paper, and these considerations would be beyond its scope.

3. Key Services and QoS

The present and future mobile-communication systems differ and will differ with time and with country. They need to be adaptive to the changing service environment. The upper limit of the data rate demand and the lower limit of the delay requirement are difficult to provide in a cost-efficient manner. The services should be provided with the highest data rate, the lowest delay and the lowest jitter that the system can provide. This is unattainable in practice and contradictory to the operator goal of an efficient system – the more delay a service can handle the more efficient the system can be. There are several key services that span the technology space. Those are:

- **Voice**: The end-to-end delay requirement for circuit – switched voice is approximately 400 ms and it is not disturbing humans in voice communication. The sufficient today quality, as the end-to-end delay is not noticeable in older implemented technologies, slows the development of voice service in 4G.

  Voice packets in 4G should require small amounts of data, frequently, with no delay jitter. The problem is that IMS (IP Multimedia Subsystem) has not been developed as fast as it was expected and voice applications stay in the circuit switched domain, still.

- **Real-time applications** (games, mainly): Experts say that players look for game servers with a ping time of less than 50 ms [2]. So these applications require small amounts of data, as game update information, but with a low delay, a limited delay jitter and relatively frequent.

- **Interactive file applications** (download and upload): They require high data rates and low delays.

- **Background file download and upload**: The example is e-mail. This service accepts lower bit rates and longer delays.

- **Television**: This is the streaming downlink to many users at the same time requiring moderate data rates (higher with HD). The very low delay jitter, though delays may be tolerated (but approximately the same delay for all users in the neighborhood.

Table 1
Selected indicators and assessed parameters of telecommunications services

Service	Indicator	Parameters
Audio	Availability	Rate of server accessibility; listening break-up ratio; listening break-up frequency; successful one minute listening ratio
	Fidelity/ accuracy	Audio quality
	Speed	Access time; starting delay
	Capability	Throughput achieved
	Reliability	Rate of overall technical reliability; levels of customers complaints
	Flexibility	Provider capacity to adjust to the user connection and equipment features
	Usability	User friendless of the interface
	Security	Protection against user identity theft; protection against intrusion and breach of customer's privacy
Directory enquiry services	Availability	Rate of accessability to the service; outage frequency; served call rate
	Fidelity/ accuracy	Rate of correctness in answering the customer questions
	Speed	Response time for directory enquiry services; replay time
	Capability	Adequacy of the number of operators to the number of call
	Flexibility	Range of availability means to access the service
	Usability	User friendless of the interface; ability of the operator to cope with the caller language
	Security	Compliance to the customer security specifications as given in the contract, in particular: protection of the customer's private data or related to the person concerned by the enquiry
E-mail	Availability	Rate of SMTP failures; rate of POP3 failures; outages rate; outages frequency; rate of message loss
	Fidelity/ accuracy	Rate of undue deletions of email by the security mechanisms
	Speed	Average time to check an empty mailbox; average delivery time
	Capability	Server throughput; speed to upload to mail server; speed of download from mail server
	Reliability	Rate of overall technical reliability; level of customers complaints
	Flexibility	Ease to change the contractual specifications
	Usability	User friendless if the interface
	Security	Protection against intrusion, spam and any king of viruses
Fax	Availability	Refers to availability of connection
	Fidelity/ accuracy	Transmission fidelity test
	Reliability	Ratio of sent and received fax; rate of overall technical reliability; level of customer complaints
	Security	Protection against identity, theft and content violation
Internet access	Availability for Internet access	Successful login ratio; outage rate, outage frequency; rate of successful authentication; rate of successful access to generic name translation
	Availability for web of browsing	Outage rate to a set of designed sites; availability of web pages hosted by ISP; frequency of untimely breakup; rate accessibility to the input ports, rate of accessibility to the output ports
	Availability for web page hosting	Rate of accessibility to the allocated space
	Fidelity/ accuracy for web browsing	Error rate in data transmissions
	Speed for Internet access	Delay; radio channel access delay; authentication time; generic domain name translation time
	Speed for web browsing	Web response time
	Speed for web page hosting	Time to upload a test web page
	Capability for Internet access	Throughput achieved; throughput of dialup access to the Internet
	Capability for web Browsing	Occupation rate of ISP links, occupation rate of ISP input ports
	Reliability for the overall service	Rate of overall technical reliability; level of customer complaints; fault report rate per fixed access lines
	Flexibility for the overall service	Ease to change the contractual specifications
	Usability for easier the overall service	User friendless of the interface; adaptability to make use to people with disabilities
	Security for the overall service	Protection against user identity theft, intrusion and breach of customer's privacy
Multimedia Message Service (MMS)	Availability	Successful MMS Ratio
	Fidelity/ accuracy	Completion Rate for MMS
	Speed	End to end delivery time
	Reliability	Rate of overall technical reliability; level of customer complaints
	Flexibility	Ease to change the contractual specification; range of available means to send and receive MMS
	Usability	User Friendless of the interface
	Security	Protection against intrusion, spam and any kind of viruses
Operator services	Availability	Rate of accessibility to the service; outage frequency; served call rate
	Fidelity/ accuracy	Rate of correctness in fulfilling the customer request
	Speed	Response time for operator services; call setup time
	Capability	Adequacy of the number of operators to the number of call
	Reliability	Rate of overall technical reliability; level of customer complaints
	Flexibility	Range of available means to access the service
	Usability	User friendless of the interface; ability of the operator to cope with the caller language
	Security	Protection of the customer's private data
Short Message Service (SMS)	Availability	Successful SMS Ratio
	Fidelity/ accuracy	Completion rate for SMS
	Speed	End to end delivery time for SMS
	Reliability	Rate of overall technical reliability; level of customer complaints
	Flexibility	Range of available means to send and receive SMS
	Usability	User friendless of the interface
	Security	Protection against intrusion, spam and any kind of viruses
Telephony	Availability	Unsuccessful call ratio; dropped call ratio; retain ability rate; outage rate
	Fidelity/ accuracy	Audio quality, video quality
	Speed	Access time, starting time
	Capability	Throughput achieved
	Reliability	Rate of overall technical reliability; level of customer complaints
	Flexibility	Ease to change the contractual specifications
	Usability	User friendliness of the interface, adaptability to make use easier to disable people
	Security	Protection against user identity theft, fraudulent
Video broadcast	Availability	Rate of server accessibility; display breakup ratio; display breakup frequency; successful one minute watching ratio
	Fidelity/ accuracy	Audio quality; video quality
	Speed	Access time; starting time
	Capability	Throughput achieved
	Reliability	Rate of overall technical reliability; level od customer complaints
	Flexibility	Ease to change the contractual specifications
	Usability	User friendliness of the interface
	Security	Protection against user identity theft
Voice mail	Availability	Rate of success ful access to the recording server; rate of successful access to the message listening server; outage frequency of the message recording server; rate of message loss
	Fidelity/ accuracy	Rate of message spoiling failure of the information to the voice mailbox owner
	Speed	Response time of the voice guide after the reply time out; message recording server response time; time to receive the notification of a message record in the voice mailbox; listening message server response time
	Reliability	Rate of overall technical reliability; level of customer complaints
	Flexibility	Ease to change the contractual specifications; range of available means to record and receive
	Usability	User friendliness of the interface
	Security	Protection against fraudulent message listening and change of the welcome recorded message

Policy management allows operators to control granularly the availability and QoE of different services. First, policies are used to dynamically allocate network resources – for example, a particular bandwidth can be reserved in the radio base station and core network to support a live video conversation. Next, policy rules control the priority, packet delay, and the acceptable loss of video packets in order for the network to treat the video call in a particular manner. The quality of service is characterized by the combined aspects of service support performance, service operability performance, service ability performance, service security performance and other factors specific to each service [3].

TL 9000 [4] is the first unified set of quality system requirements and metrics designed specifically for the telecommunications industry. TL 9000 encompasses the ISO 9001 standard, plus additional industry-specific telecom requirements and covers industry performance based measurements including reliability, delivery, and service quality. The TL 9000 management system is applied by telecom manufacturers and suppliers engaged in the design, development, production, delivery, installation and maintenance of telecommunications products and services.

Table 1 presents QoS indicators, gathered on the base of the ETSI Guide – Quality of Telecom Services [5]. The introduced set of parameters gives the review of technical, economical and functional issues those have to be considered and indicators those should be measured and tested objectively. Some technical measurements may not be directly perceptible by customers or can vary from a particular user feeling but still the indicator value or the feature affects the quality and assurance of service. The indicators presented in ETSI Guide can be divided into two parts. The first one is related with technical, functional aspect and the second one is related to other aspects such as: sales, repair, provision, charging, billing, upgrade, complaint management, commercial and technical support. Nowadays, there are several standards describing QoS measurements, for example [4], [6], [7]. Measurements of the parameters can be made using different methods: technical measurements performed by an independent organization or by the supplier, a survey performed among users, or a mixture of user's opinion and technical measurements. Some results are obtained in terms of degrees of satisfaction and not in technical terms. For example the condition of "Seamless mobility" is kept until a user doesn't notice that the hand-over happens.

Table 2 covers standardized QCIs (QoS Class Identifiers) for LTE, where GBR stands for Ground Based Radio here [8].

Guaranteed Bit Rates (GBRs) are not part of them since as traffic handling attributes cannot be preconfigured for a QoS class. They must therefore be dynamically signaled within the service, instead. A QCI is simply a "pointer" to a TFP (Traffic Forwarding Policy) and can be associated with a TFP defined within each user plane edge/node. Within a specific node multiple QCIs may be associated with the same TFP. Within up to the 4G technologies either "best effort" policy prevails or, in the 3G (i.e. UMTS) four traffic classes with defined QoS attributes may be pointed: conversational, streaming, interactive, background.

Table 2
Standardized QCI for LTE

QCI	Resource type	Priority	Packet delay budget [ms]	Packet error loss rate	Example services
1	GBR	2	100	10^{-2}	Conversational voice
2	GBR	4	150	10^{-3}	Conversational video (live streaming)
3	GBR	5	300	10^{-6}	Non-conversational video (buffered streaming)
4	GBR	3	50	10^{-3}	Real gaming
5	Non-GBR	1	100	10^{-6}	IMS signaling
6	Non-GBR	7	100	10^{-3}	Voice, video (live streaming), interactive gaming
7	Non-GBR	6	300	10^{-6}	Video (buffered streaming)
8	Non-GBR	8	300	10^{-6}	TCP-based (i.e. WWW, e-mail), chat, FTP, P2P file sharing, progressive video, etc.

Performance-enhancing features can improve perceived quality of service (end-user's point of view) or system performance (operator's point of view). Though LTE and its evolution can yield better data rates and shorter delay, so it can greatly improve as the service experience (for an end-user) as the system capacity (for an operator).

4. Network Architecture

In up to 3G technologies, i.e., for example in majority of today implemented cellular networks, based on circuit-switching, they consist of base stations, base station controllers, switching centers, gateways, and so on. The base station (BS) plays the role of physical transmission with fast power control and wireless scheduling. The base station controller (BSC) performs the largest part of the radio resource management. Whenever a mobile terminal (MT)

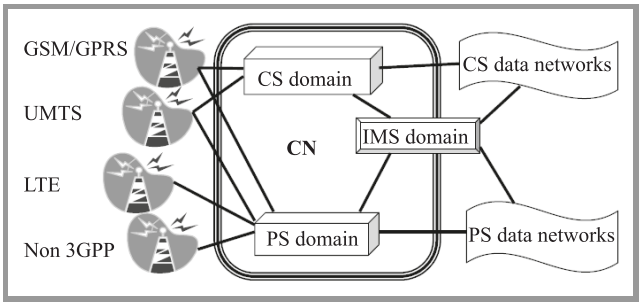


Fig. 2. Idea of coexisting technologies.

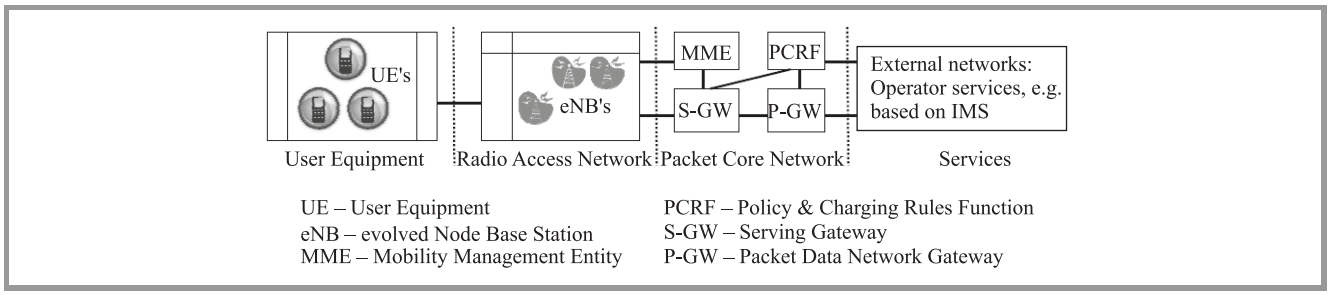


Fig. 3. LTE architecture – separated domains.

moves into another cell, it requires handoff to another base station. In 4G network the base station must function intelligently to perform radio resource management as well as physical transmission, more as a smart access router. Figure 2 introduces the idea of coexisting technologies. LTE is only connected to the packet core, while circuit core will not continued to be developed. 3GPP packet core can connect to non 3GPP technologies. IMS (IP Multimedia System) is (it should be) the integral part of the evolved packet core.

Figure 3 shows the overall LTE architecture, where one can see the clear separation and defined interfaces between different domains. In such a layout any evolution is independent of access, core, transport and services. The main principle of SAE (System Architecture Evolution) is that control and user's panels are separated one from the other (Fig. 4). Nevertheless it is only a "logical" architecture by now, though it assures the flexible deployment for various scenarios.

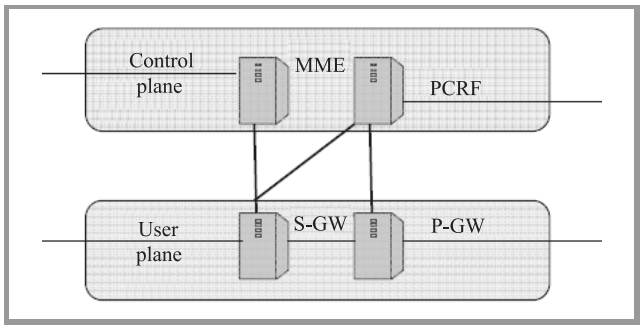


Fig. 4. Separation of control and user planes.

The separation of control and user planes gives the highly optimized implementation and scalability (number of users versus data traffic/services). The control plane covers MME, which is a powerful server and is placed at a secure location. It contains NAS (Non Access Stratum) protocol, covers such functions as security control, managing subscriber profile and service connectivity, packet core bearer control. The other server PCRF is a part of the operator's switching centre and deals with policy and charging control, with decisions on how to handle QoS in the network.

The bearer uniquely identifies packet flows that receive a common QoS treatment between the terminal and the gateway. Independent of a service type, a bearer is defined through the network to which it connects the UE (User Equipment), referred to as Access Point Name in 3GPP and the QoS Class Identifier (QCI). The bearer is the basic enabler for traffic separation, that is, it provides differential treatment for traffic with differing QoS requirements. So one UE can have several bearers at the same time, if a call is on, files are transferred, etc. The concept of the bearer and the associated signaling procedures further enable the system to reserve system resources before packet flows those are mapped to that bearer into the system. Data to be transmitted enters the processing chain in the form of IP packets on one of the SAE bearers.

Prior to transmission over the radio interface, incoming packets are passed through multiple protocol entities.

With regard to quality and quality management the most important protocol entities are:

- PDCP (Packet Data Convergence Protocol) – there is one PDCP entity per SAE bearer and it is responsible for ciphering and integrity protection of the transmitted data;
- RLC (Radio Link Control) – located in the eNodeB, which offers services to the PDCP in the form of RBs (Radio Bearers) – there is one RLC entity per radio bearer configured for a terminal; it is responsible for segmentation/concatenation, retransmission handling and in-sequence delivery to higher layers;
- MAC (Medium Access Control) – which handles scheduling (as uplink as downlink); the scheduling functionality is located in eNodeB, which has one MAC per cell;
- PHY (Physical Layer) – it offers services to the MAC layer in the form of transport channels; the PHY handles coding/decoding, modulation/demodulation, multi-antenna mapping and other typical physical layer functions; it offers services to the MAC layer in transport channels.

4.1. Logical Channels and Transport Channels

The logical channels' set (the MAC offers services to the RLC in the form of logical channels) covers control channels, used for transmitting control and configuration information necessary for LTE, and a traffic channel, used for the user data (Dedicated Traffic Channel – DTCH).

The MAC uses services in the form of transport channels, and a transport channel is defined by how and with what parameters the information is transmitted over the radio interface. Data in a transport channel is organized into transport blocks. Each transport block is related to a Transport Format (TF). The TF includes information about size, modulation scheme etc., but also the MAC layer can control different data rates by varying the transport format.
DL-SCH (Downlink Shared Channel) is the main transport channel used to transmit downlink data in LTE. It supports the dynamic rate adaptation, channel-dependent scheduling in time and frequency domains, hybrid ARQ (Automatic Repeat Request) with soft combining, controls mobile-terminal power consumption through DRX (discontinuous reception). The similar features relate to UL-SCH (Uplink Shared Channel).

4.2. Schedulling

The main principle of the LTE radio access is shared transmission, it means that the time – frequency resources are dynamically shared between users. The dynamic scheduler is a part of the MAC layer. It controls the assignment of uplink and downlink resources.
The scheduler takes advantage of the channel variations and schedules transmissions to a mobile terminal on resources with better channel conditions. But the decision is taken per mobile terminal and not per radio bearer (RB). So the terminal is the only one that handles logical-channel multiplexing and is responsible for the choice from which RB the data is taken. Each RB is assigned a priority and a prioritized data rate. Remaining resources, if any, are given to the radio bearers in priority order.
For the downlink the terminal transmits channel-status reports and for the uplink a sounding reference signal those make the base to take the proper decision. The scheduler also controls the inter-cell interference. The scheduling strategy is vendor specific and may vary for different cases.

5. Policy Management

In term of global connectivity, LTE solutions can deliver a data rate of at least 100 Mbit/s between any two points in the world, with smooth handoff across heterogeneous networks, so they result in seamless connectivity and global roaming across multiple networks. They provide a high quality of service for next-generation multimedia support including real-time audio, high-speed data, Internet protocol television (IPTV), video content and mobile TV. On top of that, carriers must provide for interoperability with existing wireless standards, and an all IP, packet switched network that can bridge the great difference of latency between networks. Quality of service provision in 4G networks present several challenges including: the specification of QoS requirements, the translation of QoS parameters among heterogeneous access networks, the renegotiation of QoS, and the management of QoS requirement within roaming agreements and mobile users profiles. Traditionally, the handover process has been considered among wireless networks using the same access technology (e.g., among cells of a cellular network). This kind of handover process is the horizontal handover (HHO). The new handover process among networks using various technologies is the vertical handover (VHO). The vertical handover in 4G networks and WiMAX networks has been developed to provide QoS routing, specification and management of QoS contracts, and handover control through the establishment of transparent procedure.
Policy management plays a fundamental role in implementing QoS in mobile broadband. It is the process of applying operator-defined rules for resource allocation and network usage. Dynamic policy management sets rules for allocating network resources, and includes policy enforcement processes. Policy enforcement involves service data flow detection and applies QoS rules to individual service data flows.
Manufacturers, vendors, mobile operators do not have unlimited resources and capital and the radio spectrum is finite. Three areas relate to the policy management:

- limiting network congestion,
- monetizing services,
- enhancing service quality.

Providing high service quality by over-provisioning network capacity would leave an operator at a competitive disadvantage to providers that offer the same or better quality service at a lower cost. Policy management starts with differentiating services and subscriber types, and controlling the QoE (Quality Of Experience) of each type.

6. Key Challenges for the Nearest Future

Company Ericsson estimates that the human-centric communication devices that are currently dominant will be surpassed tenfold by "communication machines" in the future [9] and predicts that overall traffic demands will increase in the order of a thousand times within the next 10 years. In addition to straightforward densification of a macro deployment, network densification can be achieved by the deployment of complementary low-power nodes under the coverage of an existing macro-node layer. In such a heterogeneous deployment, the low-power nodes provide very high traffic capacity and very high user throughput locally, for example in indoor and outdoor hotspot positions. Highly efficient macro base stations will ensure QoE over the entire coverage area and at the same time will have to serve as backhaul for more local access (so called: "dual connectivity"). Energy efficient load balancing, per link optimization, enhanced support for mobility are some examples of benefits in such a solution.
Complementing a cellular system with the option of Wi-Fi access can be used to further boost the overall traffic capacity and service level.

LTE is already capable of handling a wide range of D2D scenarios, though some revolutionary development is requested, i.e., mass, low cost D2D device types; allowing for very low device energy consumption; handling a very large number of devices per cell. Signaling for every connected device can result in a very high control-plane load. For that reason, lightweight signaling procedures are desired to reduce the signaling load per device that is caused to the network. A key feature of LTE D2D communication, including proximity detection, is its integration into the overall wireless access network. Whether communication occurs directly between devices or via the infrastructure should be transparent to the user, and the network should be involved and assist in the D2D communication.

7. Conclusions

The world is at the beginning of an era marked by tremendous growth in mobile data subscribers and mobile data traffic. Infonetics Research predicts that mobile data subscribers will grow from 548.9 million in 2010 to 1.8 billion in 2014 [10]. Today's mobile broadband networks carry multiple services that share access (radio) and core network resources. Each service has different QoS requirements in terms of packet delay tolerance, acceptable packet loss rates, and required minimum bit rates. Additionally one should consider two perspectives of performance: end user's one and operator's one. Given that system resources are limited, there will thus be a trade-off between a number of active users and the perceived quality of service in terms of user throughput. The 4G mobile systems focus on seamlessly integrating the existing wireless technologies including GSM, wireless LAN, and Bluetooth. The 4G networks are all-IP based heterogeneous networks that allow users to use any system at anytime and anywhere. Users carrying an integrated terminal can use a wide range of applications provided by multiple wireless networks. The 4G systems provide not only telecommunications services, but also data and multimedia services. The evolution of LTE is the most important step to ensure a high-quality wireless network for the future.

As the technology matures, traffic congestion increases, and competitive pressures mount, QoS and policy management will become more and more important. In preparation, operators must make sure they are working with vendors that have a strong framework to supply end-to-end QoS and are capable of supporting evolving needs.

A bearer has two or four QoS parameters, depending whether it is real-time or best effort service:

- QoS Class Indicator (QCI),
- Allocation and Retention Priority (ARP),
- Guaranteed Bit Rate (GBR)– real real-time services only,
- Maximum Bit Rate (MBR) – real-time services only.

Data applications are typically best effort services, characterized by variable bit rates, and are tolerant to some loss and latency before the user perceives poor quality.

The standards and recommendations provide mechanisms to drop or downgrade lower-priority bearers in situations where the network become congested.

The eNodeB is the radio base station in LTE and it plays a critical role in end-to-end QoS and policy enforcement. The eNodeB performs uplink and downlink rate policing, as well as RF radio resource scheduling. It uses ARP when allocating bearer resources. The effectiveness of radio resource scheduling algorithms in eNodeB's has a tremendous impact on service quality and overall network performance. Quality of Experience is a measure of the overall level of customer satisfaction with a service. Quantitatively measuring QoE requires an understanding of the Key Performance Indicators (KPI) that impact users' perception of quality. KPIs are unique by service type. Soon the radio base stations at the same time will have to serve as backhaul for dual connectivity. Each service type such as conversational video, voice, and internet browsing, have unique performance indicators that must be independently measured.

The evolved LTE architecture is able to provide QoS per user and per service, implementing the notion of a user profile associated with control element functions. An integrated management approach to service and network management in the case of heterogeneous and mobile network access is a key to quality management. LTE employs intelligent scheduling methods to optimize performance, from both end-user and operator standpoints of view.

Next steps on developing the technology should extent LTE (or LTE-A) to new use cases (machine type communications for D2D mainly), and probably to better possibilities for the close integration of LTE and Wi-Fi deployments.

References

[1] E. M. Zeman, "ITU Changes Tune, LTE and WiMAX Can Be Called '4G'", 18.12.2010 [Online]. Available: http://www.phonescoop.com/news

[2] E. Dahlman *et al.*, *3G Evolution. HSPA and LTE for Mobile Broadband*. Elsevier, 2008.

[3] G. Held, *Network Management. Techniques, Tools and Systems*. New York, NY: Wiley, 1992.

[4] TL 9000 Release 5.0, June 22, 2009.

[5] ETSI EG 202 009 – 1 Quality of telecom services, User Group; V1.2.1, Jan. 2007.

[6] Quality of Service Standards and Guidelines for the Telecommunications Sector – Consultative Doc., Office of Utilities Regulation, Oct. 2010

[7] N. Seitz, "ITU-T QoS Standards for IP-Based Networks", *IEEE Commun. Mag.*, vol. 41, no. 6, June 2003.

[8] S. Sesia, I. Toufik, and M. Baker, LTE – *The UMTS Long Term Evolution: From Theory to Practice*. Wiley, 2011.

[9] Ericsson Research & Development [Online]. Available: http://www.ericsson.com/thinkingahead/networked_society

[10] IXIA – 915-2731-01 Rev A: Quality of Service (QoS) and Policy Management in Mobile Data Networks, Aug. 2010.

[11] R. Ludwig, H. Ekstrom, P. Willars, and N. Lundin, "An evolved 3GPP QoS concept", in *Proc. IEEE 63rd Veh. Technol. Conf. VTC Spring 2006*, Melbourne, Australia, 2006.

[12] ITU-T – X.200 series of recommendations.
[13] ITU-T – Recommendation series M.3000, 2011.

Małgorzata Langer graduated in Analogues and Digital Circuits of Automatics, at the Lodz University of Technology and got her Ph.D. degree in electronics. After several years in international marketing she joined the TUL in 1988, first in the Institute of Materials Science and then the Institute of Electronics (since 2004 as Associate Director on Science). She works as Associate Professor in Telecommunications Team and delivers lectures and laboratories for students of various courses in Polish and English. Her main science and research interests are in non-deterministic object simulation, telecommunications, QoS, some aspects of new nonconventional technologies in microelectronics, reliability, theory of experiments, etc. She is the author and the co-author of over 60 publications, also quoted, and has been involved in several international, home, and university research and didactics programs.

E-mail: Malgorzata.langer@p.lodz.pl
Institute of Electronics
Lodz University of Technology
Wolczanska st 223
90-924 Lodz, Poland

Paper

ILP Modeling of Many-to-Many Replicated Multimedia Communication

Krzysztof Walkowiak[a], Damian Bulira[a], and Davide Careglio[b]

[a] Department of Systems and Computer Networks, Wrocław University of Technology, Wrocław, Poland
[b] Advanced Broadband Communication Center (CCABA), Universitat Politecnica de Catalunya, Barcelona, Spain

Abstract—On-line communication services were evolving from a simple text-based chats towards sophisticated videopresence appliances. The bandwidth consumption of those services is constantly growing due to the technology development and high user and business needs. That fact leads us to implement optimization mechanisms into the multimedia communication scenarios. In this paper, the authors concentrate on many-to-many (m2m) communication, that is mainly driven by the growing popularity of on-line conferences and telepresence applications. An overlay model where m2m flows are optimally established on top of a given set of network routes is formulated and a joint model where the network routes and the m2m flows are jointly optimized. In the models, the traffic traverses through replica servers, that are responsible for stream aggregation and compression. Models for both predefined replica locations and optimized server settlement are presented. Each model is being followed by a comprehensive description and is based on real teleconference systems.

Keywords—ILP modeling, many-to-many communication, network optimization, replica location.

1. Introduction

Since the beginning of the Internet, network flow paradigms have undergone significant transformation. From a one-to-one transmission that can be represented by fetching a website from a server or simple one-to-one Voice over IP call those paradigms evolved into sophisticated schemes with complex traffic matrix. To optimize the traversal of the same information from one host to the group of others, one-to-many (multicast) applications were introduced. A good example of that is IP TV streaming in triple-play services (Internet, phone and TV) [1] or synchronization messages exchange in Network Time Protocol [2]. Furthermore, one-to-one-of-many (anycast) can be distinguished. In anycast, packets are routed to one of many servers – that can be represented by a common address – with the lowest path cost from a source to a destination. Such distributed networks are called Content Delivery Networks (CDN) and they play the main role in current Internet-based business [3]. In this paper, the authors focus on many-to-many (m2m) communication as one of the fastest emerging paradigms and propose ILP models of offline problems related to optimization of m2m flows using replica servers. To achieve this, the m2m transmission with both anycast and unicast paradigm is modeled. The former, similarly to CDN, is used during the replica selection phase and the latter to transfer the data from the selected server, back to the client. In this type of transmissions, all hosts exchange the information with every other host in the m2m group. The information is forwarded first to the replica server (rendezvous point), which in turn propagates proper data to other hosts that take part in the m2m group. The examples of such traffic are: video and teleconferencing, distance learning, multiplayer on-line gaming, distributed computing, etc. The authors focus on videoconferencing as the widespread and demanding example of m2m service. Moreover, a business need for videoconference system is not anymore a nice to have feature for the enterprise, but an essential day-to-day tool that makes the business more effective and successful. According to Cisco, business videoconferencing will grow six fold between 2011 and 2016 [4]. The authors of the report claim, that business videoconferencing traffic is growing significantly faster than overall business IP traffic, at a compound annual growth rate of 48% over the forecast period.

Furthermore, using replication in videoconferencing, bandwidth used for the transmission is significantly reduced. Replicas not only aggregate the traffic, but also perform stream modifications such as format change or compression. Currently, end nodes in videoconferencing are mobile devices, PCs, dedicated videophones or special telepresence equipment. Each of them requires different audio and videostream formats due to available computational resources. Using replicas, complex multimedia transcoding is moved from end nodes to highly efficient dedicated servers. Moreover, encoded stream requires less bandwidth, that decreases network congestion and provide higher level of Quality of Service to end users.

The main contributions of this paper are integer linear programming models for many-to-many transmission in computer networks where rendezvous points are used. The authors propose overlay and joint models assuming combined optimization of overlay and underlying networks. Moreover, two different strategies of locating replica servers in the network are presented. In the first strategy, the location of the servers is known and only the client assignment and network flows have to be optimized. In the latter case, the location of the replica servers is unknown and is a subject of optimization. The models support video conference applications, but can be easily redefined for other type of m2m traffic.

This paper is an extended version of the paper [5], presented at 17th Polish Teletraffic Symposium PTS 2012, held in Zakopane, Poland on December 5–7, 2012. This extended paper contains the new results, including a ILP formulations of replica location problem for many-to-many multimedia communication. To the best of our knowledge, this work is the first one that addresses the problem of replica location in m2m networks.

The remainder of this paper is organized as follows. Section 2 provides related works study on many-to-many communication. Section 3 describes the m2m communication in computer networks. In Section 4, an ILP model for overlay network is presented. Section 5 contains similar model for joint m2m system, using the node-link notation. Two further sections extend the previous models with the replica location problem. Section 6 describes the overlay model of the replica location problem, and Section 7 refers to the joint model. Finally, the paper is concluded in Section 8.

2. Related Works

The idea of many-to-many communication in the networks is not a recent invention. The author in [6] predicted that teleconferences will be as popular as television. After many years, we know how true was this prediction. Extended view on m2m applications in background of multicast is presented in [7]. The authors define m2m traffic as a group of hosts, where each of them receives data from multiple senders while it also sends data to all of them. They also highlight that this communication paradigm may cause complex coordination and management challenges. The examples of m2m applications are, among others: multimedia conferencing, synchronizing resources, distributed parallel processing, shared document editing, distance learning or multiplayer games, to name a few. Moreover, the paper presents a brief comparison of delay tolerance and mentions that m2m applications characterize in a high delay intolerance.

In [8], the authors propose scheduling architecture for m2m traffic in switched HPC (High Performance Computing) networks. The paper also mentions other applications of m2m communications in data centers, for example process and data replication [9], dynamic load-balancing [10] or moving virtual machine resources between servers connected into a cloud [11]. In [12], the authors presents optimal and nearly optimal hot potato routing algorithms for many-to-many transmissions. In hot potato (deflection) routing, a packet cannot be buffered, and is therefore always moving until it reaches its destination. This scenario is mostly applicable in parallel computing applications.

Many-to-many communication is also extensively investigated in the area of radio networks. Overview on this topic is presented in [13]. The authors of [14] propose a Middleware for Many-to-many Communication (M2MC) system architecture for m2m applications in broadcast networks (both radio and wired). Because of broadcast orientation, M2MC do not require any resource consuming routing protocols. The system architecture comprises of Message Ordering Protocol, Member Synchronization Protocol and protocols for processes to join and leave the groups.

Other applications of m2m communication exist in a field of online gaming [15]–[18]. All the players need to exchange with the others the current state of the game. In dynamic games delay tolerance is crucial, and online gaming protocols are designed to transfer small portions of data in often transmitted packets. When more servers are available, the game world is usually splitted into several zones and users are assigned to the server, taking under account a zone in which their avatar currently exists.

Mixed-integer Linear Programming (MILP) formulation for many-to-many traffic grooming in Wavelength-Division Multiplexing (WDM) networks is presented in [19] and [20]. The authors not only formulate MILP problems, but also present approximated heuristic algorithms. Both solutions are considered for non-splitting networks, where optical-electronic-optical conversion is used and in networks capable of splitting the signal in optical domain. In WDM networks, due to wide optical spectrum even broadband many-to-many multimedia streams may be aggregated (groomed) to use available bandwidth more efficiently.

Many-to-many transmission in telepresence appliance is presented in [21]. The authors compare two architectures, namely centralized and distributed. Moreover, the video transmission is encoded using Scalable Video Coding (SVC) [22]. In SVC, a stream consists of a base layer and several enhancement layers, that after merging with the base layer, improve a video quality. Every client receives as many layers as the link, that it is connected to the network, can handle at low delay. Finally, different approaches to the video exchange during videoconferences have been presented in [23]. The authors proposed an algorithm to build separate trees for different enhancement layers in SVC based transmission. They make a theoretical analysis to show optimality of the algorithm and prove it through extensive simulations.

In [24], the authors propose a flow control protocol based on cost-benefit approach. Practical realization of this protocol framework for many-to-many flow control in overlay networks is designed and tested both in extensive simulations and real-life experiments.

Overlay networking is a subject of interest in numerous publications. An extensive work on overlay networks can be found in [25]. The author provides a complete introduction to the topic, followed by architecture description, requirements, underlying topologies, and routing information. The work is also supplemented with a discussion about security and overlay networks applications.

Replica location problem has been addressed in previous publications [26], [27]. However, most of the work has been done in a relation to the Content Delivery Network and web-content servers [28] or transparent proxying [29]. In the topic of multimedia transmission, previous work concentrates mostly on placing Video on Demand (VoD)

servers or static multimedia replicas [30], however, in [31] the authors address anycast in a field of relaying node selection and Voice over Internet Protocol (VoIP) Session Border Controller (SBC) placement.

3. Many-to-Many Communication

As mentioned in the previous section, many-to-many communication is a paradigm of data exchange between group of hosts in a way that every group member gets information from the rest of hosts involved in the transmission. Basically, during the transmission every host in the group has the same set of information (i.e. all videoconference participants see video streams from other conference members). The overall set of m2m demands is known in advance and the problem consists of optimizing the establishment of the m2m flows to serve these demands. This abstract model was divided into two more specific problems for the communication in computer networks:

- **Overlay model**. In this model, the m2m flows are determined assuming a given set of network routes already established, i.e., the service layer is decoupled from the IP layer. This model is easier to deploy since there is no need of the network topology information and the traffic routing in the network layer;
- **Joint model**. In this model, the establishment of the m2m flows involves also the underlying network layers (e.g., IP layer, MPLS layer, optical layer, etc.). This model is harder to implement but allows optimizing network routes and m2m flows together in order to minimize bandwidth usage.

4. Overlay m2m Systems – Optimization Model

In this section, the ILP model of the offline m2m flows allocation in overlay system is presented. First, we introduce the main assumptions of an overlay system with m2m flows. A set of users (overlay nodes) indexed $v = 1, 2, \ldots, V$ that participate in the system is given, i.e., each user generates some stream with rate h_v (defined in bit/s) and receives the aggregated streams from other users. For instance in the context of teleconferencing system, the value h_v depends on the selected coding standard and resolution. A special compression ratio α_v is defined for each user – the user receives the overall stream compressed according to this ratio. This assumptions also follows from real teleconference systems [32], [33]. In the considered system, servers $s = 1, 2, \ldots, S$ are rendezvous points. In a nutshell, each user sends its flow to one selected server. The server aggregates all received flows, and thus provides the stream to each user with the requested compression ratio. Each server $s = 1, 2, \ldots, S$ has a limited upload and download capacity (u_s and d_s, respectively). Another possible model – not addressed here – is a case when servers exchange information with each other and the users receives the aggregated stream of all users from one selected server.

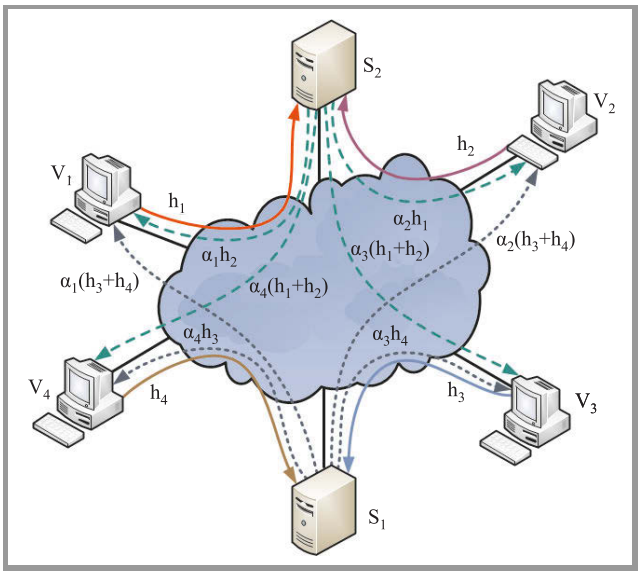


Fig. 1. Many-to-many transmission model in overlay network.

As an example, Fig. 1 shows the considered overlay model in a network with 4 clients (users) and 2 servers. Clients v_1 and v_2 are sending their streams to server s_2 and clients v_3 and v_4 to s_1. Both upstream and downstream flows are presented and transmission volume is shown. For example client v_1 transmits stream with volume h_1 to server s_2 and receives two streams compressed with requested compression ratio α_1. The former comes from s_2 and consists of stream h_2 from client v_2 (its own stream is not sent back), the latter comes from s_1 and consists of streams h_3 and h_4 from corresponding clients v_3 and v_4.

There are two sets of decision variables in the model. First, z_{vs} denotes the selection of server s for demand v. The second variable H_s is auxiliary and defines the flow of all users connected to server s. The objective is to minimize the overall streaming cost according to the allocation of users to servers. For each pair of overlay nodes (both users and/or servers) we are given constant ζ_{vw} denoting the streaming cost of one capacity unit (i.e., Mbit/s) on an overlay link from node v to node w. The cost can be interpreted in many ways, e.g., as network delay (in ms), bandwidth consumption, number of Autonomous Systems (ASes) on the path, etc., or a weighted combination of them. To present the model notation as in [34] is used:

– **indices**

$v, w = 1, 2, \ldots, V$	user (overlay nodes),
$s = 1, 2, \ldots, S$	servers (overlay nodes);

– **constants**

d_s	download capacity (bit/s) of server s,
u_s	upload capacity (bit/s) of server s,
ζ_{vw}	streaming cost on overlay link from node v to node w,

h_v stream rate (bit/s) generated by node (client) v,
α_v compression ratio of node (client) v,
N_s maximum number of users that s can serve;

– **variables**
z_{vs} = 1, if user v is assigned to server s and 0 otherwise (binary),
H_s flow aggregated at server s (continuous);

– **objective**

$$\min\ F = \sum_v \sum_s z_{vs} h_v \zeta_{vs} + \sum_v \sum_s \alpha_v (H_s - z_{vs} h_v) \zeta_{sv}\,, \quad (1)$$

– **subject to**

$$\sum_s z_{vs} = 1 \quad v = 1,2,\ldots,V\,, \quad (2)$$

$$H_s = \sum_v z_{vs} h_v \quad s = 1,2,\ldots,S\,, \quad (3)$$

$$H_s \le d_s \quad s = 1,2,\ldots,S\,, \quad (4)$$

$$\sum_v \alpha_v (H_s - z_{vs} h_v) \le u_s \quad s = 1,2,\ldots,S\,, \quad (5)$$

$$\sum_v z_{vs} \le N_s \quad s = 1,2,\ldots,S\,. \quad (6)$$

The objective (1) is to minimize the streaming cost of transferring all m2m flows in the system. In more detail, function (1) compromises two elements. The first one (i.e., $\sum_v \sum_s z_{vs} h_v \zeta_{vs}$) denotes the cost of streaming the data from users to servers. The second part (i.e., $\sum_v \sum_s \alpha_v (H_s - z_{vs} h_v) \zeta_{sv}$) defines the cost of streaming the data in the opposite direction from each server to each user. Recall that for each user a special compression ratio α_v is given. Moreover, if a particular server s is selected by user v (i.e., $z_{vs} = 1$), the flow of this server is decreased by the flow of user v. Constraint (2) assures that for each user v exactly one server is selected. In (3), the aggregated flow entering each server s is defined as the sum of all users' flows assigned to s. In constraints (4) and (5) the download and upload capacity constraints for servers is defined. Each server uploads the aggregated stream with the defined compression ratio to each user. Therefore, similarly to obj. (1), the original flow of user v is not sent back to this node. Since the upload and download flows of users are constant, we do not formulate capacity constraint in the case of user nodes. Finally, constraint (6) bounds the number of users to be served by each server. This limit follows from real m2m systems (e.g., teleoconferencing systems) [33]. The presented model in (1)–(6) is strongly NP-hard problem since it is equivalent to the Multidimensional Knapsack Problem [35].

A special case of the overlay model presented in (1)–(6) is a scenario where only one server ($S = 1$) is applied to provide the m2m transmissions in the network. Notice that in this case, this model becomes an analytical model, since there are no variables as all users are assigned to the same server (variable z_{vs}). As a consequence, the aggregated flow at the server is constant and given by

$$H_1 = \sum_v h_v\,. \quad (7)$$

The cost of one server scenario is as follows

$$F = \sum_v h_v \zeta_{v1} + \sum_v \alpha_v (H_1 - h_v) \zeta_{1v}\,. \quad (8)$$

Notice that Eq. (8) can be used as a reference cost when evaluating multi servers scenarios.

5. Joint m2m Systems – Optimization Model

Now, a joint model of m2m flows is introduced. The main assumptions are analogous to the overlay model. The key difference is that with the joint model, network routes between users and servers can be optimized. The authors will formulate joint system ILP model using node-link notation [34].

The considered network is modeled as a directed graph consisting of nodes and links. Nodes are divided into two subsets: nodes hosting servers (indexed by s = 1, 2, ..., S) and all other nodes (indexed by v = 1, 2, ..., V). Users can be connected only to nodes v = 1, 2, ..., V. We assume that server nodes are connected to the graph by a bridge (cut-edge), i.e., removal of the edge disconnects the server node from the rest of the graph. This follows from the fact that server nodes cannot be used as a transit node for forwarding data that does not originate or terminate at the server node. In contrast, nodes v = 1, 2, ..., V can be used as transit nodes. Links are denoted using index e = 1, 2, ..., E.

Recall that in the case of overlay systems, the notion of a node was used to denote a user. To simplify the notation, in this section we apply the notion of a demand d = 1, 2, ..., D to denote all flows in the system between users and servers. Let $o(d)$ and $t(d)$ denote the origin and destination node of each demand, respectively. There are two types of demands: upstream and downstream. The former one denotes the flow from a user to one of the servers, thus for each upstream demand d, $o(d)$ denotes the user node. The upstream demand is an anycast demand, since one of the end nodes is to be selected among many possible nodes. The volume of this demand is constant and given by h_d. Since an upstream demand is defined by the user node $o(d)$ we can write that that $h_d = h_{o(d)}$, i.e., volume of upstream demand d is equivalent to the bitrate generated by client located at node $o(d)$.

For each user (node v) there are S downstream demands to transmit the aggregated flow from each server to the user node. The destination node $t(d)$ of each downstream demand is always located in a user node. Consequently, candidate paths for each demand connect the server node and the user node. Downstream demands are unicast since both end nodes are defined a priori. Moreover, with every down-stream demand we introduce the index of associated

up-stream demand $\tau(d)$. Both associated demands d and $\tau(d)$ of the same request must connect the same pair of nodes: the client node and the selected replica node. However, the main novelty is that the volume of downstream demands is a variable and depends on the allocation of users to servers. In more detail, the volume of downstream demand d is defined as $\alpha_{t(d)}(H_{o(d)} - z_{t(d)o(d)}h_{\tau(d)})$.

Let a_{ev} and b_{ev} denote the binary constants that define the dependency between adjacent links and nodes. More precisely, a_{ev} is 1, when link e originates at node v and 0 otherwise. Similarly, b_{ev} is 1, if link e terminates at node v and 0 otherwise.

– **indices**

$v = 1,2,\ldots,V$	network client nodes,
$s = 1,2,\ldots,S$	network server nodes,
$d = 1,2,\ldots,D$	demands (upstream from user to server and downstream from server to user),
$e = 1,2,\ldots,E$	network links;

– **constants**

h_d	volume (requested bit-rate) of upstream demand d,
ζ_e	streaming cost on link e,
c_e	capacity of link e,
$ds(d)$	= 1, if d is a downstream demand, 0 otherwise,
$us(d)$	= 1, if d is an upstream demand, 0 otherwise,
$o(d)$	origin (source) node of demand d, for an upstream demand $o(d)$ denotes the user node, for a downstream demand $o(d)$ denotes the server node,
$t(d)$	destination node of demand d, in the case of a upstream demand $t(d)$ denotes the server, while in the case of downstream demand $t(d)$ is the user node,
$\tau(d)$	index of a demand associated with demand d. If d is a downstream demand, then $\tau(d)$ must be an upstream connection and vice versa,
M	large number,
N_s	maximum number of users that s can serve,
a_{ev}	= 1, if link e originates at node v, 0 otherwise,
b_{ev}	= 1, if link e terminates at node v, 0 otherwise;

– **variables**

z_{vs}	= 1, if user v is assigned to server s, 0 otherwise (binary),
H_s	flow aggregated at server s (continuous),
x_{ed}	flow of demand d on link e (continuous),
u_{ed}	= 1, if demand d uses link e, 0 otherwise (binary);

– **objective**

$$\min F = \sum_d \sum_e x_{ed}\zeta_e \tag{9}$$

– **subject to**

$$\sum_e a_{es}x_{ed} - \sum_e b_{es}x_{ed} = \alpha_{t(d)}(H_{o(d)} - z_{t(d)o(d)}h_{\tau(d)})$$
$$d = 1,2,\ldots,D \quad ds(d) = 1 \quad s = 1,2,\ldots,S \quad o(d) = s \tag{10}$$

$$\sum_e a_{ev}x_{ed} - \sum_e b_{ev}x_{ed} = -\alpha_{t(d)}(H_{o(d)} - z_{t(d)o(d)}h_{\tau(d)})$$
$$\text{if } v = t(d) \quad d = 1,2,\ldots,D \quad ds(d) = 1 \quad v = 1,2,\ldots,V \tag{11}$$

$$\sum_e a_{ev}x_{ed} - \sum_e b_{ev}x_{ed} = 0$$
$$\text{if } v \neq t(d) \quad d = 1,2,\ldots,D \quad ds(d) = 1 \quad v = 1,2,\ldots,V \tag{12}$$

$$\sum_e a_{ev}x_{ed} - \sum_e b_{ev}x_{ed} = h_d$$
$$\text{if } v = o(d) \quad d = 1,2,\ldots,D \quad us(d) = 1 \quad v = 1,2,\ldots,V \tag{13}$$

$$\sum_e a_{es}x_{ed} - \sum_e b_{es}x_{ed} = -h_d z_{o(d)s}$$
$$d = 1,2,\ldots,D \quad us(d) = 1 \quad s = 1,2,\ldots,S \quad t(d) = s \tag{14}$$

$$\sum_e a_{ev}x_{ed} - \sum_e b_{ev}x_{ed} = 0$$
$$\text{if } v \neq o(d) \quad d = 1,2,\ldots,D \quad us(d) = 1 \quad v = 1,2,\ldots,V \tag{15}$$

$$\sum_e a_{es}u_{ed} - \sum_e b_{es}u_{ed} = 1$$
$$d = 1,2,\ldots,D \quad ds(d) = 1 \quad s = 1,2,\ldots,S \quad o(d) = s \tag{16}$$

$$\sum_e a_{ev}u_{ed} - \sum_e b_{ev}u_{ed} = -1$$
$$\text{if } v = t(d) \quad d = 1,2,\ldots,D \quad ds(d) = 1 \quad v = 1,2,\ldots,V \tag{17}$$

$$\sum_e a_{ev}u_{ed} - \sum_e b_{ev}u_{ed} = 0$$
$$\text{if } v \neq t(d) \quad d = 1,2,\ldots,D \quad ds(d) = 1 \quad v = 1,2,\ldots,V \tag{18}$$

$$\sum_e a_{ev}u_{ed} - \sum_e b_{ev}u_{ed} = 1$$
$$\text{if } v = o(d) \quad d = 1,2,\ldots,D \quad us(d) = 1 \quad v = 1,2,\ldots,V \tag{19}$$

$$\sum_e a_{es}u_{ed} - \sum_e b_{es}u_{ed} = -z_{o(d)s}$$
$$d = 1,2,\ldots,D \quad us(d) = 1 \quad s = 1,2,\ldots,S \quad t(d) = s \tag{20}$$

$$\sum_e a_{ev}u_{ed} - \sum_e b_{ev}u_{ed} = 0$$
$$\text{if } v \neq o(d) \quad d = 1,2,\ldots,D \quad us(d) = 1 \quad v = 1,2,\ldots,V \tag{21}$$

$$x_{ed} \leq M u_{ed} \quad d = 1,2,\ldots,D \quad e = 1,2,\ldots,E \tag{22}$$

$$H_s = \sum_{d:up(d)=1} z_{o(d)s} h_d \quad s = 1,2,\ldots,S \tag{23}$$

$$\sum_s z_{o(d)s} = 1 \quad d = 1,2,\ldots,D \quad up(d) = 1 \tag{24}$$

$$\sum_d x_{ed} \leq c_e \quad e = 1,2,\ldots,E \tag{25}$$

$$\sum_{d:up(d)=1} z_{o(d)s} \leq N_s \quad s = 1,2,\ldots,S \tag{26}$$

The objective function (9) minimizes the cost of all network flows. Constraints (10)–(12) define the flow conservation laws for downstream demands. Recall that in our model the downstream demand is a unicast demand from a server to a user. Therefore, as a source node only server nodes are considered, see constraint (10). The right-hand side of (10) denotes the flow of downstream demand d, which is the flow received by the user from each server. The compression ratio is applied and the original stream generated by the node is not sent back. Constraint (11) relates to the destination node of the demand, i.e., user node. Finally, constraint (12) is formulated for other so called transit nodes. Furthermore, in (13)–(15) the flow conservation of upstream demands is defined, which are anycast. In more detail, (13) denotes the flow conservation for the user node. Constraint (14) meets the guarantee that one of the servers (defined by the value of z_{vs} variable) is selected as the destination node. Constraint (15) defines the flow conservation law for remaining transit nodes. Notice that we assume that server nodes can be used as transit nodes to forward traffic of demands not terminated or originated at particular server node. Since we assume single path routing, constraints (16)–(18) and (19)–(21) denote the flow conservation constraints for corresponding binary flow variables u_{ed}. Both flow variables are bound through using constraint (22).

Constraint (23) – similarly to (3) – defines the flow of server s according to assignment of users to servers. Constraint (24) defines variable z_{vs}. Constraint (25) is the link capacity. Finally, (26) limits the number of clients served by each server. Model (9)–(26) is NP-complete since it is equivalent to the single path allocation problem [34].

Notice that in order to obtain bifurcated version of the link-node model variables u_{ed} and constraints (16)–(22) must be removed from the above model.

6. Overlay System Replica Location Problem – Optimization Model

In this section, the ILP model of replica location problem in overlay m2m systems is introduced, that belongs to the group of LFA (Location and Flow Allocation) problems. In the previous two models, the authors assumed that the location of the replica servers is fixed. Here, the problem is to choose R replicas among V potential sites ($R < V$) taking under consideration demands in the network. In comparison to the equivalent problem (1)–(6), where location of the replicas is known, we do not distinguish client and server nodes. We are given $v, w = 1, 2, \ldots, V$ nodes from which R replica nodes will be selected. Therefore, binary variable z_w is used, which is 1 when w hosts a replica server and 0 otherwise. The problem of locating replicas in the network is NP-hard, since it is equivalent to the facility location problem [28], [36]:

– **indices**

$v, w = 1,2,\ldots,V$ overlay nodes;

– **constants**

d_v download capacity (bit/s) of node v,
u_v upload capacity (bit/s) of node v,
ζ_{vw} streaming cost on overlay link from node v to node w,
h_v streaming rate (bit/s) generated by node v,
α_v compression ratio of node v,
N_v maximum number of users that v can serve,
R number of replica servers,

– **variables**

z_{vw} = 1, if node v is assigned to replica node w, 0 otherwise (binary),
z_w = 1, if node w is selected to host a replica server, 0 otherwise (binary),
H_w flow aggregated at replica node w (continuous);

– **objective**

$$\min F = \sum_v \sum_w z_{vw} h_v \zeta_{vw} + \sum_v \sum_w \alpha_v (H_w - z_{vw} h_v) \zeta_{vw} \tag{27}$$

– **subject to**

$$\sum_w z_{vw} = 1 \quad v = 1,2,\ldots,V \tag{28}$$

$$H_w = \sum_{v:v \neq w} z_{vw} h_v = 1 \quad w = 1,2,\ldots,V \tag{29}$$

$$H_w < d_w \quad w = 1,2,\ldots,V \tag{30}$$

$$\sum_v \alpha_v (H_w - z_{vw} h_v) \leq u_w \quad w = 1,2,\ldots,V \tag{31}$$

$$\sum_v z_{vw} \leq N_w \quad w = 1,2,\ldots,V \tag{32}$$

$$\sum_w z_w \leq R \tag{33}$$

$$z_{vw} \leq z_w \quad v, w = 1,2,\ldots,V \tag{34}$$

The objective (27) is to minimize the streaming cost of transferring all m2m flows in the system. First component denotes the cost of streaming the data from users to servers. The second part defines the cost of streaming the data in the opposite direction. Constraint (28) assures that each user is assigned to exactly one replica node. The flow ag-

gregated at each replica is defined in (29). Constraints (30) and (31) are defining download and upload capacity boundaries. The number of users to be served by each server is constrained in (32). Constraint (33) guarantees that R nodes are selected to host replica servers. Finally, (34) binds variables z_{vw} and z_w, i.e., node w can be selected as the replica node for any user v, only if node w is assigned with a replica node ($z_w = 1$).

7. Joint System Replica Location Problem – Optimization Model

Analogously to the problem presented in the previous section, the base problem with replica servers selection is extended. Due to the simplicity of the model representation node-link notation is used.

– **indices**

$v, w = 1,2,\ldots,V$	network nodes,
$d = 1,2,\ldots,D$	demands (upstream from user to server and downstream from server to user),
$e = 1,2,\ldots,E$	network links;

– **constants**

h_d	volume (requested bit-rate) of upstream demand d,
ζ_e	streaming cost on link e,
c_e	capacity of link e,
$ds(d)$	= 1, if d is a downstream demand, 0 otherwise,
$us(d)$	= 1, if d is an upstream demand, 0 otherwise,
a_{ev}	= 1, if link e originates at node v, 0 otherwise,
b_{ev}	= 1, if link e terminates at node v, 0 otherwise,
α_v	compression ratio of node v,
N_v	maximum number of users that v can serve,
$o(d)$	origin (source) node of demand d, for an upstream demand $o(d)$ denotes the user node, for a downstream demand $o(d)$ denotes the server node,
$t(d)$	destination node of demand d, in the case of a upstream demand $t(d)$ denotes the server, while in the case of downstream demand $t(d)$ is the user node,
$\tau(d)$	index of a demand associated with demand d; if d is a downstream demand, then $\tau(d)$ must be an upstream connection and vice versa,
R	number of replica servers,
M	large number;

– **variables**

z_{vw}	= 1, if node v is assigned to replica node w, 0 otherwise (binary),
z_w	= 1, if node w is selected to host a replica server, 0 otherwise (binary),
H_w	flow aggregated at replica node w (continuous),
x_{ed}	flow of demand d on link e (continuous),
u_{ed}	= 1, if demand d uses link e, 0 otherwise (binary);

– **objective**

$$\min F = \sum_d \sum_e x_{ed} \zeta_e \tag{35}$$

– **subject to**

$$\sum_e a_{ev} x_{ed} - \sum_e b_{ev} x_{ed} = \alpha_{t(d)} (H_{o(d)} - z_{t(d)o(d)} h_{\tau(d)})$$
$$d = 1,2,\ldots,D \quad ds(d) = 1$$
$$v = 1,2,\ldots,V \quad v = o(d) \tag{36}$$

$$\sum_e a_{ev} x_{ed} - \sum_e b_{ev} x_{ed} = -\alpha_{t(d)} (H_{o(d)} - z_{t(d)o(d)} h_{\tau(d)})$$
$$d = 1,2,\ldots,D \quad ds(d) = 1$$
$$v = 1,2,\ldots,V \quad v = t(d) \tag{37}$$

$$\sum_e a_{ev} x_{ed} - \sum_e b_{ev} x_{ed} = 0$$
$$d = 1,2,\ldots,D \quad ds(d) = 1$$
$$v = 1,2,\ldots,V \quad v \neq t(d) \quad v \neq o(d) \tag{38}$$

$$\sum_e a_{ev} x_{ed} - \sum_e b_{ev} x_{ed} = h_d (1 - z_v)$$
$$d = 1,2,\ldots,D \quad us(d) = 1$$
$$v = 1,2,\ldots,V \quad v = o(d) \tag{39}$$

$$\sum_e a_{ev} x_{ed} - \sum_e b_{ev} x_{ed} = -h_d z_{o(d)v}$$
$$d = 1,2,\ldots,D \quad us(d) = 1$$
$$v = 1,2,\ldots,V \quad v \neq o(d) \tag{40}$$

$$\sum_e a_{ev} u_{ed} - \sum_e b_{ev} u_{ed} = z_{o(d)}$$
$$d = 1,2,\ldots,D \quad ds(d) = 1$$
$$v = 1,2,\ldots,V \quad v = o(d) \tag{41}$$

$$\sum_e a_{ev} u_{ed} - \sum_e b_{ev} u_{ed} = -z_{o(d)}$$
$$d = 1,2,\ldots,D \quad ds(d) = 1$$
$$v = 1,2,\ldots,V \quad v = t(d) \tag{42}$$

$$\sum_e a_{ev} u_{ed} - \sum_e b_{ev} u_{ed} = 0$$
$$d = 1,2,\ldots,D \quad ds(d) = 1$$
$$v = 1,2,\ldots,V \quad v \neq t(d) \quad v \neq o(d) \tag{43}$$

$$\sum_e a_{ev} u_{ed} - \sum_e b_{ev} u_{ed} = 1 - z_v$$
$$d = 1,2,\ldots,D \quad us(d) = 1$$
$$v = 1,2,\ldots,V \quad v = o(d) \tag{44}$$

$$\sum_e a_{ev} u_{ed} - \sum_e b_{ev} u_{ed} = -z_{o(d)v}$$
$$d = 1,2,\ldots,D \quad us(d) = 1$$
$$v = 1,2,\ldots,V \quad v \neq o(d) \tag{45}$$

$$x_{ed} \leq M u_{ed} \quad d = 1,2,\ldots,D \quad e = 1,2,\ldots,E \tag{46}$$

$$H_v = \sum_{d:up(d)=1} z_{o(d)v} h_d \quad v = 1,2,\ldots,V \tag{47}$$

$$\sum_v z_{o(d)v} = 1 \quad d = 1,2,\ldots,D \quad up(d) = 1 \tag{48}$$

$$\sum_d x_{ed} \leq c_e \quad e = 1,2,\ldots,E \tag{49}$$

$$\sum_{d:up(d)=1} z_{o(d)v} \leq N_v \quad v = 1,2,\ldots,V \tag{50}$$

$$\sum_v z_v \leq R \tag{51}$$

$$z_{vw} \leq z_w \quad v,w = 1,2,\ldots,V \tag{52}$$

The objective function (35) minimizes the cost of all network flows. Constraints (36)–(38) define the flow conservation laws for downstream demands. In detail, (36) presents the case, when v is a source of demand d, so it is a potential replica. If so, right hand side of (36) denotes the flow of demand d, otherwise equals 0. Constraint (37) is defined for the destination node of demand d $(v = t(d))$, hence the left-hand denotes the flow that enters to the client node v. We assume that the replica node can be located only in the nodes that are not the client nodes. Finally in (38) v represents an intermediate node and flow balance equals 0. In analogous way we formulate the flow conservation law for upstream demands (39)–(40). Constraint (40) represents two cases - when v is a replica node or an intermediate node. In the former, variable $z_{o(d)v}$ is set to 1 and right-hand site of (40) denotes flow of demand d incoming to replica v. In the latter, $z_{o(d)v}$ is set to 0 and right-hand side equals 0. In this model a single path routing is considered, thus constraints (41)–(43) and (44)–(45) denote the flow conservation constraints for corresponding binary flow variables u_{ed}. This variable is bound with continuous flow variable x_{ed} in constraint (46). Constraints (47)–(50) are analogous to the node-link problem model with known server location. Constraints (51)–(52) are equivalent of (33)–(34) in the overlay model.

8. Concluding Remarks

In this paper, ILP optimization models of computer networks with many-to-many multimedia flows was formulated. The authors addressed two problems of replica server settlement – with known replica location, and with optimized replica location selection. According to many recent developments in computer networks, m2m transmissions have been gaining much popularity in different areas. The models presented can be easily adaptaded for other traffic patterns and applications. Generic ILP models of m2m flows optimization in overlay model and joint mode as-l suming combined optimization of overlay and underlying networks (e.g., IP layer, MPLS layer, optical layer, etc.) was proposed. The models assume that special servers (rendezvous point) collect flows of individual clients and sent them back to users using some compression. In future work, the authors plan to implement the models in ILP solvers as well as to develop some heuristic algorithms to obtain numerical results, and to formulate models of m2m systems using multicasting for effective transmission.

Acknowledgements

The work was supported by the Polish National Science Centre under the grant N N519 650440. The work on this paper was held when Damian Bulira was on the PhD internship in Advanced Broadband Communication Center (CCABA), Universitat Politecnica de Catalunya, Barcelona, Spain.

References

[1] K. C. Almerotha and M. H. Ammar, "The use of multicast delivery to provide a scalable and interactive Video-on-Demand service", *IEEE J. Selec. Areas Telecommun.*, vol. 14, no. 6, 6 1996.

[2] D. Mills *et al.*, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, June 2010.

[3] J. Choi, J. Han, E. Cho, T. Kwon, and Y. Choi, "A survey on content-oriented networking for efficient content delivery", *IEEE Commun. Mag.*, vol. 49, iss. 3, pp. 121–127, 2011

[4] Cisco Visual Networking Index: Forecast and Methodology 2011–2016, 2012.

[5] K. Walkowiak, D. Bulira, and D. Careglio, "ILP modeling of many-to-many transmissions in computer networks", in *Proc. 17th Polish Teletraf. Symp. 2012*, Zakopane, Poland, 2012, pp. 123–128.

[6] C. H. Stevens, "Many-to-Many Communication", Tech. rep. MIT/Sloan/TR-175 Sloan School of Management, Massachusetts Institute of Technology, 1981.

[7] B. Quinn and K. Almeroth, "IP Multicast Applications: Challenges and Solutions", RFC 3170, September 2001.

[8] S. Banerjee *et al.*, "Contention-Free Many-to-Many Communication Scheduling for High Performance Clusters", Distributed Computing and Internet Technology, LNCS, vol. 6536, pp. 150–161. Heidelberg: Springer, 2011.

[9] N. Touheed *et al.*, "A comparison of dynamic load-balancing algorithms for a parallel adaptive flow solver", *Parallel Comput.*, vol. 26, no. 12, pp. 1535–1554, 2000.

[10] K. Sinha *et al.*, "Efficient load balancing on a cluster for large scale online video surveillance", in *Proc. 10th Int. Conf. Distrib. Comput. Netw. ICDCN 2009*, Hyderabad, India, 2009, V. Garg *et al.*, Eds. LNCS, vol. 5408, pp. 450–455, Heidelberg: Springer, 2009.

[11] T. C. Wilcox Jr., "Dynamic load balancing of virtual machines hosted on Xen", Master thesis, Dept. of Computer Science, Brigham Young University, April 2009.

[12] A. Borodin, Y. Rabani, and B. Schieber, "Deterministic many-to-many hot potato routing", *IEEE Trans. Paral. Distrib. Sys.*, vol. 8, no. 6), pp. 587–596, 1997.

[13] B. S. Chlebus, D. R. Kowalski, T. Radzik, "On many-to-many communication in packet radio networks", in *Proc. 10th Int. Conf. Princip. Distrib. Sys. OPODIS 2006*, Bordeaux, France, 2006, A. A. Shvartsman, Ed. LNCS, vol. 4305, pp. 260–274, Heidelberg: Springer, 2006.

[14] C. K. Bhavanasi, "M2MC: Middleware for many to many communication over broadcast networks", in *Proc. 1st In. Conf. Commun. Sys. Softw. Middlew. COMSWARE 2006*, New Delhi, India, 2006.

[15] F. Glinka, A. Ploss, J. Müller-Iden, and S. Gorlatch, "RTF: A real-time framework for developing scalable multiplayer online games", in *Proc. 6th ACM Worksh. Netw. Sys. Support Games NETGAMES 2007*, Melbourne, Australia, 2007, pp. 81–86.

[16] A. Ploss, S. Wichmann, F. Glinka, and S. Gorlatch, "From a Single- to Multi-Server Online Game: A Quake 3 Case Study Using RTF", in *Proc. ACM Int. Conf. Adv. Comp. Entertain. Technol. ACE 2008*, Yokohama, Japan, 2008.

[17] P. Prata, E. Pinho, and E. Aires, "Database and state replication in multiplayer online games", in Proc. *24th IEEE Int. Conf. Adv. Infor. Netw. Appl. Worksh. WAINA 2010*, Perth, Australia, 2010.

[18] M. Assiotis and V. Tzanov, "A distributed architecture for MMORPG", in P*roc. 5th Worksh. Netw. Sys. Sup. Games NETGAMES 2006*, Singapore, 2006.

[19] M. A. Saleh and A. E. Kamal, "Many-to-many traffic grooming in WDM networks", *IEEE/OSA J. Opt. Commun. Netw.*, vol. 1, iss. 5, pp. 376–391, 2009.

[20] M. A. Saleh and A. E. Kamal, "Approximation algorithms for many-to-many traffic grooming in optical WDM networks", *IEEE/ACM Trans. Netw.*, vol. 20 , iss. 5, pp. 1527–1540, 2012.

[21] T. A. Le and H. Nguyen, "Centralized and distributed architectures of scalable video conferencing services", in *Proc. 2nd Int. Conf. Ubiquitous Future Netw. ICUFN 2010*, Jeju Island, Korea, 2010, pp. 394–399.

[22] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC Standard", *IEEE Trans. Circut Sys. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, 2007.

[23] M. Ponec *et al.*, "Optimizing multi-rate peer-to-peer video conferencing applications", *IEEE Trans. Multim.*, vol. 13, no. 5, pp. 856–868, 2011.

[24] Y. Amir *et al.*, "Flow control for many-to-many multicast: a cost-benefit approach", in *Proc. IEEE Conf. Open Architec. Netw. Program.OPENARCH 2001*, Anchorage, Alaska, USA, 2001.

[25] S. Tarkoma, *Overlay Networks: Toward Information Networking.* Auerbach Publications, 2010.

[26] M. Rabinovich, "Issues in web content replication", *Data Engin. Bull.*, vol. 21, no. 4. pp. 21–29, 1998.

[27] T. Loukopoulos, I. Ahmad, and D. Papadias,"An overview of data replication on the Internet", in *Proc. 6th Int. Sym. Parall. Architec., Algorithms Netw. I-SPAN'02*, Metro Manila, Philippines, 2002.

[28] L. Qiu, "On the placement of web server replicas", in *Proc. 20th Ann. Joint Conf. IEEE Comp. Commun. Societ. INFOCOM 2001*, Anchorage, AK, USA, 2001, vol. 3, pp. 1587–1596.

[29] Y. Shavitt, "Proxy location problems and their generalizations", in *Proc. 23rd Int. Conf. Distrib. Comput. Sys. Worksh.*, Providence, RI, USA, 2003.

[30] Y. Tu, J. Yan, and S. Prabhakar, "Quality-aware replication of multimedia data", in *Proc. 16th Int. Conf. Datab. Expert Sys. Appl. DEXA '05*, Copenhagen, Denmark, 2005, K. V. Andersen, J. Debenham, and R. Wagner, Eds. LNCS 3588, pp. 240–249. Berlin Heidelberg: Springer, 2005.

[31] R. Cohen and G. Nakibly, "A traffic engineering approach for placement and selection of network services", *IEEE/ACM Trans. Netw.(TON)*, vol. 17, no. 2, pp. 487–500, 2009.

[32] D. Maldow, "Videoconferencing Infrastructure: A Primer", 2012 [Online]. Available: http://www.telepresenceoptions.com/2012/07/videoconferencing_infrastructu/

[33] Polycom RealPresence Platform: Scalable Infrastructure for Distributed Video, Polycom Whitepaper, 2011.

[34] M. Pioro and D. Medhi, Routing, Flow and Capacity Design in Communication and Computer Networks. Morgan Kaufmann, 2004.

[35] J. Puchinger, G. Raidl, and U. Pferschy, "The multidimensional Knapsack problem: structure and algorithms", *INFORMS J. Comput.*, vol. 22, no. 2, pp. 250–265, Spring 2010.

[36] "A compendium of NP optimization problems", P. Crescenzi, and V. Kann, Eds., 2005 [Online]. Available: http://www.nada.kth.se/∼viggo/wwwcompendium/

Krzysztof Walkowiak received his Ph.D. and D.Sc. (habilitation) degrees in Computer Science from the Wrocław University of Technology, Poland, in 2000 and 2008, respectively. Currently, he is an Associate Professor at the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Technology. His research interest is mainly focused on optimization of content-oriented networks; network survivability; elastic optical networks; optimization of distributed computing systems (cloud computing, grids); application of soft-optimization techniques for design of computer networks. He was involved in many research projects related to optimization of computer networks. Moreover, he was a consultant for projects for large companies including TPSA, PZU, PKO BP, Energia Pro, Ernst & Young. He has published more than 160 scientific papers. He serves as a reviewer for many international journals including: IEEE Communications Magazine, IEEE/ACM Transactions on Networking, IEEE Journal on Selected Areas in Communications, Computer Communication, Computational Optimization and Applications. He is/was actively involved in many international conferences. Prof. Walkowiak is a member of IEEE and ComSoc.
E-mail: krzysztof.walkowiak@pwr.wroc.pl
Faculty of Electronics
Department of Systems and Computer Networks
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Damian Bulira received his M.Sc. degree in Computer Science from Wrocław University of Technology, Poland, in 2011. Since that time he is a Ph.D. student at the Department of Systems and Computer Networks, Faculty of Electronics, Wrocław University of Technology. Currently, he is doing a Ph.D. internship at the Broadband Communication Systems research group, Department of Computer Architecture, Universitat Politecnica de Catalunya, Barcelona, Spain. His research interests includes network optimization, many-to-many networking, multimedia systems optimization, to name a few. He is a IEEE Student member and IEEE Communication Society member.
E-mail: damian.bulira@pwr.wroc.pl
Faculty of Electronics
Department of Systems and Computer Networks
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Davide Careglio received the M.Sc. and Ph.D. degrees in Telecommunications Engineering both from Universitat Politecnica de Catalunya (UPC), Barcelona, Spain, in 2000 and 2005, respectively, and the Laurea degree in Electrical Engineering from Politecnico di Torino, Turin, Italy, in 2001. He is currently an Associate Professor in the Department of Computer Architecture at UPC. His research interests include algorithm and protocol design in communication networks.

E-mail: careglio@ac.upc.edu
Advanced Broadband Communication
Center (CCABA)
Universitat Politecnica de Catalunya
Jordi Girona 1-3
D6-103
08034 Barcelona, Spain

Paper

Minimizing Cost of Network Upgrade for Overlay Multicast – Heuristic Approach

Maciej Szostak and Krzysztof Walkowiak

Department of Systems and Computer Networks, Wrocław University of Technology, Wrocław, Poland

Abstract—A rapid increase of the Internet users and traffic at the rate of 31% in years 2011–2016 contributes to emerging of new approaches to the content distribution. Among other approaches, the overlay multicasting seems to be one of the most interesting concepts according to relatively low deployment costs and large scalability. In this paper, the authors formulate a new incremental multicast overlay design problem. In particular, authors assumed that the overlay network is to be upgraded due to an increase of the number of participating users and the need to improve the streaming quality. However, the existing multicast tree structure is assumed to remain fixed. The goal was to minimize the cost of the upgrade, represented in euro/month. To achieve it, for each peer participating in the transmission, a link type offered by one of the ISPs was selected and overlay trees were constructed, rooted at the source of the content. The authors also present a new heuristic algorithm to efficiently solve this problem. According to experiments, the biggest factor influencing the upgrade cost and determining possible streaming quality values that the system can be upgraded to is the initial tree structure.

Keywords—multicasting, network design, optimization, overlay network, streaming.

1. Introduction

A very important aspect of any kind of design which should be taken into consideration at the time of planning is expansion. This especially applies to the network planning, due to the pace of user's number increase, Internet traffic (from 20000 PB per month in year 2011 to over 80000 PB per month in year 2016 [1]), as well as the growing number of applications and services with the high bandwidth demand. Newly created systems should incorporate such criteria as low cost of deployment, transport efficiency and fault tolerance. In this paper, we focus on the first two factors. We take into consideration real systems and business rights governing the market. This lead us to the development of a new network upgrade scenario – capacity increment with additional nodes and no changes to the existing tree structure.

In our work, we focus on one of the content delivery approaches – multimedia streaming – which has nowadays a significant role in the Internet. Not only isn't it flouting the artist's copyright but it also has a definite advantage over the Internet's major sharing mechanism, in which a user can access a file only once it has been fully downloaded. The overlay multicast meets all the requirements for such transmission without a violation of the physical core [2]. We assume that the overlay multicast is applied for a relatively static applications with a low membership change rate, e.g., videoconferencing, personal video broadcast in small groups, distance learning, collaborated workgroup, delivery of important messages (stocks, weather forecast, emergency alerts) [3]. The stream can increase or decrease a bit rate, depending on the network infrastructure capabilities. This method is called Adaptive Bit Rate Streaming, however it is designed to use unicast or anycast connections. The main reason for the network upgrade is the growing need for the bandwidth, e.g., users wanting higher quality of the video stream, which in turn means a higher bit rate. To answer this demand, the existing network must be incremented.

In this work authors continue research from [4], where three Integer Linear Problems (ILP) were formulated of join optimization of overlay multicast flows and link capacity with the objective to minimize the cost of the network upgrade. Main contributions of this paper are as follows:

- Formulation of the ILP for a new incremental multicast overlay design problem.
- Development of a new heuristic algorithm solving the proposed problem.
- Extensive experiments evaluating the performance of the proposed algorithm against optimal results and showing the behavior of the system as a function of various scenarios including number of trees, initial and final network size and QoS constraint.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 introduces the formulation of the incremental overlay multicast design problem. Section 4 contains a description of the heuristic algorithm. In Section 5 the results of experiments are presented and discussed. Section 6 concludes this work.

2. Related Works

There is an extensive literature about the topology design well covered in [5]. In addition, many surveys on the application layer multicasting have been carried out in [6],

as there is a growing need for applications that will both stream real time content and retrieve on-demand content. However, most of the approaches focus on the optimal overlay multicast topology creation ([7]–[11]). Only few studies concern the incremental approach to the network design ([5], [12]–[16]). The main aim of those studies is to propose algorithms for network design problems considering number of different constraints and objectives. Both topology design and capacity increment coupled with a routing changes approaches are presented.

3. Mathematical Formulation

In this section, a mathematical formulation of the overlay network design problem for the overlay multicasting is presented. Overlay multicast networks are built on top of a general Internet unicast infrastructure rather than point-to-point links, therefore the problem of overlay network design is somewhat different than in networks that do have their own links [17]. The objective is twofold: to determine how much capacity is needed for each user participating in the transmission, and how to economically distribute the streaming content among the participants. The former goal comes to selection of access link types offered by Internet Service Providers, whereas the latter is to construct the overlay multicast trees. Assumptions for the model are taken from our previous works and real systems, therefore continuing the analysis from [18], an approach to consider a new scenario of the system capacity increment with additional nodes is extended. In this manner, the streaming rate of the system has to be incremented and additional nodes are to be added to the system. However, the structure of the existing trees cannot be modified and the link types of existing nodes cannot be worse than the initial ones. For the business reasons this comes as no surprise, because changing the link type to the lower capacity means a contract violation and can end up in additional fees.

Used model is an overlay tree distribution graph with one source of the content, in which we assume a division to multiple substreams of the main stream. Multiple delivery trees are created, each tree carrying a different substream. This approach prevents establishment of a leaf nodes among participating peers, which do not contribute to the overall distribution, and assumes that each peer receives substreams through the different routes. In presented approach, we require each node to be connected to all the trees, and streaming rate of each substream to be equal in amount. However, this scenario can be easily modified and consider a model with nodes receiving the streaming rate of a different quality, i.e., nodes are connected to the different subsets of substream trees and streaming rates of substreams varies. To formulate the problem notation proposed in [15] is used.

We apply a binary decision variable y_{vk} equal to 1, if node v is connected to overlay network by a link of type k and 0 otherwise. Each access link type offered by a given ISP has a particular download capacity (denoted as d_{vk}), upload capacity (denoted as u_{vk}) and cost (denoted by ξ_{vk}).

To construct multicast trees, the following types of decision variables are required: x_{wvt} equal to 1, if there is a link from node (peer) w to node v (no other nodes in between) in the multicast tree t, 0 otherwise. Also x_{wvet} equal to 1, if there is a path from the root node to node e, and it traverses through the link between nodes w and v in the tree t, 0 otherwise.

We also introduce continuous decision variable s_v representing monthly cost of network upgrade of node v. Participants apart from downloading the streaming content in the overlay trees, also take part in the other network services and therefore consume upload and download resources. For this reason, each node v is given a download and upload traffic ratio, denoted by the constants a_v and b_v respectively.

Capacity Increment Model with Additional Nodes (CIMAN)

Indices

v, w, e	$= 1, 2, W, W+1, W+2, \ldots, V$ overlay nodes, where nodes $1, \ldots, W$ are the existing nodes, and $W+1, W+2, \ldots, V$ are additional nodes,
t	$= 1, 2, \ldots, T$ multicast trees,
k, a	$= 1, 2, \ldots, K_v$ access link types for node v.

Constants

a_v	download background transfer of node v (kbit/s),
b_v	upload background transfer of node v (kbit/s),
ξ_{vk}	cost of link of type k for node v (euro/month),
d_{vk}	download capacity of link of type k for node v (kbit/s),
u_{vk}	upload capacity of link of type k for node v (kbit/s),
r_v	= 1 if node v is the root of all trees, 0 otherwise,
q_t	streaming rate of the tree t (kbit/s),
t_{va}	= 1 if node v was connected to the overlay network by a link of type a, 0 otherwise,
z_{wvt}	= 1 if there was a link between node w and v in multicast tree t, 0 otherwise,
M	large number,
H	maximal number of hops from the root node to every additional node in the tree.

Variables

y_{vk}	= 1, if the node v is connected to the overlay network by a link of type k, 0 otherwise (binary),
x_{wvet}	= 1, if there is a path from the root node to node e, and it traverses through the link between nodes w and node v in the multicast tree t, 0 otherwise (binary),
x_{wvt}	= 1, if the link from node w to node v (no other nodes in between) is in the multicast tree t, 0 otherwise (binary),
s_v	cost of upgrading node v (continuous, euro/month).

Objective

The Objective (1) is to minimize the cost of upgrading the network.

$$\text{minimize } F = \sum_v s_v \,. \tag{1}$$

Subject to

Constraint (2) guarantees that for each tree $t = 1, 2, \ldots, T$ each additional node $v = W+1,\ W+2, \ldots, V$ must have exactly one parent node:

$$\sum_{w \neq v} x_{wvt} = 1 \quad v = W+1, W+2, \ldots, V \quad t = 1, 2, \ldots, T\,. \tag{2}$$

Condition (3) assures that there is a path from the root node to additional node e traversing through the link between nodes w and v only if this link exists:

$$x_{wvet} \leq x_{wvt}$$
$$w = 1, 2, \ldots, V \quad v, e = W+1, W+2, \ldots, V \quad t = 1, 2, \ldots, T. \tag{3}$$

Condition (4) represents the flow conservation constraint for the nodes being destination node.

$$\sum_{w \neq v} x_{wvet} - \sum_w x_{vwet} = 1$$
$$v = e \quad e = W+1, W+2, \ldots, V \quad t = 1, 2, \ldots, T. \tag{4}$$

Formula (5) is the flow conservation constraint for the nodes being traversing node.

$$\sum_{w \neq v} x_{wvet} - \sum_w x_{vwet} = 0$$
$$v \neq e \quad r_v = 0 \quad e = W+1, W+2, \ldots, V \quad t = 1, 2 \ldots, T. \tag{5}$$

Equation (6) represents the flow conservation constraint for the node being root node.

$$\sum_{w \neq v} x_{wvet} - \sum_w x_{vwet} = -1$$
$$r_v = 1 \quad e = W+1, W+2, \ldots, V \quad t = 1, 2, \ldots, T. \tag{6}$$

Condition (7) is in the model to assure that each node $v = 1, 2, \ldots, V$ can have only one access link type.

$$\sum_k y_{vk} = 1 \quad v = 1, 2, \ldots, V. \tag{7}$$

Formula (8) is a download capacity constraint and satisfies the requirement of the download capacity of existing nodes $v = 1, 2, \ldots, W$ being greater or equal to the background traffic of a node v and the sum of streaming rates of all the multicast trees the node is connected to.

$$a_v + \sum_{w \neq v} \sum_t z_{wvt} q_t \leq \sum_k y_{vk}\, d_{vk} \quad v = 1, 2, \ldots, W. \tag{8}$$

Condition (9) is a download capacity constraint and satisfies the requirement of the download capacity of additional nodes $v = W+1, W+2, \ldots, V$ being greater or equal to the background traffic of a node v and the sum of streaming rates of all the multicast trees the node is connected to.

$$a_v + \sum_{w \neq v} \sum_t x_{wvt} q_t \leq \sum_k y_{vk} d_{vk} \quad v = W+1, W+2, \ldots, V. \tag{9}$$

Analogously, condition (10) is the upload capacity constraint of existing nodes $w = 1, 2, \ldots, W$, and is equal to the summary upload transfer of w which follows from the number of children nodes, the streaming rate and the background traffic of the node w.

$$b_w + \sum_{v \neq w} \sum_t z_{wvt} q_t \leq \sum_k y_{wk} u_{wk} \quad w = 1, 2, \ldots, W. \tag{10}$$

Constraint (11) is the upload capacity constraint of additional nodes $w = W+1, W+2, \ldots, V$, and is equal to the summary upload transfer of w which follows from the number of children nodes, the streaming rate and the background traffic of the node w.

$$b_w + \sum_{v \neq w} \sum_t z_{wvt} q_t \leq \sum_k y_{wk} u_{wk} \quad w = W+1, W+2, \ldots, V. \tag{11}$$

Formula (12) guarantees that there is no downgrade of the link type for existing nodes.

$$\sum_k t_{vk}\, \xi_{vk} \leq \sum_k y_{vk}\, \xi_{vk} \quad v = 1, 2, \ldots, W. \tag{12}$$

We introduce to the model conditions (13) and (14) which represent the cost of upgrading the access link types in the case of existing nodes and cost of building the network for additional nodes, respectively.

$$\sum_k \sum_a y_{vk}\, t_{va} (\xi_{vk} - \xi_{va}) \leq s_v \quad v = 1, 2, \ldots, W. \tag{13}$$

$$\sum_k y_{vk} \xi_{vk} \leq s_v \quad v = W+1, W+2, \ldots, V. \tag{14}$$

Formula (15) is introduced to meet the QoS requirement of the total length of hops from the root node to every additional node e in the multicast tree t.

$$\sum_w \sum_{v \neq w} x_{wvet} \leq H$$
$$e = W+1, W+2, \ldots, V \quad t = 1, 2, \ldots, T. \tag{15}$$

4. Heuristic Algorithm

In this section a new heuristic algorithm for CIMAN given by Eqs. (1)–(15) is presented. To formulate the algorithm, several additional terms and operators are introduced. All functions presented below are executed using the current state of the problem, i.e., the current values of decision variables, which in effect yield current network flows and access links' capacity. To formulate the algorithm the following definitions are introduced.

Let x_{wvtl} be equal to 1, if in the multicast tree t there is a link from the node w to the node v, and w is located on the level l of the multicast tree t, 0 otherwise.

Transfer between any node w and additional node v is *possible* in the tree t on the level l, if node w is located in the tree t on the level l; the node v is not yet connected to the tree t, and node w has sufficient residual upload capacity to stream the rate of the tree t.
Tree t is *feasible* on the level l, if there's at least one possible transfer from any node w located on the level l, to one of the additional nodes v.
Function $ftree(l)$ returns an index of a feasible tree on the level l. If there is more than one feasible tree, the tree with the lowest number of nodes connected to it is selected.
Let $isfeasible(v,t,l)$ return 1 if the node v is a *feasible parent* node on the level l of the tree t, which means, if at least one transfer in the tree t between the node v on the level l and any other additional node is possible.
Function $fpnode(t,l)$ returns an index of a feasible parent node located on the level l of the tree t. If there's more than one feasible parent node, the node with the largest value of residual upload capacity is selected. Notice that if $l=1$, $fpnode(t,l)$ always returns an index of the root node.
Let $fcnode(v,t,l)$ return an index of a feasible child node of the node v located on the level l of the tree t. If there is more than one feasible child node, again the additional criterion is the residual upload capacity.
Function $istransfer(l)$ returns 1 if there is at least one possible transfer on the level l of any tree. Otherwise it returns 0.
Let $istree()$ return 1 if each additional node $v=W+1$, $W+2,\ldots,V$ is connected to each tree $t=1,2,\ldots,T$ (all required transfers are completed), 0 otherwise.
Function $isupdate()$ returns 1 if incrementing the upload capacity of the access link is possible for at least one node. Otherwise it returns 0.
Let $istreetransfer()$ return 1 if there is a node v with sufficient upload capacity to provide at least one transfer in any tree, 0 otherwise.
Function $updatenode()$ returns an index of a node v, for which the upload capacity can be augmented. If there is more than one such a node, an additional criterion is applied, i.e., in the algorithm, several combinations of three values are used: the access link price, the node level and the relative cost of the upload capacity increase given by the formula $(u_{v(k+1)}-u_{vk})/(\xi_{v(k+1)}-\xi_{vk})$.
Set E denotes nodes updated after every iteration of the main loop of the algorithm.

4.1. Minimizing Cost of Upgrading the Network Heuristic Algorithm

Step 0. Load the existing tree structure and set $x_{wvtl}=1$ for such nodes w, v, tree t and level l, that there is a link from existing node w to existing node v, and w is located on the level l of the multicast tree t.

Step 1. Create table E.

Step 2. Set $x_{wvtl}=0$ for each $v=W+1,\ W+2,\ldots,V$, $w=1,2,\ldots,V$, $t=1,2,\ldots,T$, $l=1,2,\ldots,L$. Set y_{vk} as the minimal values that guarantee sufficient download capacity for each node $v=W,W+1,\ldots,V$ (i.e., $d_{vk}\geq a_v+Tq_t$) except for the root node as well as nodes from the table E, and the sufficient upload capacity for the root node ($r_v=1$), nodes from table E and existing nodes $v=1,2,\ldots,W$ (i.e., $u_{vj}\geq b_v+Tq_t$).

Step 3. Set $l=1$.

(a) Let $t=ftree(l)$. If $isfeasible(r,t,l)=0$ set $l=l+1$ and go to Step 4. Otherwise, go to Step 3b.

(b) Calculate $w=fcnode(r,t,l)$ and set $x_{rwtl}=1$. Go to Step 3a.

Step 4. If $istreetransfer()=0$ and $istree()=0$ go to Step 5. If $istree()=1$, go to Step 7, otherwise:

(a) If $istransfer(l)=0$ set $l=l+1$ and go to Step 4. Otherwise go to Step 4b.

(b) Set $t=ftree(l)$, $w=fpnode(t,l)$, $v=fcnode(w,t,l)$ and $x_{wvtl}=1$. Go to Step 4a.

Step 5. If $isupdate()=1$, go to Step 6. Otherwise stop the algorithm, there is no feasible solution.

Step 6. Set $e=updatenode()$. Find k, for which $y_{ek}=1$. Set $y_{ek}=0$, $k=k+1$, $y_{ek}=1$, $l=1$, update table E, and go to Step 2.

Step 7. Find all nodes $v=1,2,\ldots,W,W+1,\ldots,V$ with the unused upload capacity and decrease it if possible. Set $y_{vk}=0$, $k=k-1$, $y_{vk}=1$. Go to Step 8.

Step 8. Calculate the cost of upgrading the network denoted as C, as the sum of link type upgrade cost for existing nodes $v=1,2,\ldots,W$ $(y_{vk}\xi_{vk}-t_{va}\xi_{va})$, and used access link type prices for each additional node $v=W+1,W+2,\ldots,V$ $(y_{vk}\xi_{vk})$. Go to Step 9.

Step 9. Stop the algorithm. The cost of network upgrade is equal to C.

The main idea of the Minimizing Cost of Upgrading the Network Heuristic Algorithm is as follows. The algorithm starts with loading the existing network structure and setting initial connections between existing nodes. Variable x_{wvtl} is set to 1 for such nodes w, v, tree t and level l, that there is a link from the existing node w to the existing node v, and w is located on the level l of the multicast tree t (Step 0). Then, we move on to the creation of a table to store updated nodes' indices (Step 1), which is updated after every access link type increase (Step 6).
In Step 2, an initialization of all of the remaining variables x_{wvtl} and variables y_{vk} is proceeded. The idea behind the selection of the latter is to find for each node a link that has a sufficient download capacity to transmit the background traffic and the overall streaming rate of multicast trees. For the root node, existing nodes and updated nodes, an additional procedure is run to ensure the satisfactory upload capacity to fulfill the transmission. Next, in Step 3, we check if there are any possible connections from the source of the content to additional nodes $v=W+1,W+2,\ldots,V$ in

each tree $t = 1, 2, \ldots, T$. If the root node has enough residual upload capacity the connection is established. Step 4 creates multicast trees denoted by variables x_{wvtl}. The main loop of Step 4 is repeated for the subsequent tree levels to allocate the resources of the upload capacity proportionally to all of the trees. If after Step 4, all nodes are connected to each tree, the algorithm tries to decrease the access links of every node (Step 7), and calculates the cost of upgrading the network (Step 8). Otherwise, there is an attempt to increase the capacity of the access link of the selected node (Step 6) and the network is rebuilt. Once all of the additional nodes $v = W+1, W+2, \ldots, V$ are connected, the algorithm stops.

5. Results

We implemented the presented heuristic algorithm in C++. The goal of numerical experiments was to examine the performance of presented approach against the traditional approach, and the heuristic approach against the optimal results. In all the experiments, we use DSL price lists of three ISPs operating in Poland (TP, Dialog and UPC) with prices in euro/month. To each node we randomly assign one of the ISPs, so that access link can be chosen from the pool of available options. The values of the download background traffic were selected at random in the range from 1024 to 2048 kbit/s. Analogously, the values of the upload background traffic were selected at random in the range from 256 to 512 kbit/s.

In order to obtain optimal results we solved the CIMAN using CPLEX 12.5 solver [19]. Due to the complexity of the problem, we decided to test the networks consisting of up to 25 initial and up to 40 final overlay nodes (peers), in order to obtain close to optimal results in a reasonable time. The streaming rate in the examined system was divided proportionally to 1–3 substreams. We introduced additional constraint following from the real systems, which assumed limitation of the number of hops from the source of the content to any additional node, in the range of 2–7, and is Quality of Service type of constraint. Tests were run for a fixed root node location and selection of ISP. Number of the final nodes was set to 15–40, depending on the size of initial system. Initial networks consisted of 15–25 nodes, 1–3 trees. For our investigation, we considered four different streaming bit rates corresponding to four different qualities of the video stream: 320p, 480p, 576p and 720p (HD). To compute the streaming rate to be distributed we used On2 VP6 video codec, NTSC frame rate, average motion, 16:9 aspect ratio, mp3 audio codec, stereo channels, medium audio quality and 44.1 kHz sampling rate. Due to the limitations of maximal upload capacities available from ISPs, we couldn't achieve Full HD quality stream using our approach. Note, that since the structure of the initial trees couldn't be modified, for some scenarios where the low quality stream (320p) was to be upgraded to the high quality stream (576p or 720p), the transmission was impossible. In total, 972 different scenarios were considered. We introduced a computation time limit of one hour for CPLEX solver, therefore in some cases no solution or only a feasible solution instead of the optimal one was found.

Table 1 shows the comparison of the CPLEX results and ones delivered by the CIMAN heuristic algorithm for the scenario with 10 initial nodes and 576p initial streaming quality. Due to the fixed structure of initial trees, achieving HD streaming quality was impossible for this scenario. Column 1 represents the number of multicast trees (T), column 2 is the number of final nodes (V), column 3 is a hop limit constraint and describes the maximal number of hops from the source of the content to the additional node (H), column 4 is the end quality of the stream (EQ). Columns 5–6 are related to the increment cost in euro/month for optimal (OPT) and heuristic ($HEUR$) approach, respectively. Column 7 is related to comparison of those two approaches (GAP), whereas columns 8–9 give computation time of CPLEX solver and the heuristic algorithm, respectively.

Table 1
Heuristic algorithm versus CPLEX results for initial streaming quality of 576p and 10 initial nodes

T	*V*	*H*	*EQ*	*CPLEX* [euro/month]	*HEUR* [euro/month]	*GAP* [%]	*CPLEX* Time [s]	*HEUR* Time [s]
1	15	2	567p	73	73	0.00	0.03	0
1	15	3	576p	63	63	0.00	0.21	0
1	15	4	576p	60	60	0.00	0.14	0
1	15	5	576p	60	60	0.00	0.09	0
1	15	6	576p	60	60	0.00	0.14	0
1	15	7	576p	60	60	0.00	0.14	0
2	40	3	567p	367	382	–4.09	1635	0.03
2	40	4	576p	*INF*	382	*INF*	3600	0.05
2	40	5	576p	*INF*	367	*INF*	3600	0.01
2	40	6	576p	*INF*	367	*INF*	3600	0.01
2	40	7	576p	498	367	26.31	3600	0.03
3	35	3	567p	*INF*	307	*INF*	3600	0.03
3	35	4	576p	*INF*	302	*INF*	3600	0.01
3	35	5	576p	*INF*	302	*INF*	3600	0.02
3	35	6	576p	297	302	–1.68	3328	0.01
3	35	7	576p	370	302	18.38	3600	0.02

On average, for experiments that feasible solution was delivered by CPLEX, the proposed heuristic approach delivers solutions 0.9% worse than optimal ones. For networks consisting of one tree, two trees and three trees, CIMAN heuristic algorithm delivers solutions 1.3%, 0.6% and 0.8% worse, respectively. In 582 cases, the heuristic approach yields the results equal to the ones delivered by CPLEX, and in 79 scenarios CPLEX doesn't deliver any solution after one hour of computation. In 44 cases. the proposed algorithm outperforms feasible results obtained by CPLEX solver (with 3600 seconds execution time limit), also for 27 scenarios CPLEX yields "out of memory" error. We can easily notice that the heuristic approach

significantly outperforms CPLEX when it comes to the computation time. When the complexity of the problem increases (more nodes and trees), the heuristic approach is even 30000 times faster. Also, the computation of the proposed algorithm is finished within the fraction of a second for most of the experiments. Tests with more complex networks lead to expanding computation time, which was still relatively short.

Table 2
Upgrade cost for CIMAN

T	*W*	*V*	*H*	*IQ*	*IP* [euro/month]	Upgrade cost [euro/month] 360p	480p	576p
1	10	35	2	360p	110	275	–	–
1	10	35	3	360p	110	267	294	–
1	10	35	4	360p	110	265	287	–
1	10	35	5	360p	110	265	287	–
1	10	35	6	360p	110	265	287	–
1	15	30	3	480p	170	n/a	178	220
1	15	30	4	480p	170	n/a	178	211
1	15	30	5	480p	170	n/a	174	204
1	15	30	6	480p	170	n/a	171	204
1	15	30	7	480p	170	n/a	166	204
2	10	40	3	360p	109	320	374	–
2	10	40	4	360p	109	314	372	–
2	10	40	5	360p	109	309	372	–
2	10	40	6	360p	109	307	372	–
2	10	40	7	360p	109	307	372	–
2	10	40	8	360p	109	307	372	–
2	25	40	2	480p	282	n/a	178	–
2	25	40	3	480p	282	n/a	178	208
2	25	40	4	480p	282	n/a	173	208
2	25	40	5	480p	282	n/a	166	208
2	25	40	6	480p	282	n/a	166	208
3	20	40	2	360p	213	215	299	–
3	20	40	3	360p	213	209	298	–
3	20	40	4	360p	213	203	297	–
3	20	40	5	360p	213	203	296	–
3	20	40	6	360p	213	203	296	–
3	25	35	2	480p	280	n/a	118	181
3	25	35	3	480p	280	n/a	118	171
3	25	35	4	480p	280	n/a	115	168
3	25	35	5	480p	280	n/a	112	168
3	25	35	6	480p	280	n/a	112	168

Table 2 presents the upgrade cost in euro/month delivered by CPLEX solver, for different incremental scenarios. Column 1 represents the number of trees, columns 2–3 give the number of initial (*W*) and final nodes, column 4 is the hop limit constraint, column 5 gives the initial streaming quality (*IQ*). Column 6 shows the initial price of building the network in euro/month (*IP*), whereas columns 7–9 give the upgrade cost in euro/month to stream the quality of 360p, 480p and 576p, respectively. We can see, that using fixed initial tree structures limits the end quality that the system can be upgraded to. For the networks with the initial streaming quality of 360p, upgrades to 576p streaming quality are impossible. Moreover, upgrade to HD streaming quality (720p) cannot be achieved for any scenario. Also, the number of maximal hops between the source node and any additional node is a factor. Increasing it decreases the cost of the upgrade, plus when limited to 2, for some of the scenarios the transmission is impossible.

The second goal of experiments was to test the behavior of the systems with a larger number of participating nodes. Using the proposed heuristic we examined networks consisting of up to 200 initial nodes and 250 final nodes. Note that CPLEX solver cannot provide feasible results for such large networks, therefore to generate the initial network structures we used our different heuristic algorithm.

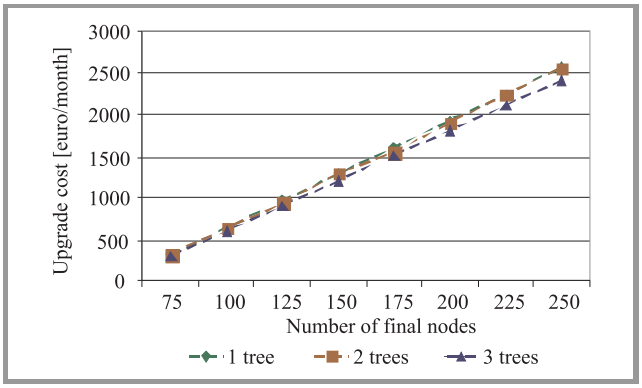


Fig. 1. Upgrade cost as a function of number of trees and number of final nodes (50 initial nodes, 576p stream quality).

Figure 1 shows the impact of introducing more trees to the system for the network consisting of 50 initial nodes, 576p streaming quality and the final network size of 75–250 nodes. Systems with three multicast trees show greater difference in price range, which comes up to over 150 euro/month, whereas the cost of upgrading the systems with one or two multicast trees is comparable.

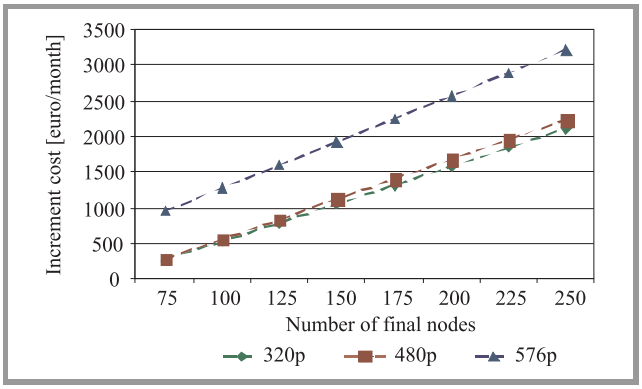


Fig. 2. Upgrade cost as a function of number of final nodes and end stream quality (50 initial nodes, 1 tree, 320p initial stream quality).

Figure 2 depicts the upgrade cost for the initial network consisting of 50 nodes, one tree and 320p streaming qual-

Table 3
Traditional versus incremental approach for initial stream quality of 360p

T	*W*	*V*	*EQ*	*TA*	*UC*	*IA*	*GAP*
				[euro/month]			[%]
1	10	15	360p	170	54	170	0.00
1	10	20	360p	229	113	229	0.00
1	10	25	360p	282	166	282	0.00
1	10	15	480p	183	70	186	–1.64
1	10	20	480p	251	138	254	–1.20
1	10	25	480p	311	198	314	–0.96
2	10	15	360p	159	50	159	0.00
2	10	20	360p	213	104	213	0.00
2	10	25	360p	262	153	262	0.00
2	10	15	480p	169	92	201	–18.93
2	10	20	480p	229	151	260	–13.54
2	10	25	480p	282	206	315	–11.70
3	10	15	360p	159	50	159	0.00
3	10	20	360p	213	104	213	0.00
3	10	15	480p	169	86	201	–18.93
3	10	20	480p	226	144	259	–14.60

Table 4
Traditional versus incremental approach for initial stream quality of 360p

T	*W*	*V*	*EQ*	*TA*	*UC*	*IA*	*GAP*
				[euro/month]			[%]
1	50	75	360p	796	263	796	0.00
1	50	100	360p	1058	526	1059	–0.09
1	50	125	360p	1322	789	1322	0.00
1	50	150	360p	1585	1052	1585	0.00
1	50	175	360p	1850	1315	1848	0.11
1	50	200	360p	2113	1578	2111	0.09
1	50	225	360p	2376	1841	2374	0.08
1	50	250	360p	2639	2106	2639	0.00
1	150	175	360p	1850	263	1848	0.11
1	150	200	360p	2113	526	2111	0.09
1	150	225	360p	2376	789	2374	0.08
1	150	250	360p	2639	1054	2639	0.00
2	100	125	360p	1294	258	1294	0.00
2	100	150	360p	1552	515	1551	0.06
2	100	175	360p	1809	773	1809	0.00
2	100	200	360p	2067	1030	2066	0.05
2	100	225	360p	2325	1288	2324	0.04
2	100	250	360p	2582	1546	2582	0.00
2	100	125	480p	1402	703	1823	–30.03
2	100	150	480p	1679	956	2076	–23.65
2	100	175	480p	1956	1227	2347	–19.99
2	100	200	480p	2238	1506	2626	–17.34
2	100	225	480p	2515	1783	2903	–15.43
2	100	250	480p	2797	2062	3182	–13.76
3	150	175	360p	1938	275	1936	0.10
3	150	200	360p	2213	552	2213	0.00
3	150	225	360p	2490	827	2488	0.08
3	150	250	360p	2765	1104	2765	0.00
3	150	175	480p	2121	665	2484	–17.11
3	150	200	480p	2423	918	2737	–12.96
3	150	225	480p	2725	1199	3018	–10.75
3	150	250	480p	3027	1488	3307	–9.25

ity, to networks of up to 250 nodes and three different streaming qualities: 320p, 480p and 576p. There is slight price difference when upgrading from 320p streaming quality to 480p, however obtaining mid quality stream (576p) from low (320p) initial stream is about twice as expensive. Due to the fixed initial tree structure, HD streaming quality cannot be delivered.

Comparison of the traditional approach versus the incremental approach is presented in Table 3 and Table 4 for small and large networks, respectively. Column 1 represents the number of trees, columns 2–3 give the number of initial and final nodes, column 4 is the end quality stream. Column 5 (*TA*) is the cost of building the network using the traditional approach in euro/month, whereas columns 6–7 give the upgrade cost (*UC*) and the total price (*IA*) in euro/month of building the network using the incremental approach, respectively. Column 7 is related to the comparison of those two approaches.

The results show for both small and large network sizes, that upgrading the network in the sense of introducing to the system more participating peers without the change of the streaming quality, brings almost the same price as using the traditional approach (building the network from the scratch). For larger networks (Table 4), where CPLEX couldn't deliver optimal results and the heuristic algorithm to generate the input data was used, for some of the scenarios the incremental approach is slightly better. This is caused by the fact, that the heuristic approach brings close to optimal results, and there is always room for the improvement. The second trend is seen when introducing more nodes to the system, combined with the streaming quality increase. This contributes to traditional approach outperforming the incremental one by up to 30%. This is again caused by the fixed initial multicast tree structure creating bottlenecks for faster transmission.

6. Conclusion

This paper addressed the problem of Capacity Increment Approach with Additional Nodes and no existing tree modifications. The objective of the optimization was to minimize the cost of upgrading the system. Authors proposed a new heuristic algorithm and illustrated this approach by showing the results using both CPLEX solver and newly proposed heuristic. In numerical experiments different incremental scenarios were considered. Results delivered by proposed algorithm were comparable to the solutions yielded by CPLEX. Moreover, for networks consisting of the larger number of nodes, CPLEX solver couldn't provide feasible solutions in one hour time limit, and either couldn't find any solution or yielded out-of-memory exception. According to the obtained results, we can conclude, that the biggest factor influencing the upgrade cost is the initial tree structure, which is the bottleneck for the bigger throughput and prevents the streaming quality upgrade. Other parameters, like the maximal num-

ber of hops from the source of the content to any of the newly connected nodes, have smaller influence on the upgrade cost.

References

[1] "Cisco visual networking index: Forecast and methodology, 2011–2016", Cisco, 2012.

[2] S. Banerjee *et al.*, "Omni: An efficient overlay multicast infrastructure for real-time applications", *Comp. Netwo.: The Int. J. Comp. Telecommun. Netw.*, vol. 50, iss. 6, pp. 826–841, 2006.

[3] J. Bufford, H. Yu, and E. K. Lua, *P2P Networking and Applications*. Morgan Kaufmann Series in Networking, D. Clark, Ed. Morgan Kaufmann, 2008.

[4] M. Szostak and K. Walkowiak, "Incremental approach to Overlay Network Design Problem", in *Proc. 17th Polish Teletraf. Symp. 2012*, Zakopane, Poland, 2012.

[5] M. Pióro and D. Medhi, *Routing, Flow and Capacity Design in Communication and Computer Networks*. Morgan Kaufman, 2004.

[6] T. Small, B. Li, and B. Liang, "Outreach: Peer-to-peer topology construction towards minimized server bandwidth costs", *IEEE J. Selec. Areas Commun.*, vol. 25, no. 1, pp. 35–45, 2007.

[7] C. Wu and B. Li, "On meeting P2P streaming bandwidth demand with limited supplies", in *Proc. 15th Ann. SPIE/ACM Int. Conf. Multime. Comput. Netwo. MMCN 2008*, San Jose, CA, USA, 2008.

[8] B. Akbari, H. Rabiee, and M. Ghanabari, "An optimal discrete rate allocation for overlay video multicasting", *Comp. Commun.*, vol. 31, pp. 551–562, 2008.

[9] Y. Cui, Y. Xue, and K. Nahrstedt, "Optimal resource allocation in overlay multicast", *IEEE Trans. Parall. Distrib. Sys.*, vol. 17, no. 8, pp. 808–823, 2006.

[10] M. Kwon and S. Fhamy, "Path-aware overlay multicast", *Comp. Netw.*, vol. 47, no. 1, pp. 23–45, 2005.

[11] Y. Zhu and B. Li, "Overlay networks with linear capacity constraints", *IEEE Trans. Parall. Distrib. Sys.*, vol. 19, no. 2, pp. 159–173, 2008.

[12] N. Geary, A. Antonopoulos, E. Drakopoulos, and J. O'Reilly, "Analysis of optimization issues in multi-period DWDM network planning", in *Proc. 20th Ann. Joint Conf. IEEE Comp. Commun. Soc. IEEE INFOCOM 2001*, Anchorage, Alaska, USA, 2001, vol. 1, pp. 152–158.

[13] S. Gopal and K. Jain, "On network augumentation", *IEEE Trans. Reliabil.*, vol. 35, no. 5, pp. 541–543, 1986.

[14] H. Lee and D. Dooly, "Heuristic algorithms for the fiber optic network expansion problem", *Telecommun. Sys.*, vol. 7, pp. 355–378, 1997.

[15] A. Meyerson, K. Munagala, and S. Plotkin, "Designing networks incrementally", in Proc. *42nd Ann. Symp. Foundations Comp. Sci. FOCS 2001*, Las Vegas, Nevada, USA, 2001.

[16] A. Tero *et al.*, "Rules for biologically inspired adaptive network design", *Science*, vol. 327, pp. 439–442, 2010.

[17] S. Shi and J. Turner, "Multicast routing and bandwidth dimensioning in overlay networks", *IEEE J. Selec. Areas Commun.*, vol. 45, no. 8, pp. 1444–1455, 2002.

[18] M. Szostak and K. Walkowiak, "Two approaches to network design problem for overlay multicast with limited tree delay – model and optimal results", *Int. J. Electron. Telecommun.*, vol. 57, no. 3, pp. 335–340, 2011.

[19] "Ilog cplex version 12.5 user's manual", IBM Software.

Maciej Szostak received his M.Sc. degree from the Wrocław University of Technology in 2008. Currently, he is a Ph.D. student at the Department of System and Computers Networks, Faculty of Electronics, Wrocław University of Technology. His research interests include overlay networks, peer-to-peer computing, multimedia service delivery and virtualization. He is a member of reviewer committees of three international conferences on simulations and computer systems and reviewer in two international journals.
E-mail: maciej.szostak@pwr.wroc.pl
Department of Systems and Computer Networks
Wrocław University of Technology
Wybrzeże Wyspiańskiego st 27
50-370 Wrocław, Poland

Krzysztof Walkowiak – for biography, see this issue, p. 64.

Paper

Performance Evaluation of the MSMPS Algorithm under Different Distribution Traffic

Grzegorz Danilewicz and Marcin Dziuba

Faculty of Electronics and Telecommunications, Poznan University of Technology, Poznan, Poland

Abstract—In this paper, the Maximal Size Matching with Permanent Selection (MSMPS) scheduling algorithm and its performance evaluation, under different traffic models, are described. In this article, computer simulation results under nonuniformly, diagonally and lin-diagonally distributed traffic models are presented. The simulations was performed for different switch sizes: 4×4, 8×8 and 16×16. Results for MSMPS algorithm and for other algorithms well known in the literature are discussed. All results are presented for 16×16 switch size but simulation results are representative for other switch sizes. Mean Time Delay and efficiency were compared and considered. It is shown that our algorithm achieve similar performance results like another algorithms, but it does not need any additional calculations. This information causes that MSMPS algorithm can be easily implemented in hardware.

Keywords—connection pattern, diagonally distributed traffic, lin-diagonally distributed traffic, MQL matrix, non-uniformly distributed traffic, switching fabric.

1. Introduction

Several well known scheduling algorithms have been proposed in the literature [1]–[6]. All these algorithms, which are responsible for configuration of a switching fabric, are very sophisticated and they achieve a good efficiency and short time delay. During designing of a new algorithm, a theoretical approach is applied. It means that designers do not pay attention to algorithm implementation constraints. Most of well known algorithms, which achieve the good performance results, are very difficult for implementation in the real switching fabric hardware. This is due to very complicated calculations which must be performed during algorithms work. The high calculations complexity makes this algorithms impractical. Instead, most of the new generation switches and routers use much simpler scheduling algorithms to control and configure switching fabric. One of this kind of algorithms is MSMPS [7], which achieve the similar performance results like the rest of algorithms but does not need to perform a lot of complicated calculation.

Other important fact, which influence on switches and routers performance, is switching fabric buffers architecture. In our research we study a switching fabric with VOQ (Virtual Output Queue) system [6], [8]. This buffering system has been proposed to solve a HOL (Head of Line) effect. In VOQ system each switching fabric input has a separate queue for a packet directed to particular output of a switching fabric. Using this kind of architecture, its performance depends only on a good scheduling algorithm. Algorithm should be very fast, achieve the good results (high efficiency and short time delay) and be easy to implement in hardware.

Before each packet will be send through the switch, it should be decided which packet, from which VOQ will be chosen. This decision is taken in each time slot – the basic unit of time in simulation environment. To solve this problem in hardware, a few scheduling mechanisms are used. There are three basic methods: random selection, first in first out (FIFO) and round-robin. In the presented architecture centralized scheduling mechanism is used. In this mechanism all decisions considering setting up connections between switching fabric inputs and outputs (connection patterns) are made by algorithm or driver implemented in a separated control module. Driver can control some connected switching fabrics located in different equipments (i.e., routers). Such solution can be used in the new generation networks for example in Software Defined Networks (SDN) [9]. Routers are responsible for direct packets in data paths but high level decisions (routing) are moved to separate module or device which is located out of routers. Routing decisions are sent to routers to execute suitable connection patterns in each switching fabric of each router. Centralized scheduling mechanism has a huge advantage over traditional scheduling mechanism. In todays network nodes, where 10 Gbit/s ports are used, each time slot is equivalent to the 50 ns. This time in not enough to realize traditional scheduling mechanism, which based on sending control signal. This signal consists of three parts: demand, confirmation and acceptance. Nowadays, all algorithms are designed in such a way, that the number of control signals is minimized. The best solution is sending only one signal between control module and switching fabric. All this things are fulfilled by MSMPS algorithm.

This paper is organized as follows. In Section 2, the switch architecture is discussed. In Section 3 of this article, all simulation parameters are explained. In Section 4 traffic distribution models which are used in our research, are described. Then in Section 5 computer simulation results under different traffic patterns are shown. Results achieved

for MSMPS algorithm, are compared with another algorithms well known in the literature. In Section 6, same conclusion are given.

2. Switch Architecture

The general VOQ switch architecture is presented in Fig. 1 [10].

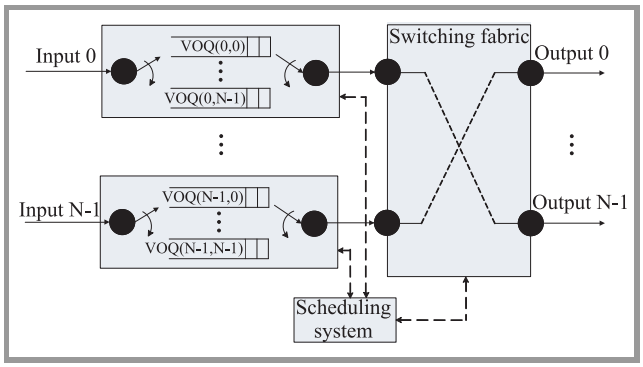


Fig. 1. General VOQ switch architecture.

In our research we use switching fabric with input queuing system (Input Queued switches), where buffers are placed at the inputs. Each input has separated queue which is divided into N independent VOQs. The total number of virtual queues depends on the number of inputs and outputs. It was assumed that in presented switch, the number of inputs and outputs is equal and in general case is N. Based on this assumption, total number of VOQs in switching fabric, with N number of inputs/outputs, is equal to N^2. Additionally, each virtual queue is denoted by VOQ (i,j), where i is the input port number and j is the output port number. It can be assumed that: $0 \leq i \leq N-1$ and $0 \leq j \leq N-1$.

Between inputs and outputs modules, the switching fabric is placed. In the switching fabric, there are electrical or optical equipments which have to be properly configured when all connections between inputs and outputs are established. Implemented algorithm is responsible for a proper configuration of mentioned equipments.

The most important module, in presented symmetrical switch, is scheduling system module. This is a module, where algorithms are implemented. In the scheduling module all information about queues conditions are stored. It means that scheduling system has knowledge about numbers of packets waiting in all queues, to be send through the switch. This information is necessary to make a right decision by MSMPS algorithm about connection pattern in the switching fabric.

3. Algorithm Description

MSMPS algorithm is based on permanent connections pattern between inputs and outputs. For example, from Fig. 2, connection pattern for 4×4 switch can be observed.

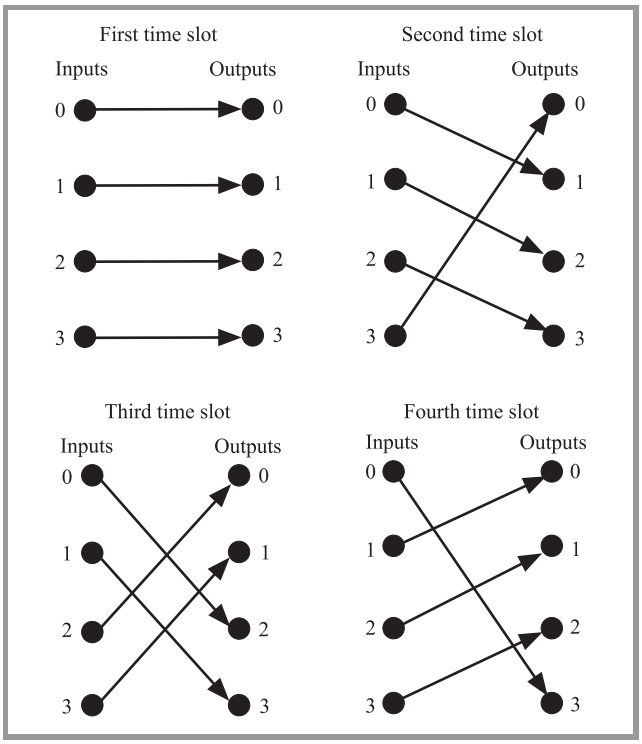


Fig. 2. Connection pattern for 4×4 switch.

Permanent connections pattern provides fair access to the each output. It means that all outputs in switch are treated equally. As mentioned before, scheduling module has information about VOQ conditions. This information is stored in MQL matrix (Matrix of Queue Lengths). This kind of matrix was the easiest way to store this information. Figure 3 shows MQL matrix for 4×4 switch.

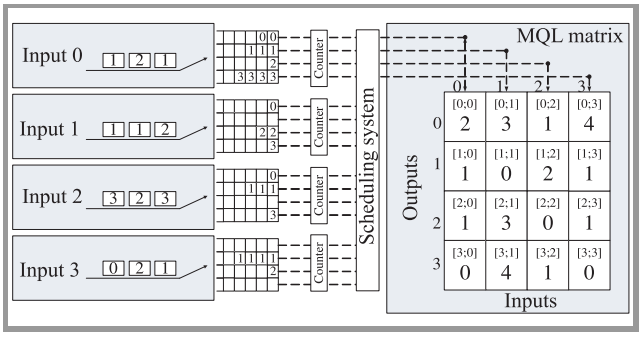


Fig. 3. MQL matrix for 4×4 switch.

Information is updated in each time slot. Each cell (one position in matrix) in matrix MQL and each VOQ has unique address. This correlation allows attribute one cell to one VOQ. For example cell [0;0] corresponds to the VOQ (0,0). In cell [0;0] information about number of packets waiting in VOQ (0,0) are stored. If there is no packets in VOQ, suitable position in matrix is filled by 0. It can be seen from Fig. 3 can be observed that matrix has N rows and N columns. It corresponds to the 4×4 switch, which is presented in our example. Based on permanent connections and information, stored in MQL matrix,

MSMPS algorithm makes decisions about connections to be set up in switching fabric. The main purpose is to avoid empty connections. Empty connection means that there is no packets to be send from an input to an output. Algorithm gives priority to the most filled VOQs. More details about MSMPS algorithm can be found in [7].

4. Simulation Conditions

In this paper, performance results for some scheduling algorithms, well known in the literature, and for MSMPS algorithm are presented. All graphs are plotted as the results of computer simulations. Packets are incoming at the inputs according to Bernoulli arrival model [11], [12]. Under this model, only one packet can arrive at the input in each time slot (basic of time unit). It was assumed that one packet may occupy only one time slot. In Bernoulli model, probability that packet will arrive at the input is equal to p, where:

$$p\,\varepsilon(0 < p \leq 1)\,. \tag{1}$$

Simulation results are presented as a mean value of ten independent simulation runs. Number of iteration in one simulation run is equal to 500,000, where the first 30,000 steps are reserved for obtaining convergence in the simulation environment. It was assumed also that our switching fabric is strict sense nonblocking. It means that there is always possible to establish connection between each suitable and idle input and suitable and idle output of the switching fabric. Performance results consider the efficiency and Mean Time Delay parameters.

Efficiency is parameter which was calculated according to Eq. (2). Numerator is the number of packets passed in n-th time slots through the switching fabric. Denominator is the number of packets which have arrived to the switch buffers in n-th time slot [7].

$$q = \frac{\sum\limits_n a_n}{\sum\limits_n b_n}\,, \tag{2}$$

where:

n – time slot number,

a_n – number of packets passed in n time slot through the switching fabric,

b_n – number of packets which can be send through the switching fabric in n time slot.

Mean Time Delay (MTD) is calculated according to Eq. (3). Numerator is a sum of difference between time when a packet is transferred by the switch and the time when the packet has arrived to the buffer system. Denominator is a total number of packets served by the switching fabric.

$$MTD = \frac{\sum\limits_n t_{out} - t_{in}}{\sum\limits_n k_n}\,, \tag{3}$$

where:

MTD – Mean Time Delay,

n – time slots number,

t_{in} – time when a packet arrived to the VOQ,

t_{out} – time when the same packet is transferred by the switching fabric,

k – number of packets.

Three distributed traffic models were taken into account in this paper. Each of this model determines the probability that packet which appear at the input, will be directed to the certain output. These considered traffic models are described in following subsections.

4.1. Non-uniformly Distributed Traffic

The probability of the packet arriving at the input i, directed to the output j is presented in Table 1. For readability, table shows traffic distribution in 4×4 switch. Analogous traffic distribution is used for other switch sizes: 8×8 and 16×16. It can be observed from Table 1 that in this type of traffic model, some outputs have higher probability of being selected [13]. This probability can be defined as: p_{ij} and it can be calculated according to the Eq. 4:

$$p_{ij} \begin{cases} \dfrac{1}{2} & \text{for } i = j, \\[2ex] \dfrac{1}{2(N-1)} & \text{for } i \neq j. \end{cases}\,, \tag{4}$$

where:

N – number of switch inputs/outputs.

Table 1
Non-uniformly distributed traffic in 4×4 switch with VOQ

	Output 0	Output 1	Output 2	Output 2
Input 0	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$
Input 1	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{6}$
Input 2	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{6}$
Input 3	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$

4.2. Diagonally Distributed Traffic

In this type of distribution model, the traffic is concentrated in two diagonals of the table (traffic matrix). The probability that packet is appeared at the suitable input i and it will be directed to the output j is equal to $p_{ij} = \frac{1}{2}$. Probability for the rest of inputs (not placed in two diagonals) is $p_{ij} = 0$ [12], [14]–[16]. From Table 2 it can be observed that input i has packets only for output i and for output $((i + (N\text{-}1)) \bmod N)$.

Table 2
Diagonally distributed traffic in 4×4 switch with VOQ

	Output 0	Output 1	Output 2	Output 2
Input 0	$\frac{1}{2}$	0	0	$\frac{1}{2}$
Input 1	$\frac{1}{2}$	$\frac{1}{2}$	0	0
Input 2	0	$\frac{1}{2}$	$\frac{1}{2}$	0
Input 3	0	0	$\frac{1}{2}$	$\frac{1}{2}$

4.3. Lin-diagonally Distributed Traffic

Lin-diagonally distributed model is a modification of diagonally distributed model. Considered lin-diagonally model and its probabilities are presented in Table 3. It can be seen from this table that a load decrease linearly from one diagonal to the other. In general case, probability can be calculated according to the following formula [17]:

$$p_d = p\frac{N-d}{N(N+1)/2} \tag{5}$$

with $d = 0,\ldots,N-1$, then $p_{ij} = p_d$ if $j = (i+d) \bmod N$, and where:

p_d – probability of packet arriving in lin-diagonally distributed traffic,

p – probability of packet arriving in Bernoulli, process,

N – number of switch inputs/outputs,

d – output number.

Table 3
Lin-diagonally distributed traffic in 4×4 switch with VOQ

	Output 0	Output 1	Output 2	Output 2
Input 0	$\frac{4}{10}$p	$\frac{1}{10}$p	$\frac{2}{10}$p	$\frac{1}{10}$p
Input 1	$\frac{3}{10}$p	$\frac{4}{10}$p	$\frac{1}{10}$p	$\frac{2}{10}$p
Input 2	$\frac{2}{10}$p	$\frac{3}{10}$p	$\frac{4}{10}$p	$\frac{1}{10}$p
Input 3	$\frac{3}{10}$p	$\frac{2}{10}$p	$\frac{3}{10}$p	$\frac{4}{10}$p

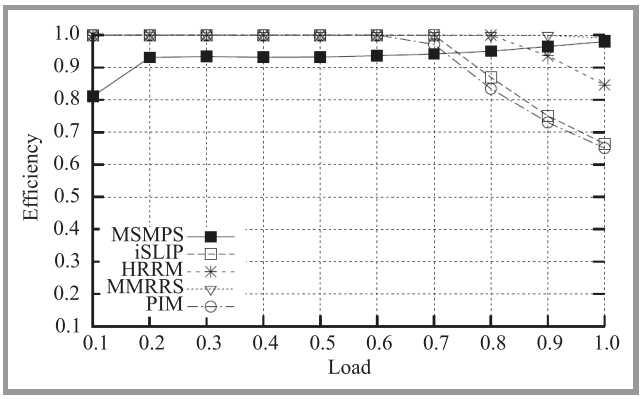


Fig. 4. The efficiency for Bernoulli arrivals with nonuniformly distributed traffic in 16×16 switches.

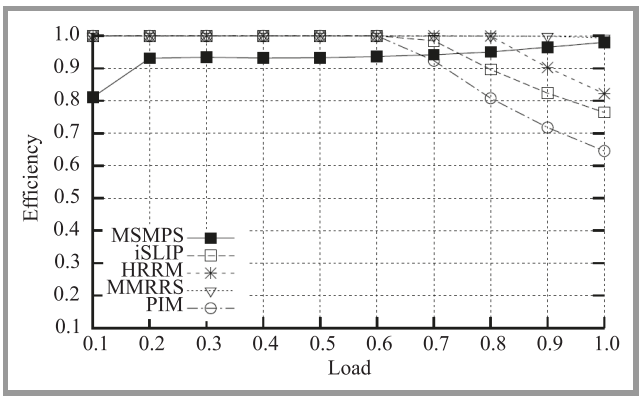


Fig. 5. The efficiency for Bernoulli arrivals with lin-diagonally distributed traffic in 16×16 switches.

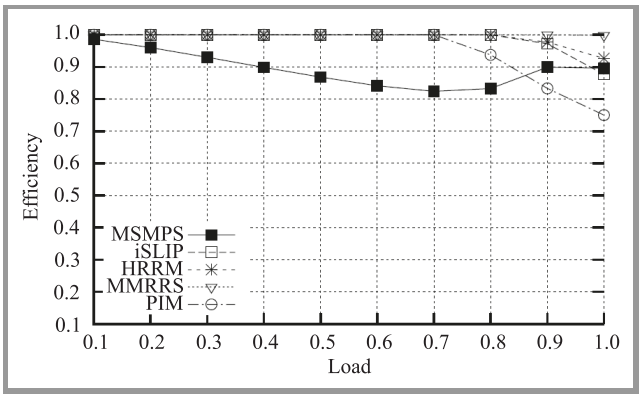


Fig. 6. The efficiency for Bernoulli arrivals with diagonally distributed traffic in 16×16 switches.

5. Simulation Results Analysis

In this section performance of the MSMPS algorithm will be compared with another algorithms for VOQ switches. Up today, several scheduling algorithms are presented in the literature [1]–[6]. It was compared and analyzed results for: iSLIP which was presented in [1], Maximal Matching with Round-Robin Selection (MMRRS) [2], [3], [4], Hierarchical Round-Robin Matching (HRRM) [5] and Parallel Iterative Matching (PIM) [6].

The efficiency is plotted in Figs. 4, 5 and 6. This parameter was calculated according to Eq. 2. Similarly as for MTD, results only for 16×16 switch size are presented. From Figs. 4 and 5 it can be observed that for low traffic load (between 10 – 20%) our algorithm achieve the worst results compared to other algorithms. Conducted simulations confirm, that MSMPS algorithm can not cope with low traffic load for different traffic models. The reason is that our algorithm focused very much on access alignment for all outputs, instead of avoiding of empty connections. Connections where no packets are to be send through the switch [7]. Above 20% load, efficiency of MSMPS algo-

rithm increases and reaches mean value about 0.95 with growing tendency. Different phenomena can be observed for other algorithms. All of them maintain efficiency on a high level about 1. But above 60% load, PIM and iSLIP rapidly decreases with nonuniformly and lin-diagonally distributed traffic. Only MMRRS maintain efficiency about 1 for both mentioned traffic distributions. It looks different with diagonally distributed traffic. Efficiency for MSMPS algorithm systematically decreases for over 40% load, efficiency is under 0.9. This type of distribution caused that packets are concentrated in two diagonals of the traffic matrix (Table 2). For this traffic model our algorithm achieve the worst results.

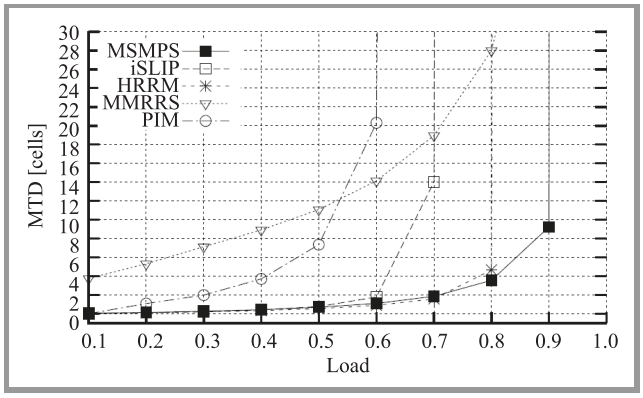


Fig. 7. The MTD for Bernoulli arrivals with nonuniformly distributed traffic in 16×16 switches.

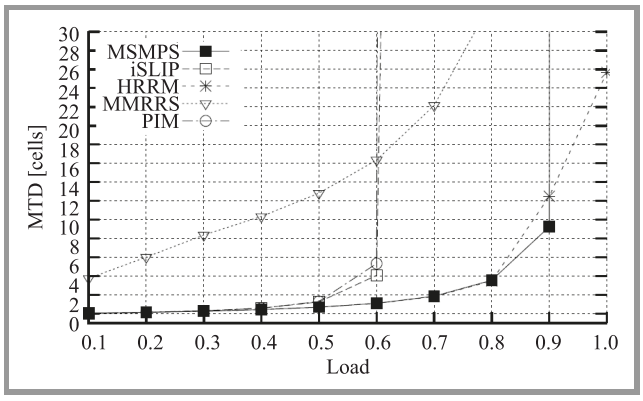


Fig. 8. The MTD for Bernoulli arrivals with lin-diagonally distributed traffic in 16×16 switches.

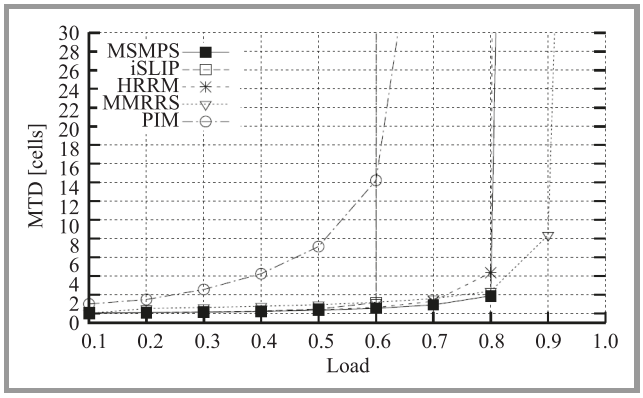


Fig. 9. The MTD for Bernoulli arrivals with diagonally distributed traffic in 16×16 switches

The MTD is a function of traffic load and is plotted in Figs. 7, 8 and 9. MTD is measured in time slots, where one slot is the basic of time unit in presented system. Computer simulations were performed for different switch sizes. Only the results for 16×16 switch size are shown. The authors assume that the input buffers are infinitely long, and have presented results for Bernoulli arrivals with different distribution traffic. From Fig. 7 it can be seen that for nonuniformly distributed traffic MSMPS algorithm achieve the best results (the lowest MTD) compared to other algorithms. Up to 75% load, only HRRM algorithm achieve similar results. The highest MTD, for this type of distribution, has reached MMRRS algorithm. For 10% load, MMRRS algorithm has already achieved 4 cells delay, when the rest of algorithms reached results close to 0. Very similar results are achieved by all algorithms with lin-diagonally distribution traffic – Fig. 8. MSMPS algorithm achieve almost the same results like for nonuniformly distribution. The same situation can be observed with MMRRS algorithm. Interesting situation occurred above 60% load, when MTD for PIM and iSLIP algorithm rapidly increase. It can be caused by arbiters synchronization problem. From the Fig. 9, with results for diagonal distribution traffic, it can be seen that MTD for our algorithm rapidly increased. This is due to our algorithm based on permanent connection patterns and for high load some outputs are blocked. According to this fact, to much empty connections are established. This effect can be eliminated by set up connections (between inputs and outputs) for more than one time slot. Acceptable results are reached by MMRRS algorithm which behave extremely well for diagonal distribution traffic.

6. Conclusions and Future Work

In this paper, performance results for MSMPS scheduling algorithm for VOQ switches under different traffic patterns were shown and described. Its performance confirms that MSMPS algorithm can be used in practice. This algorithm achieved high efficiency and in the same time low latency is provided. In the next studies, implementation of MSMPS algorithm in separate chips or in the switching fabric equipment will be discussed. Our algorithm works in simply way and there is no additional calculation needed. MSMPS algorithm can be also modified to support different traffic priorities and switch architectures.

Acknowledgements

The work described in this paper was financed from the research funding as research grant UMO-2011/01/B/ST7/03959.

References

[1] N. McKeown, "The iSLIP scheduling algorithm for input-queued switches", *IEEE/ACM Trans. Netw.*, vol. 7, pp. 188–200, 1999.

[2] A. Baranowska and W. Kabaciński, "The new packet scheduling algorithms for VOQ switches", in *Telecommunications and Networking – ICT 2004*, J. Neuman de Souza, P. Dini, and P. Lorenz, Eds. LNCS 3124, pp. 711–716. Springer, 2004.

[3] A. Baranowska and W. Kabaciński, "MMRS and MMRRS packet scheduling algorithms for VOQ switches", in *Proc. MMB & PGTS 2004 – 12th GI/ITG Conf. Measur. Eval. Comp. Commun. Sys. (MMB) & 3rd Polish-German Teletr. Symp. (PGTS)*, Dresden, Germany, 2004.

[4] A. Baranowska and W. Kabaciński, "Evaluation of MMRS and MMRRS packet scheduling algorithms for VOQ switches under bursty packet arrivals", in *Proc. Worksh. High Perfor. Switch. Rout. HPSR 2005*, Hong Kong, China, 2005, pp. 327–331.

[5] A. Baranowska and W. Kabaciński, "Hierarchical round-robin matching for virtual output queuing switches", in *Proc. Adv. Industr. Conf. Telecommun. AICT 2005*, Lisbon, Portugal, 2005, pp. 196–201.

[6] T. Anderson *et al.*, "High-speed switch scheduling for local-area networks", *ACM Trans. Comp. Sys.*, vol. 11, no. 4, pp. 319–352, 1993.

[7] G. Danilewicz and M. Dziuba, "The new MSMPS packet scheduling algorithm for VOQ switches", in *Proc. 8th IEEE, IET Int. Symp. Commun. Sys. Netw. Digit. Sig. Process. CSNDSP 2012*, Poznań, Poland, 2012.

[8] Y. Tamir and G. Frazier, "High performance multiqueue buffers for VLSI communication switches", in *Proc. 15th Ann. Int. Symp. Comp. architec. ISCA 1988*, Honolulu, Hawaii, USA, 1988, pp. 343–354.

[9] Myung-Ki Shin, Ki-Hyuk Nam, and Hyoung-Jun Kim, *Software-defined networking (SDN): A reference architecture and open APIs*, in *Proc. Int. Conf. ICT Converg. ICTC 2012*, Jeju, Korea, 2012.

[10] A. Baranowska and W. Kabaciński, "Hierarchiczny algorytm planowania przesyłania pakietów dla przełącznika z VOQ", in *Poznańskie Warsztaty Telekomunikacyjne PWT 2004*, Poznań, Poland, 2004 (in Polish).

[11] H. Jonathan Chao and B. Liu, *High Performance Switches and Routers*. New Jersey: Wiley, 2007, pp. 195–197.

[12] P. Giaccone, D. Shah, and S. Prabhakar, "An implementable parallel scheduler for input-queued switches", *IEEE Micro*, vol. 22, no. 1, pp. 19–25, 2002.

[13] K. Yoshigoe and K. J. Christensen, "An evolution to crossbar switches with virtual ouptut queuing and buffered cross points", *IEEE Network*, vol. 17, no. 5, pp. 48–56, 2003.

[14] D. Shah, P. Giaccone, and B. Prabhakar, "Efficent randomized algorithms for input-queued switch scheduling", *IEEE Micro*, vol. 22, no. 1, pp. 10–18, 2002.

[15] P. Giaccone, B. Prabhakar, and D. Shah, "Randomized scheduling algorithms for high-aggregate bandwidth switches", *IEEE J. Selec. Areas Commun.*, vol. 21, no. 4, pp. 546–559, 2003.

[16] Y. Jiang and M. Hamdi, "A fully desynchronized round-robin matching scheduler for a VOQ packet switch architecture", in Proc. IEEE Worksh. High Perform. Switch. Routing HPSR 2001, Dallas, TX, USA, 2001, pp. 407–411.

[17] A. Bianco, P. Giaccone, E. Leonardi, and F. Neri, "A framework for differential frame-based matching algorithms in input-queued switches", in *Proc. 23rd Ann. Joint Conf. IEEE Comp. Commun. Soc. IEEE INFOCOM 2004*, Hong Kong, China, 2004.

Grzegorz Danilewicz received the M.Sc. and Ph.D. degrees in Telecommunications from the Poznan University of Technology, Poland, in 1993 and 2001, respectively. Since 1993, he has been working in the Institute of Electronics, Poznan University of Technology. Currently he is a Professor of Poznan University of Technology and working as a Vice Dean of the Faculty of Electronics and Telecommunications. His scientific interests cover photonic broadband switching systems with special regard to the realization of multicast connections in such systems. He has published about 40 papers.

E-mail: Grzegorz.Danilewicz@et.put.poznan.pl
Chair of Communication and Computer Networks
Faculty of Electronics and Telecommunications
Poznan University of Technology
Polanka st 3
60-965 Poznan, Poland

Marcin Dziuba received the M.Sc. degree in Computer Science and Robotics from the Poznan University of Technology, Poland, in 2010. Since 2010, he is a Ph.D. student at Poznan University of Technology, Chair of Communication and Computer Networks Faculty of Electronics and Telecommunications.

E-mail: Marcin.Dziuba@put.poznan.pl
Chair of Communication and Computer Networks
Faculty of Electronics and Telecommunications
Poznan University of Technology
Polanka st 3
60-965 Poznan, Poland

Paper

Call and Connections Times in ASON/GMPLS Architecture

Sylwester Kaczmarek, Magdalena Młynarczuk, and Paweł Zieńko

Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

Abstract—It is assumed that demands of information society could be satisfied by architecture ASON/GMPLS comprehended as Automatically Switched Optical Network (ASON) with Generalized Multi-Protocol Label Switching (GMPLS) protocols. Introduction this solution must be preceded by performance evaluation to guarantee society expectations. Call and connections times are in ASON/GMPLS architecture important for real-time applications. Practical realization is expensive and simulations models are necessary to examine standardized propositions. This paper is devoted to the simulation results of ASON/GMPLS architecture control plane functions in OMNeT++ discrete event simulator. The authors make an effort to explore call/connection set-up times, connection release times in a single domain of ASON/GMPLS architecture.

Keywords—ASON, call control, call time, connection control, connection time, GMPLS, simulation model.

1. Introduction

Continuous information growth concerned with sophisticated applications generates the necessity of new telecommunication network architecture proposition based on optical solutions. The ITU-T Automatically Switched Optical Network (ASON) [1] concept with Generalized Multi-Protocol Label Switching (GMPLS) [2], [3] protocols has a chance to fulfill information society requirements. This solution is named as ASON/GMPLS.

The ASON/GMPLS control plane is composed of different components that provide specific functions (including routing and signaling). The main purpose of ASON/GMPLS control plane is to facilitate fast and efficient configuration of connections within a transport layer network to support both switched and soft permanent connections using GMPLS protocols like RSVP-TE [4], [5] for signaling and OSPF-TE for routing [6], [7]. The basic assumption of ASON control plane is a separation of call control from connection control. This separation makes it possible to control plane to be completely separate from transport plane.

The ASON architecture itself is only a concept. The advantages of this architecture are presented in [8]. The reference ASON control plane architecture describes the functional components including abstract interfaces and primitives. The recommendation presents interactions between call controller components, interactions among components during connection set-up and interactions among components during connection release. It also defines a functional component that transforms the abstract component interface into protocols. For a time being the standardization does not specify all protocols details needed to implementation.

Using GMPLS protocols or even mechanism of protocols gives the opportunity to ASON/GMPLS realization. Practical realizations are made only for simple network architecture [9]. For complex research simulations models are needed.

The aim of the paper is to present a series of simulation results to show the performance of ASON/GMPLS control plane functions to support switched connections and discuss the problem of call/connection set-up time and connection release time in a single domain. The work on simulation model has been preceded by practical realizations of ASON/GMPLS architecture in a laboratory testbed presented in [9], [10]. Performed tests validated correctness of all network elements operations including communication procedures and request processing. The same communication procedures are implemented in the simulation model with respect to ASON/GMPLS standardization and the latest trends in ITU-T NGN architecture [11]. The paper is organized as follows. General information about ASON/GMPLS architecture and basic control functions scenarios are depicted in Section 2. The ASON/GMPLS simulation model is presented in Section 3. Section 4 is devoted to presentation of performance tests results including call and connections times and loss probabilities. Conclusions and outlook to future are presented in Section 5.

2. Basic Control Plane Scenarios

2.1. ASON/GMPLS Control Plane Concept

This section is devoted to description of ASON recommendation and GMPLS protocols mechanisms proposed in ASON/GMPLS.

The idea of call and connection control is presented in [1]. The ASON recommendation separates the treatment of call and connection control. The call is a representation of the service offered to the user of a network, while connections are one of the means by which networks deliver required services. The ASON/GMPLS control plane is equipped with call and connection components.

The components concerned with call service are Calling/Called Party Controller (CCC) and Network Call Controller (NCC). The main roles of the CCC are call gener-

ation of call requests, acceptance or rejection of incoming call requests, generation of call termination requests. The CCC component is associated with the end of the call. The NCC component supports for calling and called party controllers and additionally supports calls at domain boundaries. Apart from call components ASON/GMPLS control plane is equipped with components involved in connection control like: Routing Controller (RC), Protocol Controller (PC), Connection Controller (CC), Link Resource Manager (LRM), Termination and Adaptation Performer (TAP). As recommended in [1] the Connection Controller is responsible for coordination among the Link Resource Manager, the Routing Controller and other Connection Controllers for the purpose of set-up, release and modification of connection. The Routing Controller provides routing functions using GMPLS routing protocol. The Link Resource Manager maintains the network topology. The role of the Protocol Controller is to map the operation of the components in the control plane into messages that are carried by GMPLS communication protocol between interfaces in the control plane. The Termination and Adaptation Performer holds the identifiers of resources that can be managed using the control plane interfaces. The group of components involved in connection control is considered in further sections as Control Element (CE).

Assumed that ASON/GMPLS control plane is equipped with two CCC (CCC_1 and CCC_2), NCC, three CE (CE_1, CE_2, CE_3) and transport plane is represented by three optical cross-connects (OXC_1, OXC_2 and OXC_3) the ASON/GMPLS architecture is presented in Fig. 1.

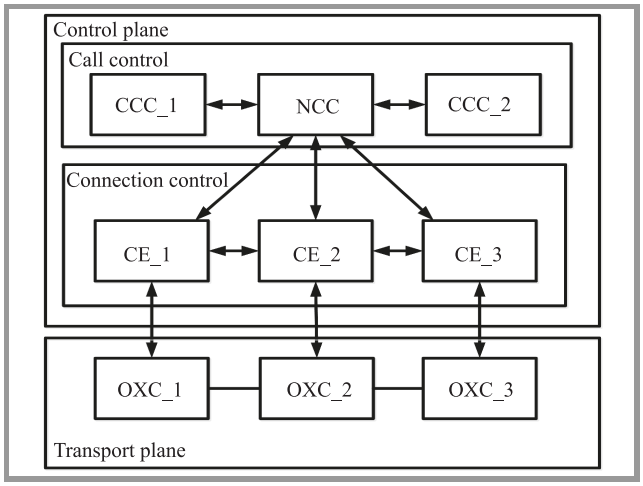


Fig. 1. The ASON/GMPLS network architecture.

The CE_i element is a representation of control elements for OXC_i (i = 1, 2, 3). The set-up and release scenario is performed by control plane components including call components and connection components.

2.2. Call/connection Set-up Scenario

In this section the authors want to present typical call/connections set-up scenario based on [4]. The same scenario is implemented in the simulation model. The scenario is graphical presented with definition of times necessary to calculate call set-up time, connection set-up time. The basic set-up scenario is presented in Fig. 2. The call set-up requests are sent by CCC_1. The Calling Party Controller CCC_1 sends a *call_request* to NCC. The NCC component sends *call_indication* to the Called Party Controller CCC_2. The CCC_2 component after call confirmation initiates connection set-up process sending *connection_request* to CE_1. Then communication between CE elements is performed by RSVP-TE signaling messages sending Path and Resv messages according to [4] until it reaches destination CE_3. After successful connection set-up in the transport plane CE_1 informs NCC sending *connection_confirmed*. Finally NCC sends *call_confirmed* to the Calling Party Call Controller.

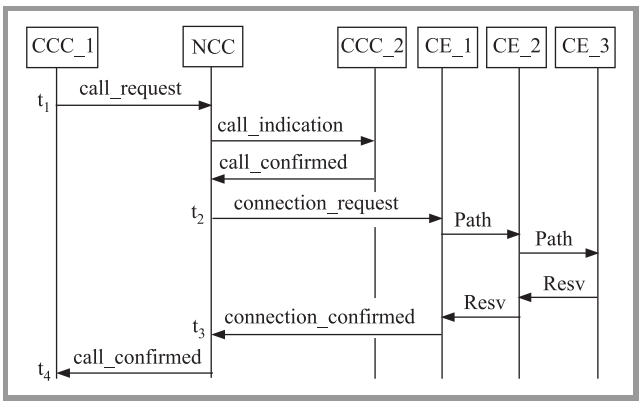


Fig. 2. The set-up scenario for ASON/GMPLS architecture.

Taking into consideration the call/connection set-up scenario presented in Fig. 2 value of call set-up time is defined as time from sending *call_request* (t_1) up to *call_confirmed* (t_4) while connection set-up time is defined as time from sending *connection_request* (t_2) to *connection_confirmed* (t_3).

2.3. Connection Release Sceanrio

The basic release scenario is performed as depicted in Fig. 3. The value of call connection release time is defined as time from sending *call_release* (t_5) up to *release_confirmed* (t_8) while connection release time is defined as time from sending *connection_release* (t_6) to *connection_release_confirmed* (t_7).

In the case of connection release ITU-T standardization group distinguishes release scenarios initiated by different call controllers [4]. The release request could be initiated by call controllers, e.g., Calling Party Call Controller, Called Party Call Controller, or any one of Network Call Controllers. The illustrations of various release requests are presented in [4]. In the release scenario presented in Fig. 3 the release request is initiated by Calling Party Controller CCC_1 by sending *call_release*. According to release scenarios in [4] the Path_release message repre-

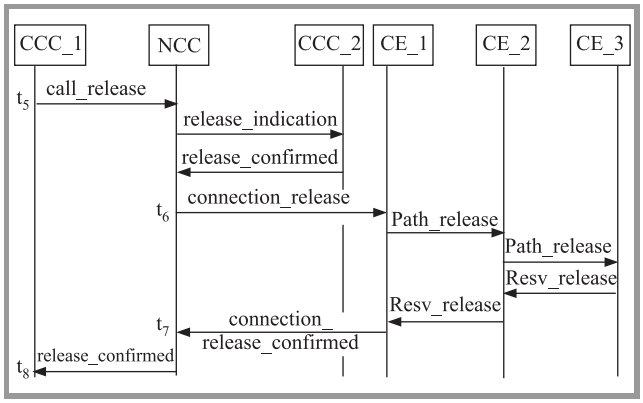


Fig. 3. The release scenario for ASON/GMPLS.

sents Path message. The Resv_release messages represents PathErr with Path_State_Removed flag.

3. Simulation Model

The ASON/GMPLS simulation model is created in OMNeT++ environment [12]. It consists six main functional blocks:

- control plane,
- transport plane,
- call generation,
- topology and resource information,
- initial configuration,
- measurements.

The control plane block consists of functional elements like: the Connection Controller (CC), the Routing Controller (RC), the Link Resource Manager (LRM), the Calling/Called Party Controller (CCC), the Network Call Controller (NCC). The transport plane block emulates Optical Cross-Connects (OXCs) operations. For each OXC blocking probability is assumed. Signaling is performed on separate wavelength. Resource allocation takes into consideration Routing and Wavelength Assignment problem (RWA) [13]. Routing functions are implemented in accordance with [1], [6].

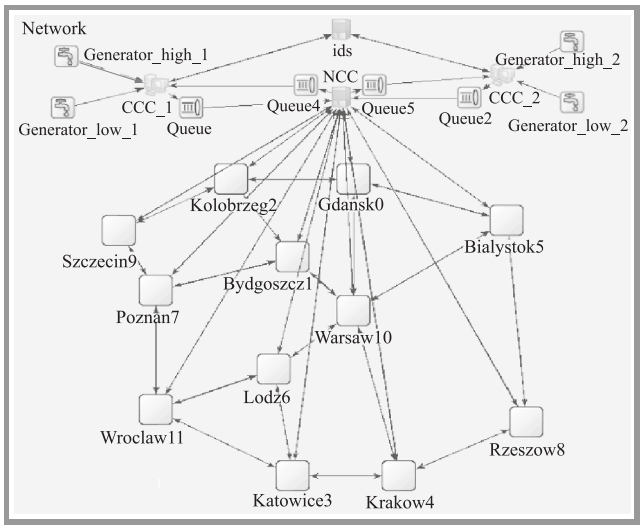


Fig. 4. The structure of simulated network Poland.

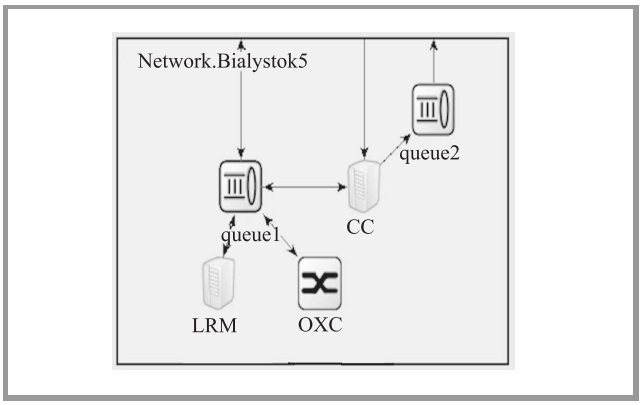


Fig. 5. The structure of node Bialystok.

Control plane functions in the simulation model are divided into call control functions and connection control functions. Call control functions concerned with call processing and connection control functions are responsible for set-up and release connections in transport plane. The call control plane is not aware of transport plane topology. The structure of the control plane model is presented in Fig. 4. The structure of node consists of control plane elements and OXC is presented in Fig. 5. In the simulation model the physical link is simulated by the single module which has a queue and a link as a representation of propagation delay. The model of ASON control plane functions is based on the following assumptions:

- call control functions are represented by elements: two Calling/Called Party Controllers (CCC_1, CCC_2), Network Call Controller (NCC), ids,
- connection control function are performed by Control Elements (CE),
- the Control Element consists of the Connection Controller (CC), the Routing Controller (RC), the Link Resource Manager (LRM), the Termination and Adaptation Performer (TAP) (see Fig. 5),
- the number of the CE is equal to the number of nodes,
- mapping of the CE to transport plane (represented as emulated OXC) is one-to-one.

The transport plane has separate resources for high and low priority requests (20% recourses are for high priority requests).
As is depicted in Fig. 4 generation block is represented by components: generator_high_1, generator_low_1, generator_high_2 and generator_low_2. The generator_high_1 and generator_low_1 generate requests with high and low

priority respectively with defined distribution and send towards CCC_1, while generator_high_2 and generator_low_2 send high and low priority requests towards CCC_2. CCC_1 and CCC_2 send received requests to ids, which assigns unique call identifier (Call_ID) to generated call requests. Afterwards Call_ID is located in the call request and send to NCC. The process of call processing is presented in Fig. 6.

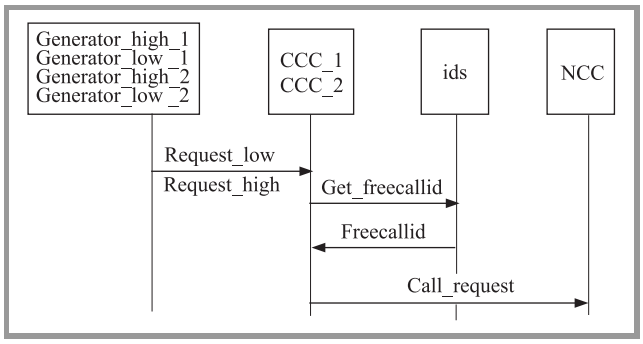


Fig. 6. The call control in ASON/GMPLS control plane.

The topology and resource information block is in charge of storage control plane topology, transport plane topology, domain allocation (links, distance between nodes). All parameters including network topology are configurable during initial configuration. To make realistic network conditions the topology is based on [14].
Due to initial configuration block we are able to set initial values for:

- call generator (distributions of call requests for low/high priority, distribution of connection release for low/high priority),
- traffic matrix (coefficient matrix),
- measurement and run the simulation (i.e., simulation time limit, warm-up period, event log module recording, seed, call time distribution),
- control plane (assignment sources addresses to nodes, assignment unique id number to each node),
- transport plane (blocking probability of OXC).

The measurements are performed in NCC component and ids. The ids is responsible for call set-up measurement, call release measurement. The NCC component is responsible for call release measurement, connection release measurement. Necessary times depicted in Fig. 2 and Fig. 3 (t_1–t_8) to calculate call/connection set-up/release times are stored in .vec files. The simulation execution consists of warm-up period and n measurements periods. Comprehensive statistical analysis is performed offline based on .vec files.
Taking into consideration the presented model and call/connection set-up and release scenarios presented in Section 2, the simulation model makes it possible to evaluate connection loss probability. The connection loss probability could be caused by lack of optical resources or blocking probability of OXC.

4. Control Plane Performance Results and Discussion

The performance evaluation of control plane functions in ASON/GMPLS is presented by performance results. The simulation scenario includes call and connection set-up times and connection release times which were estimated using t-Student distribution with confidence level equal 0.95. The confidence intervals are low and they aren't marked. The simulation was executed for a single domain Poland network using following assumptions:

- total simulation time: 3600 s,
- warm-up period: 200 s,
- 15 measurements intervals,
- exponential distribution of call request,
- exponential distribution of connection release requests,
- 20% of all generated requests are high priority,
- mean connection duration time (ConnD): 15 minutes, 30 minutes,
- blocking probability of OXC: 0.001,
- signaling link capacity 10 Mbit/s,
- wavelength capacity: 1 Gbit/s
- capacity of single connection requests: 5 Mbit/s, 10 Mbit/s, 15 Mbit/s,
- the number of wavelengths per fiber: 40.

In the simulation Poland topology was assumed with shortest path algorithm (Dijkstra) for routing. The section presents exemplary results based on OMNeT++ simulations.

4.1. Call/connection Set-up Results

Results presented in Figs. 7–8 (for connection duration 30 minutes) indicate that mean values of call set-up time and connection set-up time significantly depend on request intensity assumed as the sum of call/connection set-up requests and connection release requests. Mean values of call/connection set-up time are presented for: all generated call/connection set-up requests, call/connection set-up requests successful ended and for call/connection set-up requests unsuccessful ended. Unsuccessful connection set-up request results from lack of free resources in emulated transport plane or blocking probability of OXC.
All call set-up times and connection set-up times have tendency to low in the range from 300 requests per second to 9700 request per second with raid growth of connection loss probability. For intensities smaller than 700 re-

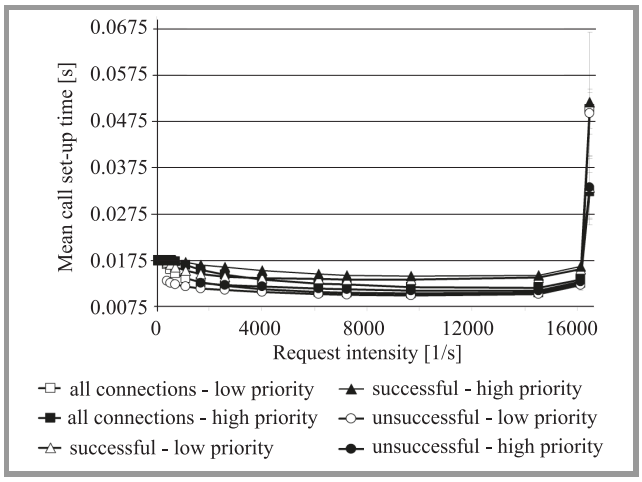


Fig. 7. Mean call set-up time.

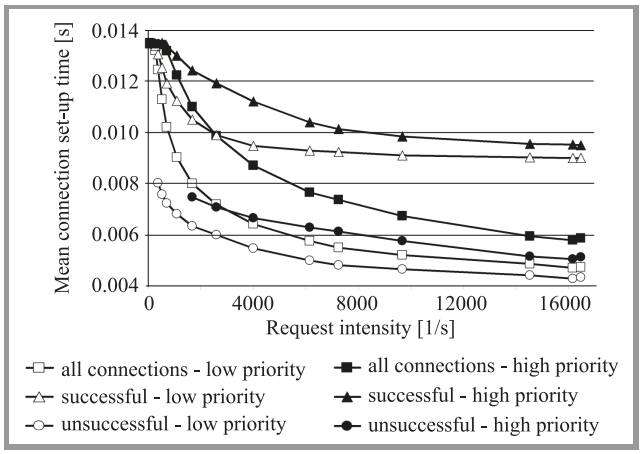


Fig. 8. Mean connection set-up time.

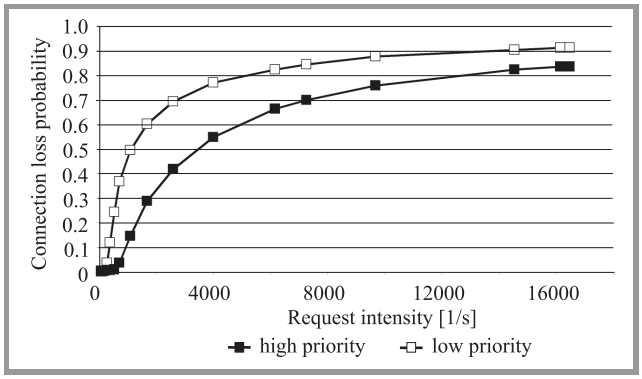


Fig. 9. Connection loss probability.

quests per second loss probability for high priority requests is smaller than 0.04. Detailed analysis of the OMNeT++ event log verified that for intensity greater than 1000 requests per second successful connections were established to the nearest nodes. This effects smaller call/connection set-up time. For intensity greater than 9700 requests per second the bandwidth of signaling link was too small to service call traffic. The greater intensity is, the longer the waiting times for RSVP massages send between control plane components. Additionally, results presented in Fig. 9 indicate that the connection loss probability for intensity 1098 requests per second is equal 0.15 for high priority connection requests and 0.5 for low priority connection requests. For high priority connection requests and intensities 290, 392, 700 requests per second loss probabilities equal 0.04, 0.1 and 0.3 respectively. For low priority connection request loss probabilities 0.04, 0.1, 0.3 are for intensities 714, 1000, 1690 requests per second respectively. Figure 10 presents loss probabilities for connection

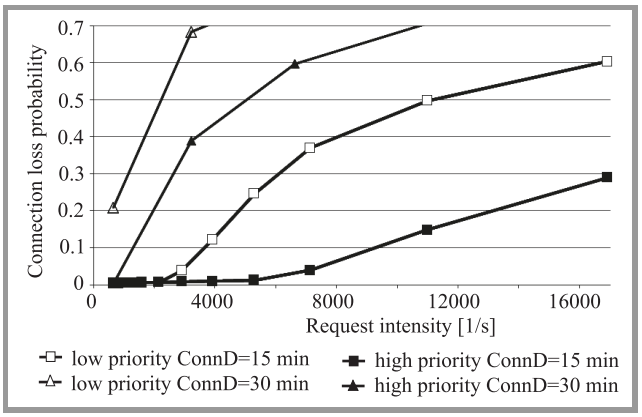


Fig. 10. Connection loss probability.

duration time (ConnD) equals 15 minutes and connection duration time (ConnD) equals 30 minutes. The call set-up time and connection set-up time decrease is caused by reducing amount of connections on long distance and high loss probability equal more than 0.3. Additionally, the path computation algorithm takes into consideration distance and RWA requirements. Due to this for greater intensity more often shortest connections are established. In the simulation Poland topology was assumed. Taking into consideration topology and routing assumption we noticed that the majority of connections for intensities greater than 700 requests was established on shorter distance. There are more shorter connections than farther connection in Poland topology which consists of 12 nodes. The greater intensity is, the greater the number of connections to near nodes is. Figures 11–13 present number of established connections in Poland from node 0 (Gdansk) for intensity equals 65 requests per second, 714 requests per second and 9671 requests per second respectively. Figures 14–16 present number of established connections in Poland from node 3 (Katowice) for intensity equals 65 requests per second, 714 requests per second and 9671 requests per second respectively.

The greater intensity is, the greater shortage of free resources probability is. The number of established connection changes for different length of connections. The majority of connections are established with the length of connections equals two nodes.

The number of established connections from Gdansk in the length of nodes for intensity equals 65 requests per second, 714 requests per second and 9671 requests per second respectively present Figs. 17–19. The number of established connections from Katowice in the length of nodes for

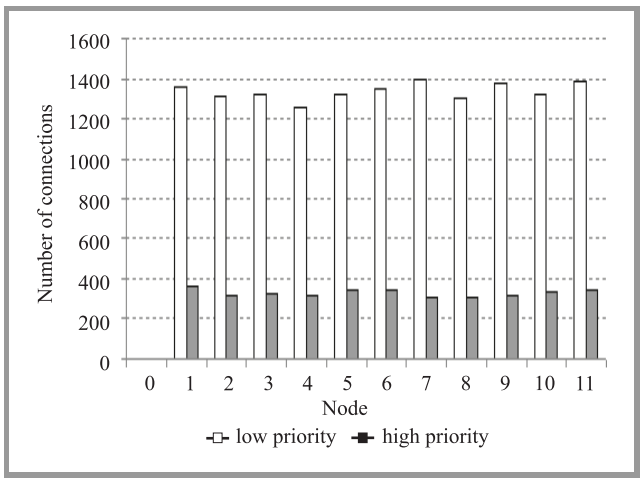


Fig. 11. The number of connections from Gdansk node for intensity 65 requests per second.

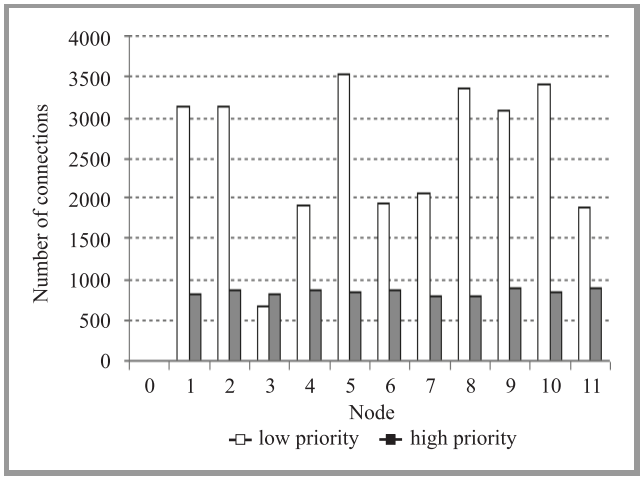


Fig. 12. The number of connections from Gdansk node for intensity 714 requests per second.

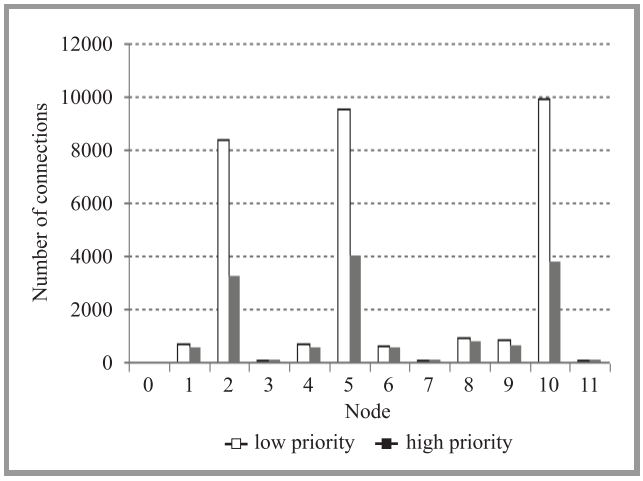


Fig. 13. The number of connections from Gdansk node for intensity 9671 requests per second.

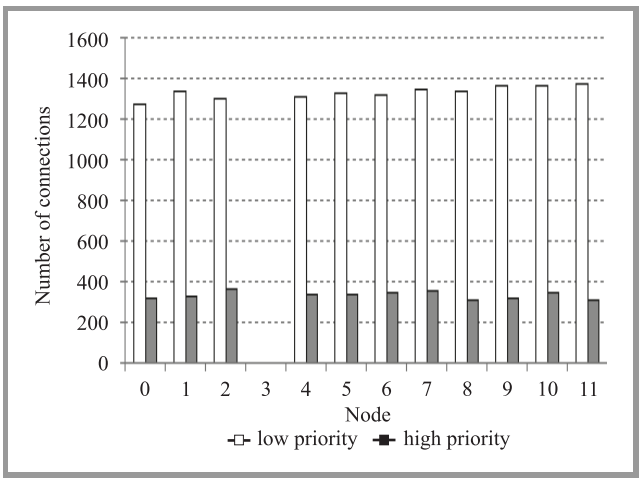


Fig. 14. The number of connections from Katowice node for intensity 65 requests per second.

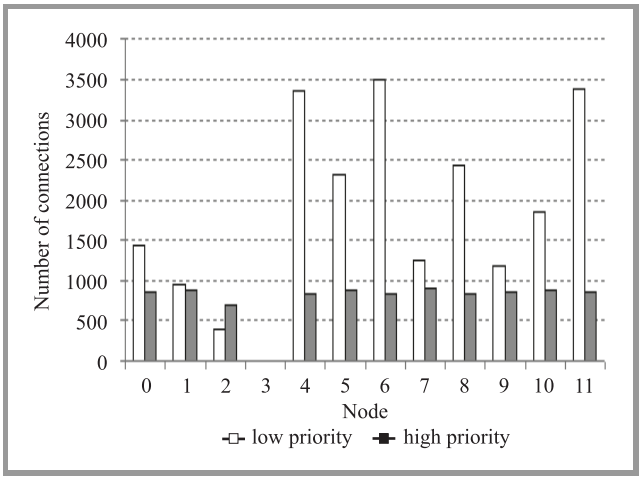


Fig. 15. The number of connections from Katowice node for intensity 714 requests per second.

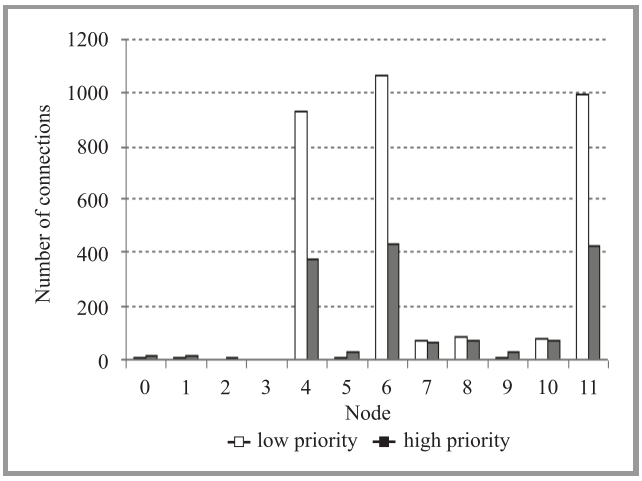


Fig. 16. The number of connections from Katowice node for intensity 9671 requests per second.

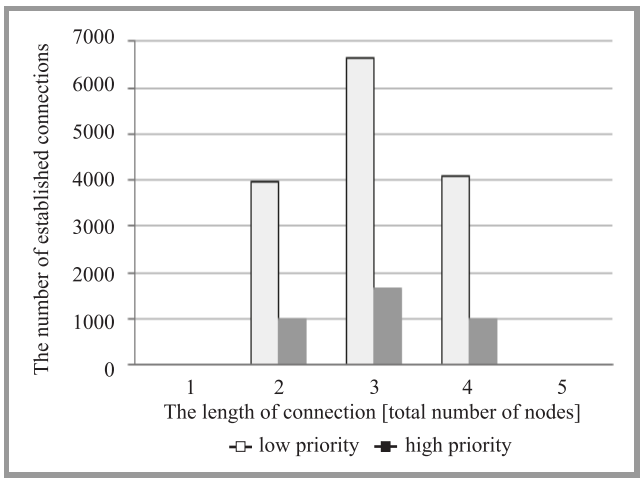


Fig. 17. The number of established connections in length of connection from Gdansk for intensity 65 requests per second.

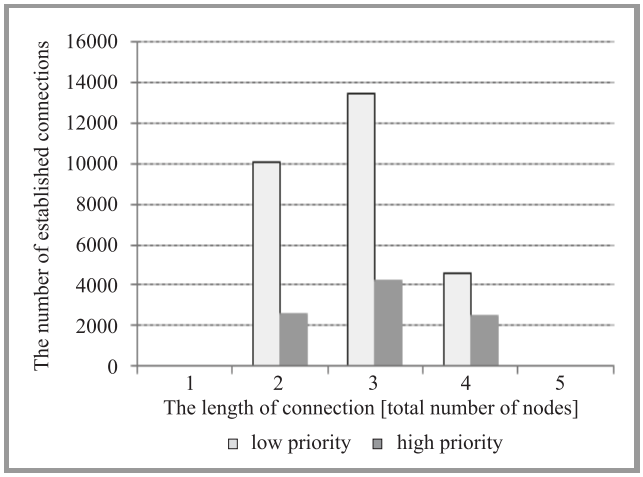


Fig. 18. The number of established connections in length of connection from Gdansk for intensity 714 requests per second requests per second.

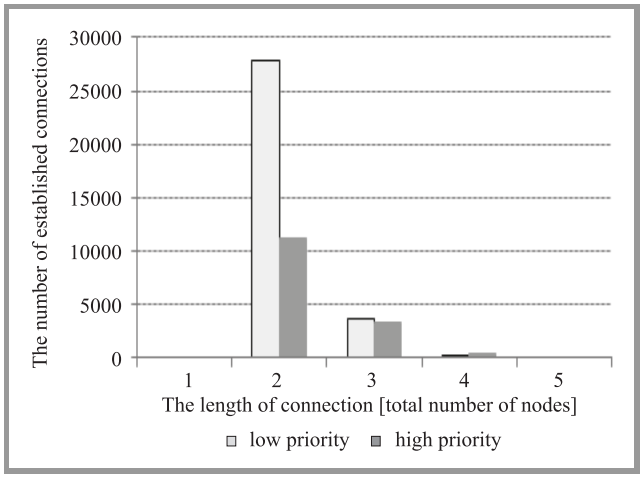


Fig. 19. The number of established connections in length of connection from Gdansk for intensity 9671 requests per second requests per second.

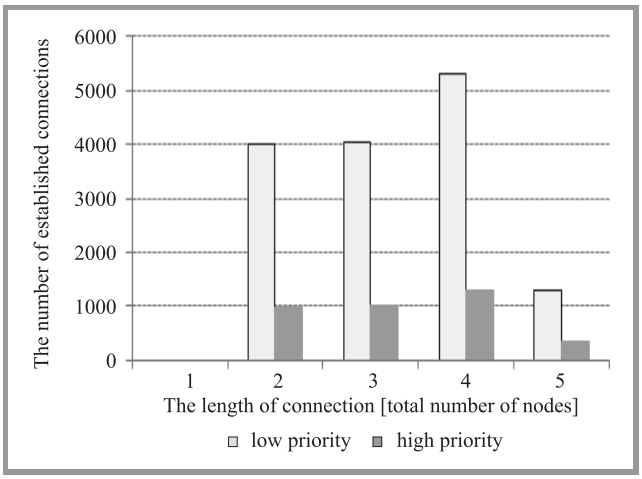


Fig. 20. The number of established connections in length of connection from Katowice node for intensity 64 requests per second.

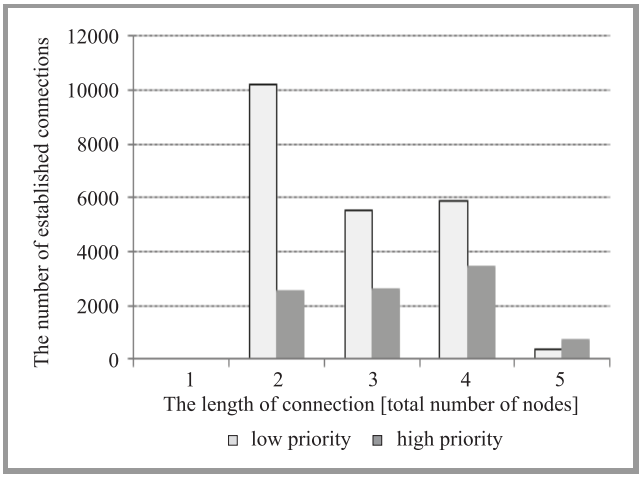


Fig. 21. The number of established connections in length of connection from Katowice node for intensity 714 requests per second.

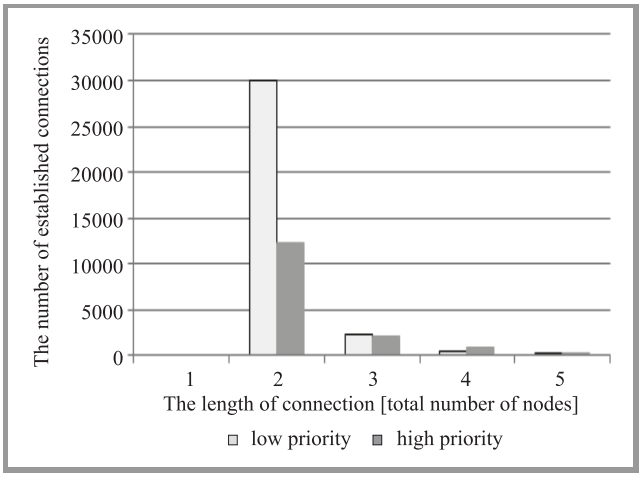


Fig. 22. The number of established connections in length of connection from Katowice node for intensity 9671 requests per second.

intensity equals 65 requests per second, 714 requests per second and 9671 requests per second respectively present Figs. 20–22. In all figures first node is a source node. As depicted in Figs. 17–22 the greater intensity requests the number of established connections in the length of three, four and five nodes decrease in comparison with connections established in the length of connection equals two nodes.

4.2. Connection Release Results

Connection release time results are presented in Fig. 23 and Fig. 24. Call connection release time includes time of connection release and call release in call control of control plane. The connection release time depends on request intensity. Similar to mean values of call/connection set-up time, mean values of connection release time have tendency to low. Results presented in Fig. 23 indicate that connection release time is smaller than connection set-up time. It is associated with realizing connection scenario. The authors assumed that the control elements only sends resource release requests and do not wait for confirmation from emulated optical resources.

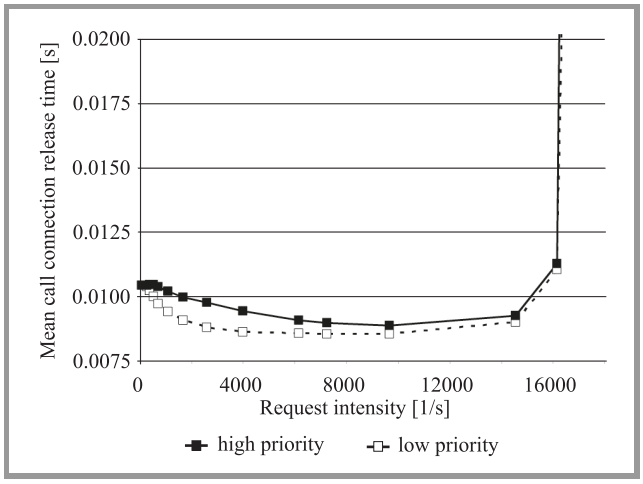


Fig. 23. Mean call connection release time.

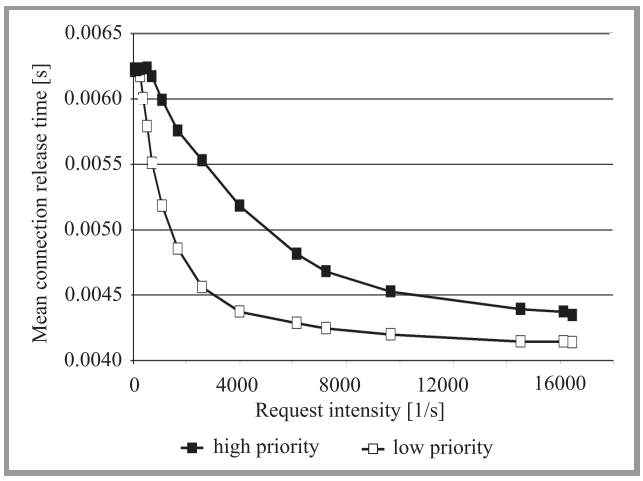


Fig. 24. Mean connection release time.

Presented in Fig. 24 results of mean connection release times indicate in the range from 300 requests per second to 7500 requests per second to decrease. In Fig. 23 for intensity greater than 7500 requests per second time of call service increases. The log analysis shows that assumed value of signaling link capacity in call control is too small and leads to rapid grow of call connection release time.

5. Conclusions

In the paper a simulation model of ASON/GMPLS domain architecture is presented. The model was implemented in OMNeT++ simulator, which was proved to be efficient to ASON/GMPLS application. The model allows to determine mean values of call/connection set-up time, connection release time. The results of performance evaluation of ASON/GMPLS architecture are demonstrated. Call and connections times are presented. According to simulation results, implementing ASON/GMPLS architecture leads to achieving a very high availability real time applications in Next Generation Network.

The simulation results show that the loss probability significantly increases for low priority connection requests while call/connection set-up time is shorter about 2 ms than for high priority call/connection requests. Moreover, presented results indicate the importance of routing function implementation and RWA type resource allocation mechanism. Verification by detailed analysis shows that the most chosen path was the shortest. Due to this call/connection set-up time decreases.

The model allows to simulate the impact of various network topology on processing time. Using the model many set of input variables presented in Section 3 can be changed. The model is a modular design based on compound modules and can be easily expanded. Due to space limitation, only call/connection time results for structure of Poland are presented. The structure was investigated as a single domain. The authors are in agreement that further efforts should be made into researching the relationship between time consuming concerned with multi-domain topology and call/connection set-up time and connection release time.

Acknowledgements

This research work was partially supported by the system project "InnoDoktorant – Scholarships for PhD students, Vth edition", which is co-financed by the European Union in the frame of the European Social Fund.

References

[1] "Architecture for the automatically switched optical network", ITU-T Rec. G.8080/Y.1304, Feb. 2012.

[2] E. Mannie, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", IETF RFC 3945, Oct. 2004.

[3] A. Farrel and I. Bryskin, *GMPLS: Architecture and Applications*. Morgan Kaufmann, 2006.

[4] "Distributed Call and Connection Management: Signalling mechanism using GMPLS RSVP-TE", ITU-T Rec. G.7713.2/Y.1704.2, Mar. 2003.

[5] OIF Guideline Document: Signaling Protocol Interworking of ASON/GMPLS Network Domains, Jun. 2008.

[6] "ASON routing architecture and requirements for link state protocol", ITU-T Rec. G.7715.1/Y.1706.1, Feb. 2004.

[7] "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", K. Kompella and Y. Rekhter, Eds., IETF RFC 4203, Oct. 2005.

[8] A. Jajszczyk, "Automatically switched optical networks: Benefits and Requirements", *IEEE Opt. Commun.*, pp. 510–515, Feb. 2005.

[9] S. Kaczmarek, M. Narloch, M. Młynarczuk, and M. Sac, "The Realization of NGN Architecture for ASON/GMPLS Network", *J. Telecommun. Inform. Technol. (JTIT)*, no. 3, pp. 47–56, 2011.

[10] S. Kaczmarek, M. Narloch, M. Młynarczuk, and M. Sac, "Evaluation of ASON/GMPLS Connection Control Servers Performance", in *Information Systems Architecture and Technology, Service Oriented Network Systems*. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, 2011, pp. 267–278.

[11] "Functional Requirements and architecture for next generation networks", ITU-T Rec. Y.2012, Apr. 2010.

[12] "OMNeT++ Network Simulation Framework" [Online]. Available: www.omnetpp.org

[13] H. Zang, J. P. Jue, and B. Mukherjee, "A Review of Routing and Wavelength Assignment Approaches for Wavelength Routed Optical WDM Networks", *Opt. Netw. Mag.*, pp. 47–60, Jan. 2000.

[14] Network library, Zusse Institut Berlin [Online]. Available: http://sndlib.zib.de/

Magdalena Młynarczuk received her M.Sc. degree in Telecommunication Systems and Networks from Gdańsk University of Technology in 2004. Since 2008 till March 2011 she has been an assistant at Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics. Now she works as lecturer at GUT. Her research interests include control of optical networks, transmission and switching technology and network design. Her doctoral thesis is entitled: "QoS Routing in multi-domain optical network with hierarchical structure control plane".

E-mail: magdam@eti.pg.gda.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Paweł Zieńko received his M.Sc. degree in Teleinformation Networks and Systems from Gdańsk University of Technology in 2012. His Master's thesis includes realization of routing simulation model in optical networks ASON/GMPLS. Since 2012 he works in Intel Technology Poland as Graphics Software Engineer.

E-mail: pawel.zienko@interia.pl
Department of Teleinformation Networks
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Sylwester Kaczmarek – for biography, see this issue, p. 17.

Paper

Single Hysteresis Model for Limited-availability Group with BPP Traffic

Maciej Sobieraj, Maciej Stasiak, Joanna Weissenberg, and Piotr Zwierzykowski

Chair of Communication and Computer Networks, Poznan University of Technology, Poznan, Poland

Abstract—This paper presents a single hysteresis model for limited-availability group that are offered Erlang, Engset and Pascal traffic streams. The occupancy distribution in the system is approximated by a weighted sum of occupancy distributions in multi-threshold systems. Distribution weights are obtained on the basis of a specially constructed Markovian switching process. The results of the calculations of radio interfaces in which the single hysteresis mechanism has been implemented are compared with the results of the simulation experiments. The study demonstrates high accuracy of the proposed model.

Keywords—hysteresis mechanism, limited-availability group, multiservice BPP traffic, threshold models, WCDMA radio interface.

1. Introduction

Many network systems make use of traffic management mechanisms that aim at an increase in the traffic capacity of the network. Such mechanisms are to be primarily found in access networks that are characterized by low capacity of resources. A good example of the above is provided by, for example, 2G, 3G, and 4G radio access networks in which radio interface capacities are very limited. Traffic management mechanisms in these systems usually employ such mechanisms as [1]: resource reservation, partial limitation of resources, priorities, traffic overflow, non-threshold compression and threshold compression. The operation of the reservation mechanism is based on making the capabilities of the reservation of resources for pre-selected call classes dependent on the load level of the system [2], [3]. Many operators take advantage of the mechanism of partial limitation of resources that limits the number of serviced calls of appropriate traffic classes to a predefined value [4]. Priorities are designated to particular classes of calls. Prioritized calls can – in the case of the lack of free resources – effect a termination of service for calls with lower priority [1]. The overflow mechanism is one of the oldest mechanisms used in telecommunications [5], [6]. When the mechanism applies, calls that cannot be serviced in a given system due to its current occupancy level are redirected to other systems that still have free resources. Non-threshold compression is, in turn, based on a possibility of making the throughput of serviced calls of selected classes decreased in order to obtain free resources for servicing new calls [1]. This mechanism forms a basis of the High Speed Packet Access technology (HSPA) in Universal Mobile Telecommunications System (UMTS) networks [7].

In the threshold compression mechanism, the bit rate allocated to a new call depends on the load of the system. The mechanism is used to service elastic and adaptive traffic [8]. The first model of a threshold system, the so-called Single Threshold Model (STM), was devised in [9] and concerned a system that was called Single Threshold System (STS). Works [8], [10] considers systems with a number of independent thresholds, the so-called Multi Threshold Systems (MTS) and the corresponding analytical models, the so-called Multi Threshold Models (MTM). Paper [11], describe a variant of the single-threshold system – Single Hysteresis System (SHS) and the corresponding analytical model – Single Hysteresis Model (SHM). In SHS, two thresholds, in place of one, are introduced. The operation of each of the thresholds is dependent on the direction of changes in the load in the system. The introduction of hysteresis is followed by a more stable operation of the system, which can be proved by a decreased number of transitions between areas with high and low load.

The present paper for the first time proposes a SHM for limited-availability group [12] with traffic streams of BPP type. In the paper [13] a SHM was presented only for full-availability group. The very name – BPP [4], [10] stems from the names of the types of call streams, (Bernoulli, Poisson and Pascal) that comprise Engset, Erlang and Pascal traffic, respectively.

The paper is structured as follows. Section 2 presents analytical models of the BPP traffic. Section 3 discusses STM and structure of limited-availability group, whereas Section 4 describes SHS for limited-availability group with BPP traffic. In Section 5, the results of the analytical calculations are compared with the results of simulation experiments of two different structures of systems. Section 6 presents the conclusions resulting from the study.

2. Multi-Service BPP Traffic

Multi-service traffic is a mixture of different traffic streams that are differentiated from one another by the number of allocation units, the so-called Basic Bandwidth Units (BBU) [3] that are necessary to set up a connec-

tion in the system. Traffic streams can be generated by an infinite (Erlang) or finite (Engset and Pascal) number of traffic sources. The intensity of Erlang traffic of class i, generated in the occupancy state of the system n BBUs, can be expressed by the following formula:

$$A_i(n) = A_i = \lambda_i/\mu_i = \text{const}, \tag{1}$$

where: λ_i – the average call intensity of calls of class i, μ_i – the average intensity of service of calls of class i.

The intensity of Engset traffic of class j and that of Pascal traffic of class k depend on the state of the system and is defined in the following way [10]:

$$A_j(n) = [N_j - y_j(n)]\alpha_j = [N_j - y_j(n)]\frac{\gamma_j}{\mu_j}, \tag{2}$$

$$A_k(n) = [N_k + y_k(n)]\alpha_k = [N_k + y_k(n)]\frac{\gamma_k}{\mu_k}, \tag{3}$$

where: N_c – the number of traffic sources of class c[1], $y_c(n)$ – the number of calls of class c, serviced in state n, α_c – traffic intensity from one free source of class c:

$$\alpha_c = \gamma_c/\mu_c, \tag{4}$$

where γ_c is the call intensity of calls from one free source of class c.

3. Single Threshold System

3.1. Single Threshold System – Working Idea

Assume that in the system for pre-defined call classes one threshold Q, has been introduced in [9]. Figure 1 shows the operation of the system with single threshold with the

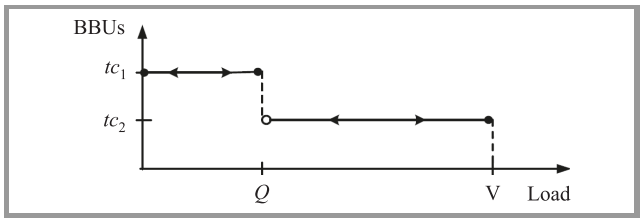


Fig. 1. Single Threshold System – working idea.

example of calls of one class c. If the load of the system is lower than the adopted value Q $(0 \le n \le Q)$, the Call Admission Control (CAC) function allows for a new call of class c with the maximum number of BBUs, equal to $t_{c,1}$, to be serviced. Following an increase in the load in the system and after exceeding the threshold Q, within the area of maximum load $(Q < n \le V)$, the CAC function admits for service a call of class c with the minimum number of BBUs, equal to $t_{c,2}$.

[1] In the adopted notation, the indexes i, j, and k are used to denote classes of Erlang, Engset and Pascal traffic, respectively. The index c is in turn used to consider traffic related to any traffic class.

3.2. Model of Limited-availability Group

The limited-availability group is the model of communication system that consists of k identical separated links [12]. Each link has the capacity equal to v BBUs. Thus, the total capacity of the system V is equal to $V = kv$ BBUs. The system services a call – only when this call can be entirely carried by the resources of an arbitrary single link. Thus, limited-availability group is an example of the system with state-dependent service process. Figure 2 shows the model of the limited-availability group [10].

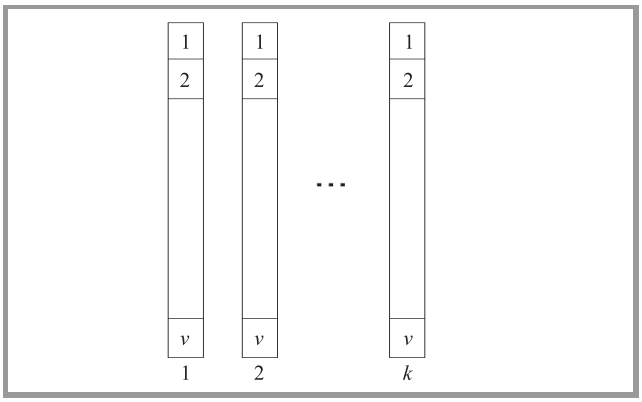


Fig. 2. Model of the limited-availability Group.

3.3. Adaptive and Elastic Traffic in SHS

In systems servicing the so-called adaptive traffic [8] the change applies only to the number of BBUs necessary to set up a connection of a given class. The assumption is that traffic of this type requires sending of all data, while a decrease in the number of allocated BBUs will be followed by a deterioration of the Quality of Service (QoS) parameters. A good example of the above is the voice service "full-rate" and "half-rate" in the GSM network. Elastic traffic [8] requires all data to be transferred, thus a decrease in the allocated number of BBUs will be followed by an increase in the service time, i.e., the parameter $1/\mu_c$. HSDPA traffic in the UMTS network is an example of the above. Thus, the service of elastic traffic causes the value of offered traffic in particular load areas to be changed. Therefore, in the case of the service of elastic traffic, Formulas (1), (2) and (3) can be rewritten in the following way:

$$A_{i,s}(n) = A_i = \lambda_i/\mu_{i,s} = \text{const}, \tag{5}$$

$$A_{j,s}(n) = [N_j - y_{j,s}(n)]\alpha_{j,s} = [N_j - y_{j,s}(n)]\frac{\gamma_j}{\mu_{j,s}}, \tag{6}$$

$$A_{k,s}(n) = [N_k + y_{k,s}(n)]\alpha_{k,s} = [N_k + y_{k,s}(n)]\frac{\gamma_k}{\mu_{k,s}}, \tag{7}$$

where s indicates load area. We can distinguish two load areas in STM: $s=1$ for $n \in \langle 0;Q\rangle$ and $s=2$ for $n \in (Q;V\rangle$.

3.4. Occupancy Distribution in Limited-availability Group with STM and BPP Traffic

The occupancy distribution in the limited-availability group with single threshold mechanism and BPP traffic can be determined on the basis of the model worked out in [9] for STM with Erlang traffic and in [10] for MTM with BPP traffic. According to this model, the occupancy distribution in considered model can be rewritten as follows:

$$n[P_n]_Q^{(V)} = \sum_{s=1}^{2}\Bigg\{ \sum_{i\in M_1} A_{i,s}(n-t_{i,s})t_{i,s}\sigma_{i,s,Total}(n-t_{i,s})\left[P_{n-t_{i,s}}\right]_Q^{(V)} + \sum_{j\in M_2} A_{j,s}(n-t_{j,s})t_{j,s}\sigma_{j,s,Total}(n-t_{j,s})\left[P_{n-t_{j,s}}\right]_Q^{(V)} + \sum_{k\in M_3} A_{k,s}(n-t_{k,s})t_{k,s}\sigma_{k,s,Total}(n-t_{k,s})\left[P_{n-t_{k,s}}\right]_Q^{(V)}\Bigg\}, \quad (8)$$

where: $n[P_n]_Q^{(V)}$ – probability of n BBUs being busy in STS with capacity V BBUs, M_x – a set of call classes of Erlang $(x=1)$, Engset $(x=2)$ and Pascal calls $(x=3)$, respectively, $t_{c,s}$ – the number of BBUs required to set up a connection of class c in load area s, $A_{c,s}(n)$ – the average traffic intensity of class c offered to the system in the occupancy state n that belongs to the load area s. For adaptive traffic, this parameter is determined by Eqs. (1)–(3); for elastic traffic, by Eqs. (5)–(7). $\sigma_{c,s,Total}(n)$ – conditional transition coefficient that determines which part of the input call stream in the threshold area s will be transfered between the states n and $n+t_{c,s}$:

$$\sigma_{c,s,Total}(n) = \sigma_{c,s,LAG}(n)\cdot\sigma_{c,s}(n), \quad (9)$$

where $\sigma_{c,s,LAG}(n)$ is a conditional transition probability which determines the part of class c arrival stream which is transfered between states n and $n+t_{c,s}$, $\sigma_{c,s}(n)$ – conditional transition probability that in Eq. (8) is a switching coefficient between appropriate load areas.

The conditional transition probability can be determined with the help of following equation [15]:

$$\sigma_{c,s,LAG}(n) = 1 - \frac{F(V-n,k,t_{c,s}-1,0)}{F(V-n,k,v,0)}, \quad (10)$$

where $F(x,k,v,t)$ is the number of possible allocations of x free BBUs in k links, calculated with the assumption that the capacity of each link is equal to v BBUs and each link has at least t free BBUs:

$$F(x,k,v,t) = \sum_{i=0}^{\lfloor\frac{x-kt}{v-t+1}\rfloor}(-1)^i\binom{k}{i}\binom{x-k(t-1)-1-i(v-t+1)}{k-1}. \quad (11)$$

The threshold mechanism introduces dependence between the traffic stream and the current state of the system. This dependence can be determined as follows. For traffic classes that do not undergo the threshold mechanism, this parameter always takes on the value equal to one:

$$\sigma_{c,s}(n) = 1. \quad (12)$$

For all traffic classes that undergo the threshold mechanism, the value of the parameter $\sigma_{c,s}(n)$ is defined in the following way:

$$\sigma_{c,1}(n) = \begin{cases} 1 & \text{for } n \le Q, \\ 0 & \text{for } n > Q, \end{cases} \quad \sigma_{c,2}(n) = \begin{cases} 0 & \text{for } n \le Q, \\ 1 & \text{for } n > Q. \end{cases} \quad (13)$$

To determine the occupancy distribution in STM according to Eq. (8) it is necessary to determine the values of intensities of offered Engset $A_{j,s}(n)$ and Pascal $A_{k,s}(n)$ traffic streams in individual states of the service process. These values can be determined on the basis of the parameter $y_{c,s}(n)$, i.e., the number of calls of a given class serviced in state n that belong to the load area s. This parameter can be approximated by the average number of calls of a given class that are serviced in the occupancy state n [1], [10]:

$$y_{c,1}(n) = \frac{A_{c,1}(n-t_{c,1})\sigma_{c,1,Total}(n-t_{c,1})[P_{n-t_{c,1}}]_Q^{(V)}}{[P_n]_Q^{(V)}}, \quad \text{for } n \le Q+t_{c,1}, \quad (14)$$

$$y_{c,2}(n) = \frac{A_{c,2}(n-t_{c,2})\sigma_{c,2,Total}(n-t_{c,2})[P_{n-t_{c,2}}]_Q^{(V)}}{[P_n]_Q^{(V)}}, \quad \text{for } n > Q+t_{c,2}. \quad (15)$$

Notice that in order to determine the occupancy distribution by Eq. (8) in STM it is necessary to determine the values $y_{c,s}(n)$. These parameters can be determined on the basis of Formulas (14) and (15) which in turn require the knowledge of the distribution (8). Therefore, the determination of the occupancy distribution in STM requires a construction of a special iterative program which is discussed in detail in [10].

After the determination of the occupancy distribution in STM it is possible to determine blocking probabilities for individual call classes:

$$E_c = \sum_{n=V-t_{c,2}+1}^{V}\left(1-\sigma_{c,2,Total}(n)\right)[P_n]_Q^{(V)}. \quad (16)$$

Formula (16) expresses the sum of blocking states for calls of class c in the highest $(s=2)$ load area.

3.5. Modified Threshold Values

Consider a STS in which the Q threshold for c class calls has been introduced (Fig. 1). In the model, the occupancy states of the system are divided into two load areas. Assume that the mode of operation of STS depends on the direction of the load change. To analyze the system two scenarios for its operation can be considered [11].

The first scenario assumes that the load of the system increase. This situation for class c corresponds to Fig. 3a. It is assumed that the number of demanded BBUs changes after exceeding the threshold Q from the value $t_{c,1}$ BBUs in the area $(n \leq Q)$ to the value $t_{c,2}$ BBUs in the area $(Q < n \leq V)$. In such system, the occupancy distribution can be approximated by Eq. (8) determined for the STM described in Section 3.4.

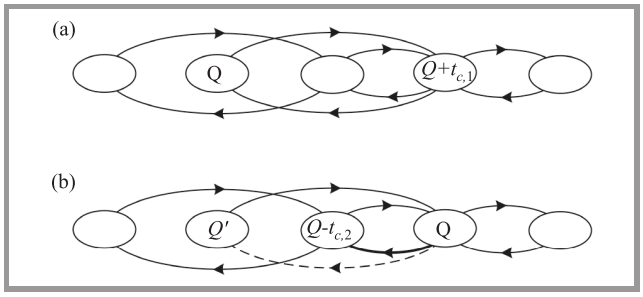


Fig. 3. Threshold for the scenarios: (a) – first, (b) – second.

The second scenario assumes that the loads of the system decrease. According to the definition of the threshold, it is the last state in which a call that demands a reduced number of BBUs $(t_{c,2})$ can appear. This transition is marked in Fig. 3b with bold line. In order to determine the occupancy distribution, the so-called residual traffic, marked with dotted line, has to be additionally considered. Residual traffic is traffic that results from calls admitted in the lower load area $(n \leq Q)$ that have not yet been terminated before the system has been transferred from the lower load area to the higher load area $(Q < n \leq V)$. The relation between threshold values for the first and the second scenarios can be determined on the basis of relation [11]:

$$Q' = Q - t_{c,2} - 1, \tag{17}$$

where Q' defines the threshold for the second scenario that corresponds to the threshold Q for the first scenario.

4. Single Hysteresis System

4.1. Single Hysteresis System – Working Idea

The operation of the single hysteresis system for calls of one class c is show in Fig. 4a. In STS, one threshold Q (Fig. 1) has been introduced, while in SHS a pairs of thresholds Q_1, Q_2 is introduced. With a change in the load from low to high, threshold Q_1 operates. With a change from high to low load, thresholds Q_2 is used. Between Q_1 and Q_2 transition areas appear that form hysteresis. In SHS, two, partly overlapping, load areas can be distinguished. In the low load area $(s = 1, 0 \leq n \leq Q_1)$, the Call Admission Control function (CAC) admits for service a new call of class c with the maximum number of BBUs, equal to $t_{c,1}$. In the high-load area $(s = 2, Q_2 < n \leq V)$ the CAC function admits a new call of class c with a lowest number of BBUs, equal to $t_{c,2}$.

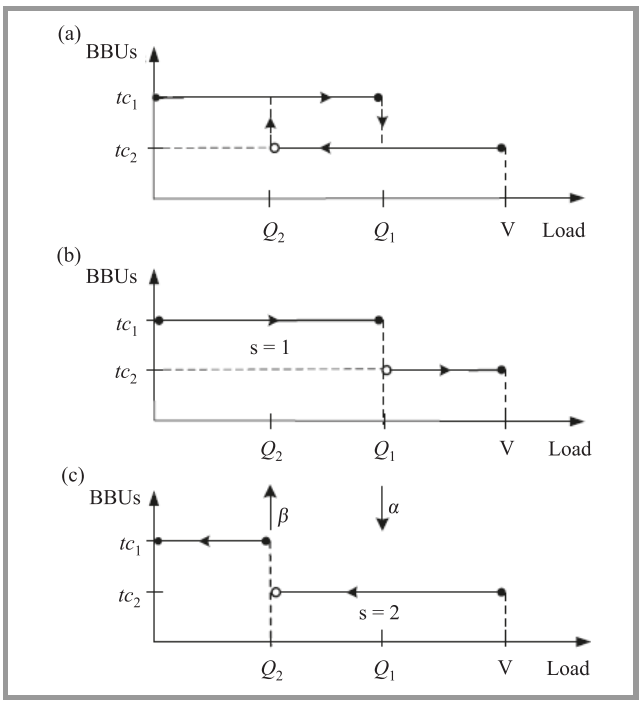


Fig. 4. System: (a) with single hysteresis mechanism and (b)–(c) its decomposition into STM components.

4.2. Occupancy Distribution in SHS

Figures 4b–c show a decomposition of SHM (Fig. 4a) into two STMs, where s indicates the area of the considered load. STMs, models are selected in such a way as to have the corresponding load area as high as possible. The arrows between Figs. 4b,c indicate possible transitions between neighboring STMs. The arrows between STM_1 (Fig. 4b) and STM_2 (Fig. 4c) indicate that the instance of exceeding of threshold Q_1 triggers a change from the STM_1 model to STM_2, whereas the instance of exceeding of threshold Q_2 is followed by a change from the STM_2 model to STM_1. The parameters α and β that correspond to the arrows define intensities of the transitions between appropriate STMs. They are determined by values of streams that exceed indicated thresholds. How these parameters are determined will be presented in the Section 5.

The transition $STM_2 \rightarrow STM_1$ is aligned with the direction of the change in the load from high to low, therefore this transition will be described by STM_2 with the threshold that corresponds to the second scenario (Sec-

tion 3.4), i.e., threshold Q_2' (Eq. (17)). Hence, the thresholds in the considered system can be written in the following way:

$$Q_x = \begin{cases} Q_x & \text{for odd } x, \\ Q_x' = Q_x - t_{c,s} - 1 & \text{for even } x, \end{cases} \tag{18}$$

where $t_{c,s}$ is the number of BBUs that is necessary to set up a connection of class c in the load area s, if threshold Q_x defines the transition $\text{STM}_s \rightarrow \text{STM}_{s-1}$.

Let's denote the occupancy distributions in STM_s presented in Fig. 4b–c with the symbols $[P_n]_{Q_1}^{(V)}$ ad $[P_n]_{Q_2'}^{(V)}$. These distributions can be determined on the basis of Eq. (8) for appropriate pairs of thresholds adopted for a given STMs (Eq. (18)). The occupancy distribution in SHM $[P_n]_{H_1,H_2}^{(V)}$ can be modeled on the basis of the weighted sum of the occupancy distributions in STM_s into which the SHM under consideration is decomposed [12]:

$$[P_n]_{H_1,H_2}^{(V)} = P(1)[P_n]_{Q_1}^{(V)} + P(2)[P_n]_{Q_2'}^{(V)}, \tag{19}$$

where $P(s)$ is the probability that SHS stays in the load area s, that corresponds to the average time the system spends in this particular load area.

4.3. Switched Process in SHM

Probabilities $P(s)$ can be determined on the basis of the two-state Markov process [11], whose diagram is presented in Fig. 5. This process is an analytical model for switches between appropriate load areas. The states in the diagram correspond to the execution of the service process in a given load area (described by a corresponding STM_s), whereas the parameters α and β denote the intensities of transitions between the appropriate load areas.

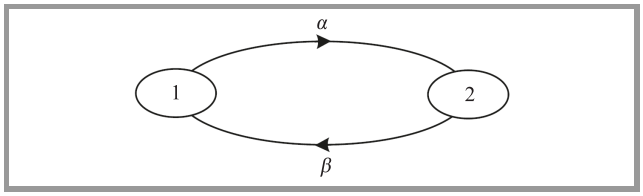


Fig. 5. Markovian switching process in SHM.

On the basis of the process presented in Fig. 5, it is possible to add and solve in a convenient way the state equations. The solution is expressed with the following formulas:

$$P(1) = \frac{\beta}{\alpha+\beta}, \; P(2) = \frac{\alpha}{\alpha+\beta}. \tag{20}$$

The intensities of transition α determine the transitions in the direction lower load → higher load and are the sum of all traffic streams that exceed the appropriate thresholds. Thus:

$$\alpha = \sum_{n=Q_1-t_{\max}+1}^{Q_1} \sum_{c \in M_1 \cup M_2 \cup M_3} A_{c,s}(n) t_{c,s} \varphi_{c,s}(n). \tag{21}$$

The parameter $\varphi_{c,s}(n)$ is calculated in the following way:

$$\varphi_{c,s}(n) = \begin{cases} 1 & \text{for } n > Q_x - t_{c,s}, \\ 0 & \text{for } n \le Q_x - t_{c,s}. \end{cases} \tag{22}$$

The intensities of transition β determine transitions in the direction higher load → lower load and are the sum of all service streams that exceed appropriate thresholds. Therefore:

$$\beta = \sum_{n=Q_2}^{Q_2+t_{\max-1}} \Bigg\{ \sum_{c \in M_1 \cup M_2 \cup M_3} y_{c,s}(n) t_{c,s} \varphi_{c,s}(n) + \\ + \sum_{c \in M_1 \cup M_2 \cup M_3} y_{c,s-1}(n) t_{c,s-1} \varphi_{c,s-1}'(n) \Bigg\}. \tag{23}$$

The parameters $\varphi_{c,s}(n)$ and $\varphi_{c,s}'(n)$ are calculated as follows:

$$\varphi_{c,s}(n) = \begin{cases} 1 & \text{for } n < Q_x + t_{c,s}, \\ 0 & \text{for } n \ge Q_x + t_{c,s}, \end{cases} \tag{24}$$

$$\varphi_{c,s}'(n) = \begin{cases} 1 & \text{for } n < Q_x + t_{c,s-1} - t_{c,s}, \\ 0 & \text{for } n \ge Q_x + t_{c,s-1} - t_{c,s}. \end{cases} \tag{25}$$

The second sum within the brace bracket in Formula (23) includes residual traffic of class c that is serviced in area s (Section 3.5).

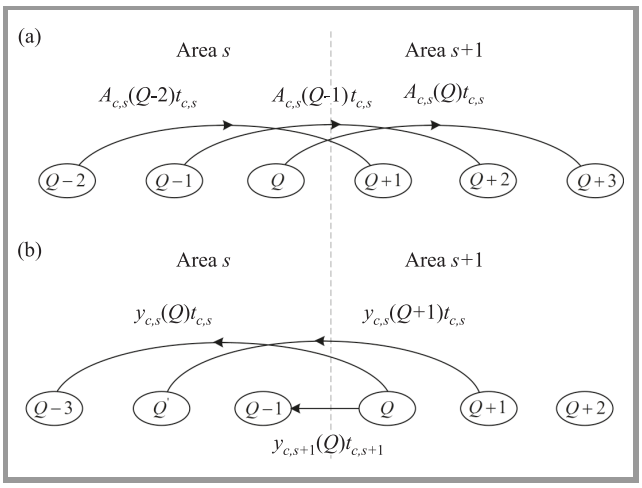


Fig. 6. Interpretation of passages: (a) $\text{STM}_s \rightarrow \text{STM}_{s+1}$ and (b) $\text{STM}_{s+1} \rightarrow \text{STM}_s$.

Figure 6a shows traffic streams of class c for the transition $\text{STM}_s \rightarrow \text{STM}_{s+1}$, whereas Fig. 6b presents service streams for the transition $\text{STM}_{s+1} \rightarrow \text{STM}_s$. The accompanying assumption is that $t_{c,s} = 3$ and $t_{c,s+1} = 1$.

5. Numerical Study

The presented method for a determination of the blocking probability in limited-availability systems with hysteresis mechanisms is an approximate method. In order to confirm the adopted assumptions, the results of the analytical calculations were compared with the simulation data. The research was carried for two systems.

The study was carried out for users demanding a set of four traffic classes. In the examined WCDMA interface with virtual links it was assumed that the SHS was applied to the second traffic class. The structure of traffic offered to considered systems can be described in the following way:

- the number of BBUs required by calls of particular classes:
  $t_{1,1} = 53$ BBUs, $t_{2,1} = 257$ BBUs, $t_{2,2} = 129$ BBUs, $t_{3,1} = 503$ BBUs, $t_{4,1} = 1118$ BBUs.

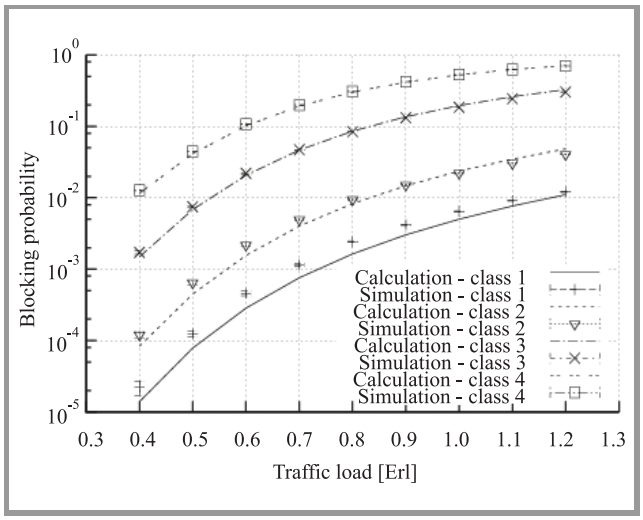


Fig. 7. Blocking probability in SHS with Engset traffic streams (System 1, $N_1 = 1000$, $N_2 = 1000$, $N_3 = 1000$ and $N_4 = 1000$).

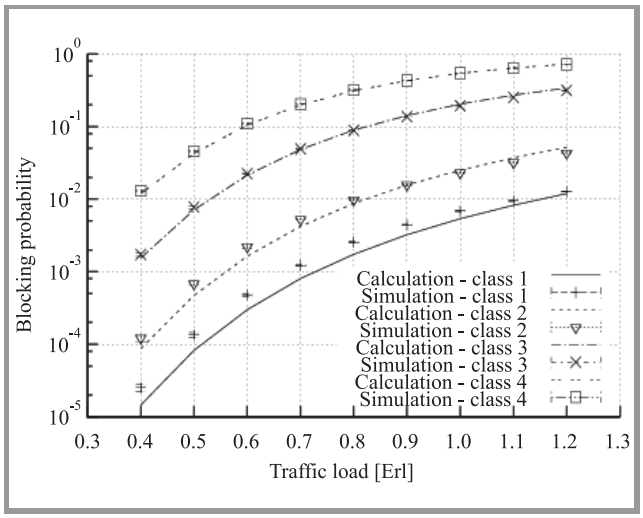


Fig. 8. Blocking probability in SHS with Erlang (class 1), Engset (class 2, $N_2 = 1000$) and Pascal (classes 3 and 4, $N_3 = N_4 = 1000$) traffic streams (System 1).

- traffic of particular classes was offered to the system in the following exemplary proportions: $A_{1,1}(0)t_{1,1} : A_{2,1}(0)t_{2,1} : A_{3,1}(0)t_{3,1} : A_{4,1}(0)t_{4,1} = 1 : 1 : 1 : 1$.
- the hysteresis thresholds are assumed to be equal to, respectively: $Q_2 = 4000$ BBUs, $Q_1 = 6500$ BBUs.

The research was carried for two systems described below:

System 1

- number of virtual links: $k = 2$.
- capacity of single virtual link: $\upsilon = 4000$ BBUs.
- total capacity of system: $V = 8000$ BBUs.

System 2

- number of virtual links: $k = 4$.
- capacity of single virtual link: $\upsilon = 2000$ BBUs.
- total capacity of system: $V = 8000$ BBUs.

The results of the research study are presented in Figs. 7–12, depending on the value of traffic a offered to a single BBU. The results of the simulation are shown in the charts in the form of marks with 95% confidence intervals that have been calculated according to the t-Student distribution for the five series with 1,000,000 calls of each class. For each of the points of the simulation, the value of the confidence interval is at least one order lower than the mean value of the results of the simulation. In many a case, the value of the simulation interval is lower than the height of the sign used to indicate the value of the simulation experiment.

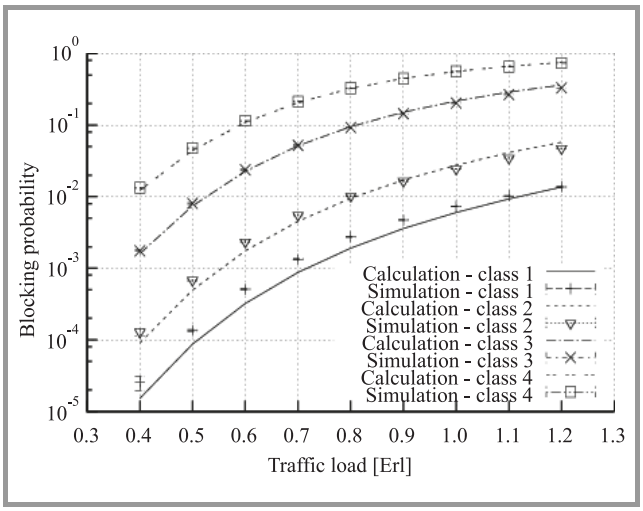


Fig. 9. Blocking probability in SHS with Pascal traffic streams (System 1, $N_1 = 1000$, $N_2 = 1000$, $N_3 = 1000$ and $N_4 = 1000$).

The results of the research study confirm high accuracy of the proposed SHM model for limited-availability group

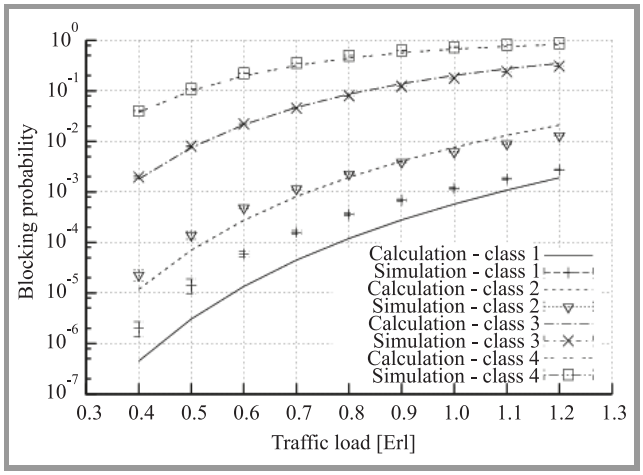


Fig. 10. Blocking probability in SHS with Engset traffic streams (System 2, $N_1 = 1000$, $N_2 = 1000$, $N_3 = 1000$ and $N_4 = 1000$).

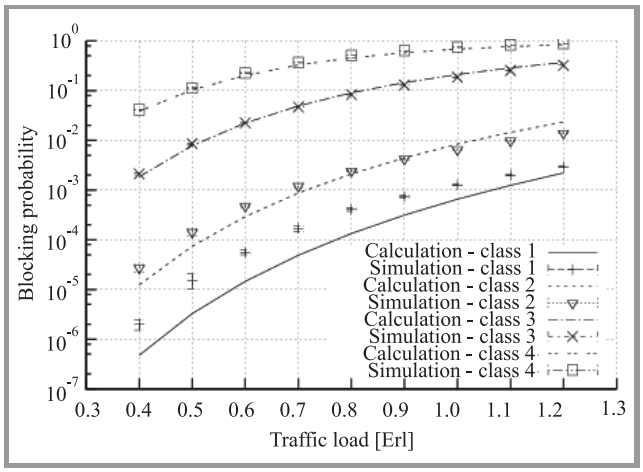


Fig. 11. Blocking probability in SHS with Erlang (class 2), Engset (class 2, $N_2 = 1000$) and Pascal (classes 3 and 4, $N_3 = N_4 = 1000$) traffic streams (System 1).

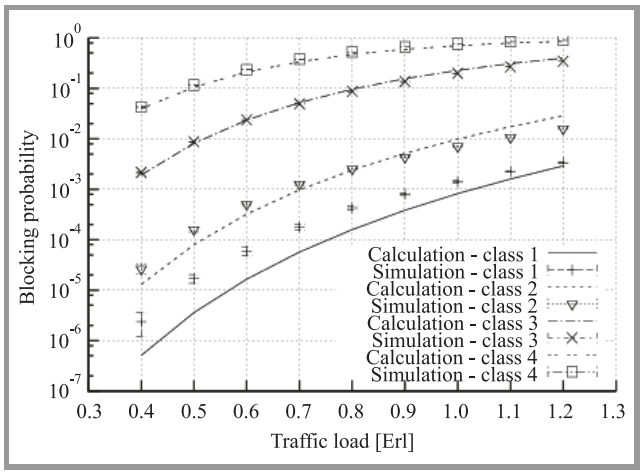


Fig. 12. Blocking probability in SHS with Pascal traffic streams (System 2, $N_1 = 1000$, $N_2 = 1000$, $N_3 = 1000$ and $N_4 = 1000$).

with BPP traffic. Greater accuracy we can obtain for traffic classes which require the largest number of BBUs.

6. Conclusions

This paper proposes a new analytical model of SHS for limited-availability group to which a mixture of different BPP traffic streams is offered. The SHS, introduced into a given system, allows the blocking probability to be decreased for particular traffic classes and leads to a reduction in fluctuations in the load. The paper also presents a possibility of the application of SHS for traffic control in the UMTS network. All the presented simulation experiments for the considered systems confirm good accuracy of the proposed analytical SHM model for traffic streams of BPP type. Summing up, the single hysteresis mechanism can be successfully used in the call admission control function of communications and cellular networks.

References

[1] M. Stasiak, M. Głąbowski, A. Wiśniewski, and P. Zwierzykowski, *Modeling and Dimensioning of Mobile Networks: from GSM to LTE*. Chichester: Wiley, 2011.

[2] J. Roberts, "Teletraffic models for the Telcom 1 integrated services network", in *Proc. 10th Int. Teletraf. Congr. ITC 83*, Montreal, Canada, 1983, p. 1.1.2.

[3] M. Pióro, J. Lubacz, and U. Körner, "Traffic engineering problems in multiservice circuit switched networks", *Comp. Netw. ISDN Sys.*, vol. 20, pp. 127–136, 1990.

[4] V. Iversen, "Teletraffic Engineering and Network Planning", Lyngby, Technical University of Denmark, 2009.

[5] Q. Huang and V. Iversen, "Approximation of loss calculation for hierarchical networks with multiservice overflows", *IEEE Trans. Commun.*, vol. 56, no. 3, pp. 466–473, 2008.

[6] M. Głąbowski, K. Kubasik, and M. Stasiak, "Modeling of systems with overflow multi-rate traffic", *Telecommun. Systems*, vol. 37, no. 1–3, pp. 85–96, 2008.

[7] H. Holma and A. Toskala, *WCDMA for UMTS: HSPA Evolution and LTE*, 5th ed. New York, London: Wiley, 2010.

[8] V. Vassilakis, I. Moscholios, and M. D. Logothetis, "Call-level performance modelling of elastic and adaptive service-classes with finite population", *IEICE Trans. Commun.*, vol. E91-B, no. 1, pp. 151–163, 2008.

[9] J. Kaufman, "Blocking with retrials in a completly shared recource environment", *Perform. Evaluation*, vol. 15, no. 2, pp. 99–113, 1992.

[10] M. Gląbowski, A. Kaliszan, and M. Stasiak, "Modeling product form state-dependent systems with BPP traffic", *Perform. Evaluation*, vol. 67, no. 2, pp. 174–197, 2010.

[11] M. Sobieraj, M. Stasiak, J. Weissenberg, and P. Zwierzykowski, "Analytical model of the single threshold mechanism with hysteresis for multi-service networks", *IEICE Trans. Commun.*, vol. 95, no. 1, pp. 120–132, 2012.

[12] M. Sobieraj, M. Stasiak, and P. Zwierzykowski, "Model of the threshold mechanism with double hysteresis for multi-service networks", in *Communications in Computer and Information Science*, vol. 291, A. Kwiecien, P. Gaj, and P. Stera, Eds. Springer, 2012, pp. 299–313.

[13] M. Stasiak, M. Sobieraj, J. Weissenberg, and P. Zwierzykowski, "Single hysteresis model for multi-service networks with BPP traffic", in *Proc. 17th Polish Teletraf. Symp.*, Zakopane, Ponad, 2012, pp. 53–58.

[14] J. Roberts, U. Mocci, and J. Virtamo, Eds., "Broadband Network Teletraffic", Final Report of Action COST 242, Springer, 1996.

[15] M. Stasiak, "Blocking probability in a limited-avalilability group carrying mixture of different multichannel traffic streams", *Annales des Télécommunications*, vol. 51, no. 11–12, pp. 611–625, 1996.

Maciej Sobieraj received his M.Sc. degree in Electronics and Telecommunications from Poznan University of Technology, Poland, in 2008. Since 2007 he has been working at the Chair of Communications and Computer Networks at the Faculty of Electronics and Telecommunications at Poznan University of Technology. He is the co-author of a dozen scientific papers. Maciej Sobieraj is engaged in research in the area of modeling of multiservice cellular systems, switching networks and traffic engineering in TCP/IP networks.

E-mail: maciej.sobieraj@put.poznan.pl
Chair of Communication and Computer Networks
Faculty of Electronics and Telecommunications
Poznan University of Technology
Polanka st 3
60-965 Poznan, Poland

Maciej Stasiak received M.Sc. and Ph.D. degrees in Electrical Engineering from the Institute of Communications Engineering, Moscow, Russia, in 1979 and 1984, respectively. In 1996 he received D.Sc. degree from Poznan University of Technology in Electrical Engineering. In 2006 he was nominated Full Professor. Between 1983–1992 he worked in Polish industry as a designer of electronic and microprocessor systems. In 1992, he joined Poznan University of Technology, where he is currently Head of the Chair of Communications and Computer Networks at the Faculty of Electronics and Telecommunications. He is the author, or co-author, of more than 250 scientific papers and five books. He is engaged in research and teaching in the area of performance analysis and modeling of queuing systems, multiservice networks and switching systems. Since 2004 he has been actively carrying out research on modeling and dimensioning cellular networks 2G/3G/4G.

E-mail: maciej.stasiak@put.poznan.pl
Chair of Communication and Computer Networks
Faculty of Electronics and Telecommunications
Poznan University of Technology
Polanka st 3
60-965 Poznan, Poland

Joanna Weissenberg received the M.Sc. degree in Mathematics from Kazimierz the Great University, Bydgoszcz, Poland in 2007. Since 2008 she is a Ph.D. student at the Chair of Communications and Computer Networks at Poznan University of Technology. Her interests include application of stochastic processes theory in telecommunication systems (queuing theory). Recently the main area of her professional activity is Markovian analysis of multirate systems in cellular networks. She is a scholarship holder within the project: "Scholarship support for Ph.D. students specializing in majors strategic for Wielkopolska's Region development".

E-mail: joannaweissenberg@gmail.com
Chair of Communication and Computer Networks
Faculty of Electronics and Telecommunications
Poznan University of Technology
Polanka st 3
60-965 Poznan, Poland

Piotr Zwierzykowski received the M.Sc. and Ph.D. degrees in Telecommunications from Poznan University of Technology, Poland, in 1995 and 2002, respectively. Since 1995 he has been working at the Faculty of Electronics and Telecommunications, Poznan University of Technology. He is currently an Assistant Professor at the Chair of Communications and Computer Networks. He is the author, or co-author, of over 200 papers and three books. He is engaged in research and teaching in the area of computer networks, multicast routing algorithms and protocols, as well as performance analysis of multiservice switching systems. Recently, the main area of his research is modeling of multiservice cellular networks.

E-mail: piotr.zwierzykowski@put.poznan.pl
Chair of Communication and Computer Networks
Faculty of Electronics and Telecommunications
Poznan University of Technology
Polanka st 3
60-965 Poznan, Poland

Paper

Interworking and Cross-layer Service Discovery Extensions for IEEE 802.11s Wireless Mesh Standard

Krzysztof Gierłowski

Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

Abstract—With the rapid popularization of mobile end-user electronic devices, wireless network technologies begin to play a crucial role as networks access technologies. While classic point-to-multipoint wireless access systems, based on fixed infrastructure of base stations providing access to clients, remain the main most popular solution, an increasing attention is devoted to wireless mesh systems, where each connecting client can extend overall resources of the network by becoming a network node capable of forwarding transit traffic. This ability results in severe reduction of the necessary network infrastructure, provides through coverage (thereby offering significant step towards ubiquity of network access) and offers massive redundancy. One of the most promising wireless mesh solutions currently being developed is an IEEE 802.11s standard, based on popular Wi-Fi technology. It combines low deployment costs with comprehensive suite of mechanisms able to operate a self-forming, autoconfigurable, dynamically extending, and secure mesh solution. However, despite its advantages, the standard lacks sufficient support for a number of functionalities, which can lead to significant inefficiency and degradation of service quality in real-world IEEE 802.11s network deployments. In the paper we propose a number of extensions of IEEE 802.11s mechanisms, designed to provide better service quality in case of real-world deployment scenarios, especially in case of large systems. Both propositions introduce modifications to mesh path discovery and interworking procedures, while retaining compatibility with standard solution. Their basic functionality and efficiency have been verified by means of simulation model in large-scale, self-organizing mesh structure. Subsequently they have been implemented and tested in real-world, access network testbed deployment. The results clearly indicate their utility, particularly in case of larger deployments of this network system type.

Keywords—802.11, cross-layer, interworking, mesh, wireless networks.

1. Introduction

The rapid growth of the quantity and capabilities of end-user electronic devices, both stationary and mobile, resulted in their use in increasing number of applications. Their popularity and high functionality created such concepts as:

- **Smart Cities**, emphasizing ubiquity of ICT-based (Information and Communication Technology) services in as many elements of our daily lives as possible,
- **Internet of Things**, proposing an introduction of electronic devices into as many elements of our physical environment as possible,
- **Cloud Computing**, bringing ability to use computing resources, both hardware and software, as services provided by ubiquitous infrastructure,
- **Machine-to-Machine** systems, allowing free interaction of different devices to extend their functionality set and overall usefulness.

It is evident from the above list of examples, that number of electronic devices will continue to grow and that an efficient communication between them is a crucial component of modern ICT systems. Requirements set before both core and access network systems are high and will continue to grow, both in terms of raw throughput and quality of service (QoS) provided. Apart from these requirements, growing percentage of mobile devices and associated services creates new requirement of ubiquitous network access - user should always be offered some form of network connectivity, as its lack would result a severe reduction of device's functionality. Modern smartphone operating systems and various "aaS" (as a Service) approaches are good examples of this trend. Wireless network technologies have a crucial role in access networks of such systems, as cable-based solutions are of limited utility in case of necessity to provide network access to easily portable or mobile devices. Moreover, their use is often preferable even in case of stationary devices, due to lack of necessity of deploying a costly and cumbersome cable infrastructure. Most of popular wireless access systems follow point-to-multipoint architecture, with operator maintained infrastructure of access points or base stations (connected with fast cable or point-to-point wireless links), each serving a set of client devices over its coverage area. Such systems must be carefully designed, taking into account both current and future signal propagation characteristics, expected user density, traffic requirements and economic constraints.

However, a new emerging type of wireless network can be used to create system, where each connecting client can extend its overall resources by becoming a network node capable of forwarding transit traffic. Such ability allows a severe reduction of the necessary operator provided infrastructure, compared to classic point-to-multipoint systems. It is also a significant step towards ubiquity of

network access, as mesh systems tend to provide through coverage. Mesh networks can be used in a variety of applications, starting from small ad hoc systems, through a highly robust and redundant infrastructure of an access network, and ending with emergency or military communication networks or self-organizing office/building/campus integrated infrastructure systems. In all of these scenarios, the main advantages of mesh networks include autoconfiguration and self-forming abilities. However, as an emerging technology, fully self-forming mesh solutions are still in process of being standardized and a number of technical problems remain to be solved.

2. IEEE 802.11s Standard

One of the most promising mesh solutions currently being developed is an IEEE 802.11s standard [1], aimed to create a broadband, fully autoconfigurable, dynamically extending, and secure mesh solution, based on widely popular Wi-Fi (IEEE 802.11 [2]) wireless local area network (WLAN) technology. It is designed to serve in wide variety of environments, starting with small ad-hoc, isolated networks (for example, groups of laptops), through industrial network deployments, office LANs, and ending with large, self-extending, public access systems. The created mesh structure is called a Mesh Basic Service Set (MBSS). The fact that this solution is based on cheap and popular Wi-Fi technology and can be deployed on existing hardware makes it one of very few mesh solutions able to successfully appear and remain on popular WLAN market. Additionally, a number of design decision have been made to make an IEEE 802.11s mesh as compatible and as easy as possible to integrate with existing network systems. Examples include mesh gates – specialized network nodes responsible for integration with external networks, hybrid routing protocol – ensuring that there are relatively small delays in routing to external destinations, higher layer transparency – making mesh network seem as a single Ethernet broadcast domain, complete with 802.1D [3] compatibility (bridging and spanning tree protocol). It is evident, that standard authors aimed to provide a robust building block for modern networks systems, both functional and inexpensive to deploy.

Due to its robustness, an IEEE 802.11s-based mesh can be used in a variety of previously described roles, including the most complex office/building/campus infrastructure system. Despite the above advantages, a number of areas in the discussed standard lack sufficient support, which can lead to significant inefficiency and degradation of service quality in real-world IEEE 802.11s network deployments.

The first serious limitation is the fact, that the standard in its current form does not include support for creating multichannel mesh networks, which leads to severe throughput degradation due to both intra and inter-path interference. This limitation is especially important in case of dense mesh networks, where each transit node directly affects a high number of neighboring nodes. The inability to perform concurrent transmissions within a given neighborhood by spreading them across a number of orthogonal frequency channels or for a single node to concurrently receive and transmit on different channels results in highly inefficient RF resource utilization. In this situation, each additional hop on the transmission path consumes high amount of limited RF resources, not only resulting in lowering QoS parameters of a given transmission, due to intra-path interference, but also affecting all neighboring transmission paths, causing inter-path interference. Figure 1 illustrates the theoretical maximum throughput as a function of number of hops in a single channel Wi-Fi mesh, where only one transmission is currently conducted (there is no inter-path interference). Two cases have been considered:

- an optimistic – where it is assumed, that each of transit nodes has only two neighbors in interference range, its predecessor and the next node on transmission path,
- a pessimistic – where it is assumed that all nodes are within interference range of each other.

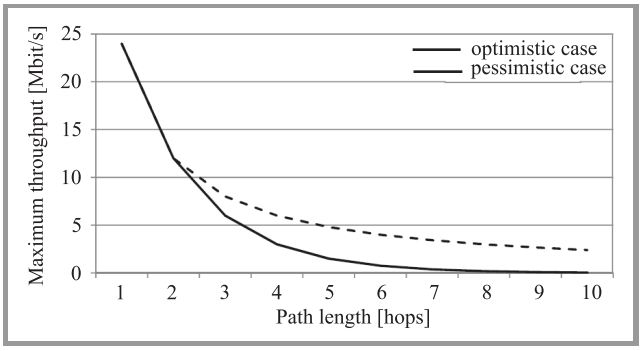


Fig. 1. Maximum theoretical single-channel mesh throughput for IEEE 802.11g transmission technology.

As can be seen from the above description of efficiency problems of RF resource usage in case of single channel mesh networks, it is crucial to limit mesh transmission path length (number of hops) as much as possible – it will result in both resource conservation, better (and more predictable) QoS level for users.

This task requires appropriate path discovery protocols, which are adequately addressed in the current standard by employing a hybrid (reactive/proactive) solution. However, we would like to propose an extension of these standard mechanisms, which can provide significant advantages in case of larger mesh structures.

3. IEEE 802.11s Path Selection Mechanism

The discussed standard utilizes Mesh Discovery and Mesh Peering Management protocols to discover neighbor nodes and create peer relationship between them, creating the base topology of a mesh system. Its operation procedures are outside of our current interest, as for the purpose of this research, we assume an already existing system topology.

To discover transmission paths over a given network topology, the IEEE802.11s system utilizes Hybrid Wireless Mesh Protocol (HWMP). This hybrid protocol, consisting of both reactive and proactive path discovery mechanisms, is able to function concurrently to provide both fast response, adherence to changing transmission conditions, and minimization of management overhead.

Moreover, the standard allows for extensibility of path selection protocols, including the ability to use alternate path discovery solutions, as long as the same, single solution is utilized uniformly through the mesh network. Despite the fact, the obligatory HWMP protocol must also be supported by all IEEE 802.11s devices, for the sake of compatibility. Peering management mechanisms are responsible for assessing node capabilities and deciding, if connecting station is able to participate in a given mesh procedures. The same mechanisms are then responsible for configuration of the newly connected node to use the appropriate path selection protocol.

The basic path discovery mechanism of HWMP is a reactive Radio Metric Ad hoc On-Demand Distance Vector (RM-AODV) protocol. It is a modification of well-known AODV protocol [4], but extended to use a radio aware link routing metric. In case of unmodified AODV metric used for path selection is a number of transmission hops on a given path. Such solution does not take into account the quality of links traversed which makes it poorly suited for wireless environment, where different links can provide a radically different transmission quality. To address these issues, RM-AODV employs Airtime Metric as a measure of link quality, taking into account both their current maximum data rate and transmission error rate – see Eq. (1)

$$c_a = \left[O + \frac{B_t}{r}\right] \frac{1}{1 - e_f}, \tag{1}$$

where c_a is link Airtime Metric value, O – technology dependent transmission overhead, r – link throughput, B_t – size of the test frame and e_f – frame error rate for a given B_t.

Link load is not taken into account directly, due to rapid and unpredictable changes of this parameter in case of wireless multihop systems, but its impact is reflected by link error rate parameter, which is significantly higher in case of highly loaded links. Airtime Metric can be seen as an amount of link resources necessary to transmit a frame.

3.1. Reactive Routing

As a reactive protocol, RM-AODV is activated by a source station, when it has a frame to send to a previously unknown destination, or destination for which forwarding information is suspected not to be current. In such case the node initiates path discovery. Each new path discovery initiated by a given station is assigned a unique (incrementing) HWMP Sequence Number. This number, along with intended destination address and path metric field set to 0 is included in Path Request (PREQ) message, subsequently sent by originating station to all of its neighbors as a broadcast HWMP Path Selection frame. Each receiving neighbor checks if it already received PREQ with:

- greater HWMP Sequence Number – the PREQ is considered to contain stale information and is discarded, as more recent PREQ already have been received,
- the same HWMP Sequence Number and the same or greater path metric – the PREQ contains current information, but it is also discarded, because the node already received PREQ of the same discovery which arrived through the better path (smaller path metric),
- the same HWMP Sequence Number and smaller path metric – the PREQ contains current information and arrived by the best path (smallest metric) yet.

In the last case the receiving station updates its forwarding information, by remembering the station that it received PREP from, as its best next-hop on a path to PREQ originator. The station then increases path metric of received PREP by the metric of the link it arrived by, and rebroadcasts it to its neighbors. That way all stations in the mesh learn a next-hop towards PREQ originator, thereby creating mesh paths toward it. It should be noted, that the described procedure is conducted with use of HWMP Path Selection Frames sent to a broadcast address and such frames are not subject to reception acknowledgement and retransmission procedures. Such solution have been chosen to lower the resource consumption and process delay at a cost possibility of missing an optimal path due to a loss of Path Selection frames.

When an intended recipient of the communication (destination station) receives PREQ which would trigger a rebroadcast according to above rules, it updates its forwarding information and, instead of re-broadcasting PREQ sends a unicast Path Replay (PREP) to PREQ originator along the just discovered (reverse) path. The PREP is forwarded by transit nodes along the path, each of them updating its forwarding information by remembering from which neighbor it received PREP, thus forming path to PREP originator. As PREP reaches the source station, a bidirectional path between the initiator of the discovery procedure and its subject is formed, by presence of current next-hop forwarding information in all transit nodes.

It should be noted, that it is possible that the transit stations will re-broadcast PREP from the same discovery procedure multiple times and the destination node will generate multiple PREP messages, if they receive multiple subsequent PREQs with decreasing path metric. However, due to the fact that smaller metric most often corresponds to shorter transmission delay, it is not likely.

It is evident that such procedure can be time consuming, especially in case of large mesh networks and distant destinations. Moreover, broadcast procedures tend to consume a considerable amount of resources. To optimize the described procedure, it is possible to allow transit stations which already have current next-hop information toward the destination station, to respond with PREP. That allows

the source station to learn the forward path to destination quickly, but its PREQ still must be broadcasted all the way towards the destination station, to form the reverse path from destination to source.

3.2. Proactive Routing

Apart from the obligatory RM-AODV protocol, the IEEE 802.11s standard defines an optional proactive path discovery solution, which can be deployed concurrently with RM-AODV. This solution, sometimes called Tree-Based Routing (TBR) protocol, consists of two independent mechanisms: Proactive Path Request (PPREQ) and Root Announcement (RANN). Both can be used to proactively create and maintain current paths between a selected mesh station (Root Mesh Station) and all other stations in the mesh. Moreover, they reuse a significant number of mechanisms of RM-AODV protocol, thereby simplifying their implementation.

In case of the first approach, PPREQ, a selected root station periodically originates PPREQ messages, which can be thought of as PREQ messages addressed to a broadcast destination. They are re-broadcasted through the network according to the same rules as in case of RM-AODV. That results in periodic refresh of unidirectional paths leading towards Root Station in all stations of the mesh. If mesh station predicts that a bidirectional path will be required, it can respond to PPREQ with unicast PPREP, which will be forwarded to Root Station and create the path in opposite direction. Root Station can also request that all stations respond in such fashion.

Due to a relatively high consumption of resources in case of PPREQ method, an alternative solution has been included in the standard – the Root Announcement mechanism (RANN). Instead of periodically sending PPREQ messages in HWMP Path Selection frames, Root Station can choose to send RANN messages instead. The first difference between those two messages is that RANN does not need to be send in a dedicated frame, but can be included in beacon frames, which are periodically broadcasted by each mesh station as a part of neighbor discovery mechanism, resulting in significant resource conservation.

Moreover, while RANN messages are propagated through the network using the same procedure as PPREQ messages, their reception does not result in updating of receiving station forwarding information. Instead they can be used to decide, if active update of this information should be undertaken. For this purpose the information from which neighbor RANN message has been received is stored, and its path metric is compared with metric of the current path that the station has to a Root Station. If the station does not have a path to Root Station or if its metric is worse than RANN metric, the station can perform a reactive, unicast, RANN assisted path discovery.

For that purpose, the station sends a unicast PREQ message to the Root, by forwarding it to the station form which it received RANN announcement. Such PREQ is ten forwarded retracing RANN path to Root Station, updating forwarding information in transit stations and forming path towards PREQ originator. Upon reception of the PREQ, Root Station responds with unicast PREP, which forms the path in opposite direction.

Due to strictly unicast nature of such discovery, it is both efficient (no broadcast flooding) and reliable, as unicast frames are subject to reception acknowledgement and retransmission.

3.3. Interworking

As stated before, IEEE 802.11s standard aims to provide easy integration of mesh network with other network technologies, in particular Ethernet wired technologies. An IEEE 802.11s MBSS integrates with external networks as another IEEE 802 access domain, complete with support for various IEEE 802.1D mechanisms (such as bridging). However, in contrast to other popular IEEE 802 technologies (for example Ethernet), the MBSS operation is not primarily based on broadcast data delivery, as such approach is not acceptable due to limited resources of wireless system.

In this situation, delivery of data to addresses unknown within MBSS cannot be conducted by simple broadcasting it to all stations for the bridging ones to receive and forward to external systems. To emulate this popular method of delivery, mesh stations connected to external networks, named Mesh Gates, support an extended suite of mesh mechanisms.

Presence of Mesh Gates is advertised in the MBSS by proactive PPREQ/PPREP messages with a special Gate Announcement field set (if Mesh Gate is also Root Station) or dedicated Gate Announcement frames (GANN), distributed in fashion similar to already described RANN messages.

Each Mesh Gate maintains a dedicated database (Proxy Information) of addresses known to be located in external networks accessible through a given gate. Moreover it forwards its contents to other Mesh Gates through MBSS with use of Proxy Update (PXU) messages. That makes all Mesh Gates aware, which gate should receive frames addressed to a particular external destination. If any other Mesh Gate receives frame proxied by other Mesh Gate, it will forward it to the correct gate through MBSS. However it should be noted, that both proxy information exchange and gate to gate data forwarding is conducted through MBSS and thus consuming limited RF resources.

Due to the fact that Mesh Gate is functional equivalent of IEEE 802.1D bridge, if more than one gate is connected to a given external network, only one of them can be active at a time, as more could lead to creation of broadcast loops. For this purpose a dedicated protocol must be employed at Mesh Gates – most often a well-known Rapid Spanning Tree Protocol (RSTP) [3].

Mesh station which cannot discover a MBSS path to a given address, assumes that it is located in external network. In such case it sends the data frame to all known Mesh Gates, for further delivery – due to RSTP activity, the frame is delivered to each external network only once, as only one

Mesh Gate is active per such network. To optimize further data delivery, Mesh Gate appropriate for a given destination sends PXU message to the station, which allows it to use a single Mesh Gate in continued transmission.

4. Path Discovery and Interworking Extensions

The described mechanisms allow for a significant flexibility in path discovery operations, taking advantage of both reactive routing optimized paths created on demand, and fast connection establishment to critical mesh locations, provided by proactive solutions. Moreover, interworking with outside networks is a subject of much attention, allowing seamless integration of mesh network with other IEEE 802 layer 2 ISO-OSI systems.

However, while IEEE 802.11s mesh network can provide connectivity with destinations in external networks, it is unable to use resources provided by these networks to support connectivity of intra-mesh destinations. Furthermore, it is unable to utilize multiple mesh gates connecting MBSS with the same external network segment (RSTP protocol disables frame forwarding in all but one) resulting in formation of longer paths within the MBSS, as Mesh Gates near a given station can be disabled by RSTP.

The proposed modifications of IEEE 802.11s mesh path discovery and forwarding protocols can be summarized as follows:

- introduction of ability to form a peering relationship between mesh gates, using external network as transmission medium,
- introduction of ability for a mesh station to use any of the mesh gates connected to a single, external network segment.

The first proposition allows using fast, cable connections frequently present between mesh gates connected to fixed infrastructure, to form transmission paths between intra-MBSS destinations. Due to the fact that Airtime Link Metric for such transmission link will be very low (throughput higher and error rate lower than in wireless technology by several order of magnitude), use of such links will be highly preferred.

Also, because it is highly probable in large mesh deployment, that there is a significant number of mesh gates able to communicate with use of external network and distributed through the mesh in more or less uniform fashion, there is a high probability of reducing the average path lengths of intra-mesh transmission. In fact, for a standard mesh deployment in large, multi-company office building, utilizing about 0.5–1 mesh gate for a single floor, the described mechanism will trend to reduce average mesh path lengths to 2–3 hops – which makes it suitable even for real-time multimedia communication.

To provide such functionality it is necessary to allow transmission of frames of IEEE 802.11s format through the external network. For the sake of simplicity and robustness (ability to function in case of different external network technologies) of the proposed solution and maintaining compatibility with IEEE 802.11s standard, to employ a simple mesh frame encapsulation in external network's unicast frames for the purpose is proposed.

Moreover, it is necessary to provide the ability for mesh gates to detect each other presence and addresses for unicast communication through the external network. In case of popular broadcast networks, such as Ethernet, an encapsulation of standard mesh peering messages and their transmission to broadcast address will serve the purpose. However we should remember that in case of wireless network, peer relation establishment and maintenance mechanisms use constantly and frequently generated messages, which are not retransmitted by receiving nodes, thereby limiting such traffic to 1-hop neighborhood. In case of broadcast transmission in wired network, such messages will be transmitted across the whole broadcast domain, consuming network resources. While dissemination of mesh gate presence information widely through the network is advantageous to our purpose, the frequency required in case of unstable IEEE 802.11 RF environment is highly superfluous in case of cable technologies. Because of that, frequency of mesh detection and peering message exchanges through the cable network need to be reduced. Moreover, a multicast group transmission can be considered instead of broadcast, if used external network technology supports it.

As an additional advantage of the described modification, mesh gate to mesh gate communication, such as proxy related management messages and gate-to-gate data forwarding, can be accomplished by forwarding appropriate messages through the external network instead of MBSS, thereby conserving network resources in general, and particularly in important Mesh Gate neighborhoods, where we expect a high levels of network traffic.

Also to allow using all existing mesh gates as points of contact with external networks is proposed, in contrast with unmodified IEEE 802.11s standard, where only one active mesh gate can be connected to the same external network segment due to already mentioned broadcast loops effect.

The proposal limits using of RSTP to disabling forwarding between Mesh Gate and external network, while leaving remaining functionality of Mesh Gate active. The frames addressed to external (or unknown) addresses, received by non-forwarding Mesh Gates will be delivered (by the mesh link through external network, due to is low Airtime Metric), to fully active gate for further forwarding. This modification will prevent the unnecessary reduction of the number of Mesh Gates which can be used by mesh station to reach external networks, thereby limiting path lengths through MBSS.

However, that station initially forwards frames with unknown address to all known Mesh Gates. Due to our modification it would result in unnecessary traffic and forwarding of multiple copies of such frames to a single external network. To prevent it, an additional element of Gate An-

nouncement messages is introduced, which uniquely identifies the network segment a sending mesh gate is connected to (Segment Identifier – SGID), by containing the address of Mesh Gate currently actively forwarding to this segment. Mesh stations will send data frames only to one Mesh Gate from a group of Mesh Gates sharing the same SGID.

To illustrate the results of introduction of the proposed extensions, a simulation experiment utilizing OMNet++ simulation engine [5] was performed. As modifications are intended mainly for larger mesh networks, a series of simulations of systems containing 100 stations deployed randomly over 2000×2000 m area were performed. A log-distance path loss propagation model has been used [6], to predict the loss a signal encounters in densely populated areas. All nodes were equipped with a single IEEE 802.11g [7] radio interface using the same RF transmission channel.

The standard IEEE 802.11s discovery and peering mechanisms were then used to form a mesh structure, but configured to not form peer links with nodes of medium or worse link quality, if the node if able to form at least one better quality peer link. It limited the number of links in the mesh structure, but improved their quality.

For each scenario a 50 simulation runs have been performed, each with 1 test TCP connection between randomly selected mesh stations at least 500 m apart, and 20 such connections generating background traffic. A chosen number of modified Mesh Gates have been activated at randomly selected stations, starting with 0 (unmodified IEEE 802.11s MBSS). All Mesh Gates are connected with a single Gigabit Ethernet network.

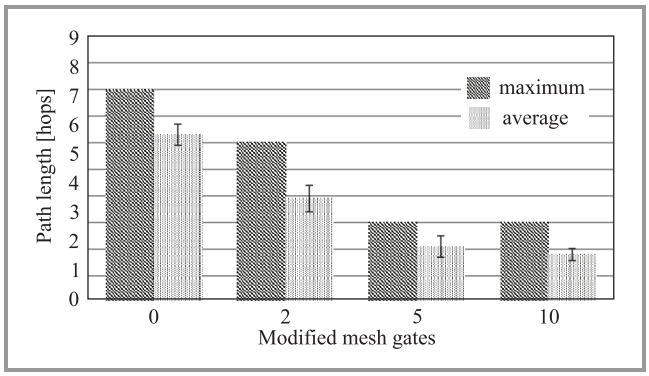


Fig. 2. Maximum and average wireless path length for different number of modified Mesh Gates.

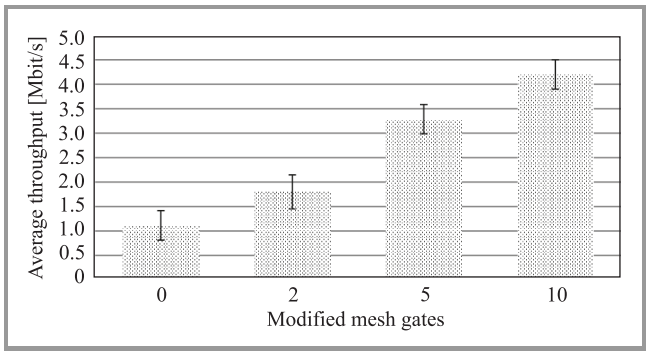


Fig. 3. Average test connection throughput for different number of modified Mesh Gates.

The results, in form of maximum and average MBSS wireless path length and average throughput for test connection, are presented in Figs. 2 and 3.

It is evident, that presence of Mesh Gates modified in proposed fashion results in reduction of both average and maximum wireless path lengths. Moreover, obtained throughput of the connection is also significantly better, despite the observed tendency to concentrate transmission paths in vicinity of Mesh Gates. Resulting concentration of traffic does not impact the transmission as bad as intra-path interference in case of longer mesh paths.

5. Cross-layer Application Discovery Extension

On the other hand, the fact that traffic sources, such as various application layer servers, can be located within a mesh network (one of its most attractive usage scenarios is self-organizing LAN/campus infrastructure), makes it important to utilize efficient application level service discovery solutions. Such mechanisms will allow clients to connect to the most appropriate server in terms of quality of service and resource consumption. Unfortunately, the fact that an IEEE 801.11s mesh is presented to higher ISO-OSI layers as a single layer 2 broadcast domain, does not allow any higher layer mechanisms to obtain reliable information about its structure. For example, IP-based service discovery procedures can be used to select a server, from a preconfigured client-side list, which provides client with the best IP connectivity at a given moment. They are not, however, taking into account an actual mesh transmission path length – for IP layer all destinations within a mesh are only 1 hop away. Such approach can lead to inefficiently long transmission paths, and dynamic organization of mesh network tends to make such paths very unstable. The analysis based on real-world device usage data in multi-office building environment confirm the above problem and indicates high QoS parameter variation for long mesh paths.

To mitigate this problem a cross-layer integration solution, allowing application level services and servers to advertise themselves using layer 2 mesh management messages is proposed. Such approach allows a mesh structure aware dissemination of information about available services and facilitates a selection of the most appropriate server for a given client. Additionally, application servers can include elements such as:

- information about supported access methods, authentication schemes and other client requirements,
- server load information,
- an initial configuration information for connecting clients,

in advertisement messages, resulting in significant reduction of initial high level service access time.

For the servers to advertise their presence, creaton a new type of IEEE 802.11s management message – Higher Layer Service Advertisement (HLSA) is proposed. This message is to be generated by active application servers in form based on RANN message. All the information present in RANN message should be included in HLSA, functionally making the generating server a Root Station using Root Announcement proactive mechanism. Furthermore, the server should include in the message additional informational elements, describing the provided application layer service. A specific set of information parameters will vary depending on a particular service type, but the information should be prepared in a way to minimize high layer message exchanges necessary to access the service. It is also advisable for the servers to set Time To Live (TTL) field in HLSA messages in accordance with their requirements of the service they provide – that way the dissemination of HLSA messages is limited to clients which are able to access the service with satisfying Quality of Experience, and network resources are conserved.

Stations receiving HLSA should process it in the same way as RANN messages, by dispersing it through the network, up to TTL limit. That way all of receiving clients will obtain information about:

- services available in the network,
- Airtime Metric and number of hops to the most appropriate server for a given client,
- expected server performance (for example a measure of current server load) and a set of application level, service specific requirements for the client, allowing it to decide if it should use a given server for service access,
- an optional set of network layer (for example IP-MAC address mapping making broadcast ARP resolution unnecessary) and service specific configuration parameters.

As a result a number of valuable advantages can be expected:

- a significant reduction of initial service access delay for connecting clients,
- easy selection of the nearest available servers resulting in shorter transmission paths, providing both good and stable transmission parameters and significant conservation of network resources,
- reduction of necessary network traffic generated by higher layer network mechanisms, such as (in case of overlying IP network): DNS request-response, ARP broadcast request-response, etc.

To verify the proposed solution the already described simulation environment have been utilized, but instead of Mesh Gates, which are not present in this scenario, a number of IP application servers have been activated randomly through the MBSS. The results, presented in Figs. 4 and 5, compare maximum and average path lengths between an application client and its chosen server for IP-based server discovery method [8] and the proposed, cross-layer HLSA procedure.

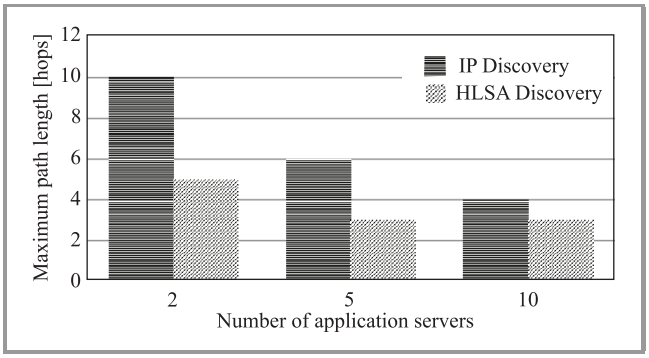


Fig. 4. Maximum client-server path length for IP and HLSA-based server discovery.

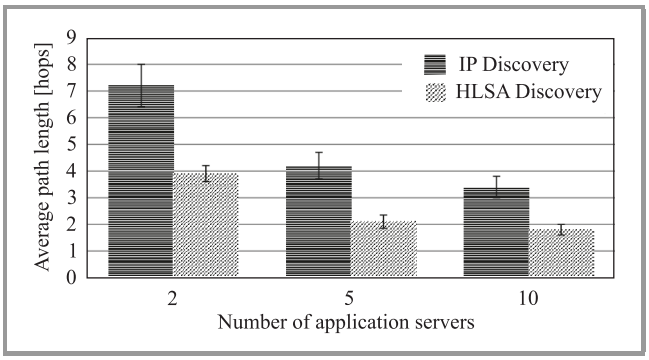


Fig. 5. Average client-server path length for IP and HLSA-based server discovery.

It can be seen, that IP-based discovery results in selection of more distant application servers, compared to the proposed solution, which provides clients with information necessary to select the server with the best Airtime Metric. The cause for this effect is a large number of factors (originating from both ISO-OSI layer 2 and 3 mechanisms) which impact relatively high-level IP-based communication quality assessment. Also, limited time which can be devoted to such on-demand assessment makes the result prone to errors caused by short-time fluctuations of path quality. It can be observed that the longer the distance between client and server, the more IP discovery is prone to errors and less predictable. On the other hand, Airtime Metric values for wireless links are calculated by mesh stations proactively and over longer periods of time, producing more reliable results. The use of proposed mechanism allows that information to be delivered to the client almost instantly. The proposed HLSA mechanism is clearly able to provide shorter paths, thus conserving network resources and providing better service to clients.

Additionally, as an example of advantages in initial service connection time provided by the solution, Table 1 contains a comparison of popular DNS-based service discov-

Table 1
Example IP-based and cross-layer server discovery procedure timings

Step	IP Discovery	Time	HLSA	Time
1 Service discovery	Broadcast ARP Request-Resp. (for predefined DNS server)	< 1 s	Gathering/comparing HLSA announcements	100 – 800 ms
	DNS Query-Response (for SRV records)	< 200 ms		
2 Server selection	Broadcast ARP Request-Resp. (for obtained set of 10 application servers)	< 1,5 s		
	ICMP Test Request-Resp. (for obtained set of 10 application servers)	< 5 s		
3 Server connection	Connection to server and initial, service specific handshake	< 500 ms	HLSA assisted mesh path discovery	< 300 ms
			Connection to server and initial, service specific handshake initiated by HLSA	< 300 ms
	Total time	< 8.2 s		< 1.4 s

ery [9] and server selection method deployed in previously described, simulated IEEE 802.11s environment, with the proposed HLSA-based mechanism.

It should be noted, however, that the presented time values are highly dependent upon mesh structure and various network and service configuration parameters as well as network load, and, as such, should be threated only as an example.

Also, highly inefficient IP ARP resolution can be eliminated altogether by including appropriate layer 3/2 mappings in the HLSA messages. An optional inclusion of initial, service specific configuration parameters can allow further time gain, but will most often require a modification of server and client software.

6. Testbed Experiments

Due to clearly advantageous impact of proposed extensions of path discovery mechanisms on efficiency of mesh operation, a practical implementation of the described mechanisms have been created.

A standard Linux kernel 3.6.10-4 implementation of IEEE 802.11s mechanisms have been used as a base and extended with:

- additional topology control functions, aimed to disallow unnecessary formation of low quality mesh links in dense mesh environment,
- ability to form peering relationships between mesh gates through external, wired infrastructure,
- ability to utilize any mesh gate present for communication with external networks.

The implementation have been tested in testbed installation designed to verify the proposed solutions in IEEE 802.11s wireless mesh deployed in office building environment. For this purpose, twenty mesh stations have been activated in 50×15 m area of two subsequent floors of Faculty of Electronics, Telecommunications and Informatics of Gdańsk University of Technology. Of these 20 stations, 2 on each floor were designed as mesh gates connected to a common wired infrastructure. Stations utilized a single, least utilized at a time of each test run, frequency channel in 2.4 GHz ISM band and IEEE 802.11g transmission technology.

The test scenario involved a single, 5 minute long, TCP test transmission between a two randomly selected mesh stations located at least 15 meters apart, while 3 other such TCP connections provided background traffic. A selected number of standard mesh gates have been substituted with modified ones: 0 (standard IEEE 802.11s MBSS), 2, 3, and 4.

For each number of modified mesh gates the test has been performed 20 times during working hours and 20 times during nighttime, to help assess the impact of external system interference on our mesh installation. The state of wireless medium have been additionally monitored with use of physical layer analyzer, to detect potential anomalous conditions such as particularly strong interfering signals. The results, in form similar to these of simulation experiment (maximum and average MBSS path length and average throughput for test connection), are presented in Figs. 6 and 7.

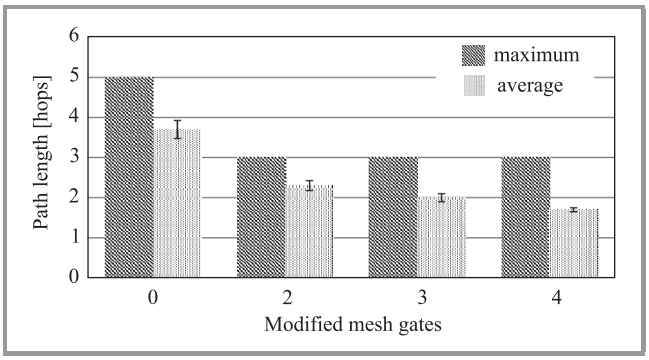


Fig. 6. Maximum and average path length for different number of modified Mesh Gates in testbed environment.

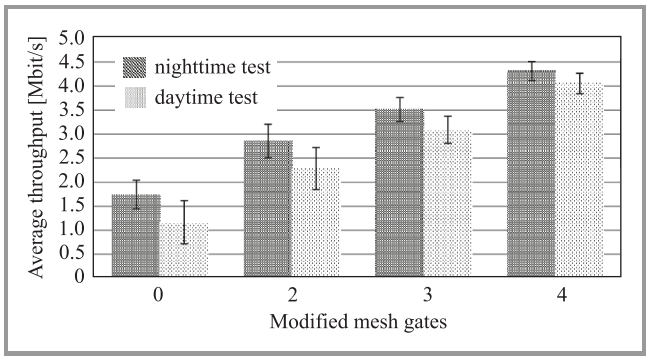


Fig. 7. Average test connection throughput for different number of modified Mesh Gates for daytime and nightime test in testbed environment.

Testbed installation is significantly smaller than the system previously considered in simulated scenario, which, combined with the fact that its node layout is fixed, results in more uniform path length results visible in Fig. 6. Despite the above fact it is evident, that simulation and testbed results show distinct similarities, confirming utility of the proposed solution. In both cases length of transmission paths through wireless domain is significantly shortened, which results in both higher and more stable throughput, as shown in Fig. 7. It can also be observed that external interference in testbed system is an important factor, resulting in reduced and less stable throughput. Unsurprisingly, the effect is more pronounced during working-hours, when external wireless system located in the area show significant activity, making it even more important to minimize path lengths through wireless MBSS.

7. Conclusions

All proposed extensions retain full compatibility with IEEE 802.11s standard, taking advantage of its inbuilt customization functions and concentrating rather on extending their scope of usage then introducing completely new solutions in place of already standardized ones. Proposed solutions can be generally classified as integration mechanisms, aiming to provide smooth network system operation across network and ISO-OSI layer boundaries.

Their basic functionality and efficiency have been verified by means of simulation model in sizable (100 nodes), self-organizing mesh deployment scenario and subsequently by testbed experiment in real-world environment. The results clearly indicate their expected utility in IEEE 802.11s mesh deployments, and further studies concerning their impact on specific applications performance are currently being conducted.

Presented solutions are part of an ongoing research concerning cross layer integration, interworking and mobility support for IEEE 802.11s-based systems.

Acknowledgments

Work supported in part by the Ministry of Science and Higher Education, Poland, under Grant N519 581038.

References

[1] "IEEE Standard for Information Technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements" – Part 11: "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking", IEEE Standard 802.11s, 2011.

[2] "IEEE Standard for Information technology-Telecommunications and information exchange between systems Local and metropolitan area networks – Specific requirements" – Part 11: "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", IEEE Standard 802.11-2012 (Revision of IEEE Std 802.11-2007), pp. 1–2793, 2012.

[3] "IEEE Standard for Local and metropolitan area networks: Media Access Control (MAC) Bridges", IEEE Standard 802.1D, 2004.

[4] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", RFC 3561 (Experimental), Internet Engineering Task Force, July 2003 [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[5] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment", in *Proc. 1st Int. Conf. Simul. Tools Tech. Commun. Netw. Sys. Worksh. SIMUTools 2008*, Marseille, France, 2008, pp. 1–10.

[6] M. Hidayab, A. Ali, and K. Azmi, "Wi-Fi signal propagation at 2.4 GHz", in *Proc. Asia Pacific Microw. Conf. APMC 2009*, Singapore, 2009, pp. 528–531.

[7] "IEEE Standard for Information Technology – Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – Specific Requirements" – Part 11: "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", IEEE Standard 802.11g, 2003.

[8] A. Delphinanto, T. Koonen, and F. den Hartog, "Real-time probing of available bandwidth in home networks", *IEEE Commun. Mag.*, vol. 49, no. 6, pp. 134–140, 2011.

[9] M. Cotton, L. Eggert, J. Touch, M. Westerlund, and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", RFC 6335 (Best Current Practice), Internet Engineering Task Force, Aug. 2011 [Online]. Available: http://www.ietf.org/rfc/rfc6335.txt

Krzysztof Gierłowski works as a researcher and lecturer at Department of Computer Communications, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Poland. He has published over 60 research papers to date and has taken part in numerous research and engineering projects. His scientific interests include local and metropolitan wireless networks, mobility support mechanisms, host and network virtualization, IP network systems, security of computer networks and systems and e-learning solutions. Designer and administrator of production grade computer systems, including multiservice corporate and access networks.

E-mail: krzysztof.gierlowski@eti.pg.gda.pl
Faculty of Electronics, Telecommunications
and Informatics
Gdańsk University of Technology
Gabriela Narutowicza st 11/12
80-233 Gdańsk, Poland

Paper

Cooperative Games with Incomplete Information for Secondary Base Stations in Cognitive Radio Networks

Jerzy Martyna

Institute of Computer Science, Faculty of Mathematics and Computer Science, Jagiellonian University, Krakow, Poland

Abstract—Cognitive radio (CR) technology is considered to be an effective solution for enhancing overall spectrum efficiency. Using CR technology fully involves the providing of incentives to Primary Radio Networks (PRNs) and revenue to the service provider so that Secondary Base Stations (SBSs) may utilize PRN spectrum bands accordingly. In this paper, a cooperative games with incomplete information for SBSs in a CR network is presented. Each SBS can cooperate with neighboring SBSs in order to improve its view of the spectrum. Moreover, proposed game-theory models assume that the devices have incomplete information about their components, meaning that some players do not completely know the structure of the game. Using the proposed algorithm, each SBS can leave or join the coalition while maximizing its overall utility. The simulation results illustrate that the proposed algorithm allows us to reduce the average payoff per SBS up to 140% relative to a CR network without cooperation among SBSs.

Keywords—Bayesian equilibrium, cognitive radio networks, game theory, wireless communication.

1. Introduction

Cognitive radio (CR) was first proposed by J. Mitola [1] as a way of "scavering" fragments of unused spectrum and designing signals accordingly. In the United States, the Federal Communications Commission (FCC) later come up with its Spectrum Policy Task Force (SPTF) report [2] that opened up the television band as a start for CR purposes. Moreover, the most recent FCC measurement [3] concludes that 70% of allocated spectrum is not utilized in the United States. Because of this, CR technology is considered one of the most attractive candidates to tackle such challenge [4].

Node cooperation is fundamental to ensure acceptable performance in CR networks. Cooperation in CR networks has been studied by, among others, A. Ghasemi *et al*. In their work, the authors showed that the collaboration among SUs (Secondary Users) and the effects of the hidden terminal problem can be reduced and the probability of detecting the PU (Primary User) can be improved [5]. Moreover, Zhang [7] has proposed that collaborative spectrum-sensing spatial diversity techniques for improving collaborative spectrum-sensing performance by detecting means of combating error probability caused by fading the reporting channel between the SUs and the central fusing center. Thus, it is obvious that the deployment of all available PUs in the exploration and use of the spectrum is a key technology for the development of cognitive radio systems. It allows to improve the quality and the amount of information transmitted over radio channels. Unfortunately, the PUs may belong to different service providers, and they interact with each other by means of the cooperation between the management centers.

In the literature, the Cognitive Pilot Channel (CPC) has been proposed by P. Houz *et al*. [8] and M. Filo *et al*. [9] as a means of providing frequency and geographical information to cognitive users. As explained in these papers, the CPC concept is based on control channels that carry information such as available spectrum opportunities and existing frequencies. Additionally, Sallent has proposed a broadcast and on-demand method for delivering the CPC data to the SUs [10]. According to this approach, each SU can exchange its own information about the entire spectrum to identify spectrum holes and available PUs. Collaboration between the SBSs can lead to a significant decrease of the costs of use and can improve the network structure's stability. In other words, a network of SBSs in every CR network is responsible for gathering all information about new PUs (the view of the spectrum, the position change detection, etc.). Recently, W. Saad has proposed a coalition formation among SBSs that can account for the tradeoffs between the costs of receiving inaccurate information and the benefit from learning about new channels through coalition members [11].

Game theory is an essential tool for CR networks. Most games considered in these systems are games with complete information. For example, games with complete information have been studied in the distributed collaborative spectrum sensing [12], for the sake of a dynamic spectrum sharing [13], interference minimalization in the CR networks [14], designing independent parallel channels (i.e., OFDM) [15], etc. However, there are no existing methods of calculating the equilibrium policy in a general game with incomplete information. The imperfect information or partial Channel State Information (CSI) means that the CSI is not perfectly estimated/observed at the transmitter/receiver side. This is a common situation which usually happens in a real wireless communication, since it may be too "expensive" for every radio receiver/transmitter to keep the information from the channels of all other devices.

Harsanyi and Selten [18] at first proposed an extension of the Nash solution to Bayesian bargaining problems. A new generalization of the Nash bargaining solution for two player games with incomplete information was presented by Myerson [19].

The main contributions of this paper are twofold. Firstly, the coalition formation among SBSs with incomplete information in the CR networks is studied. A Bayesian equilibrium which allows to formulate the study of the coalition formation among SBSs with incomplete information in CR network is given. Secondly, an algorithm for building a coalition of SBSs is formulated. Each SBS decides to enter or leave the coalition for the sake of maximizing its utility function. Finally, the system model is validated through simulation.

The remainder of this paper is organized as follows. In Section 2 the system model is presented in details. Section 3 is devoted to the coalition formation among the SBSs with incomplete information. In Section 4, an algorithm for the coalition formation of SBSs with incomplete information is described. Section 5 presents some simulation results. The paper and its possible extensions are summarized in the concluding remarks of Section 6.

2. The System Model

In this section, the model of the CR system consisting of the PUs and the SUs is presented. Assuming that to the CR network also belong N secondary base stations (SBSs). Each i-th SBS can service number L_i of SUs in a specific geographical area. It means that each SBS provides coverage area for a given cell or mesh. Let $\mathscr{N}$ be the set of all SBSs and $\mathscr{K}$ be the set of all PUs. Each PU can use a number of admissible wireless channels. We assume that each SU can employ the k-th channel of PU, if the k-th channel is not transmitting and this channel is available for the SU. According to the approach given by D. Niyato [21] each i-th SBS can be characterized by accurate statistics regarding a subset $\mathscr{K}_i \in \mathscr{K}$ of PUs during the period of time the channels remain stationary.

Let each i-th SBS use energy detectors which belong to the main practical signal detectors in the CR network. Assuming the Raleigh fading, the probability that the i-th SBS accurately received the signal from PU $k \in \mathscr{K}$ is given by [5]

$$P^i_{det,k} = e^{-\frac{\lambda_{i,k}}{2}} \sum_{n=0}^{m-2} \frac{1}{n!} \left(\frac{\lambda_{i,k}}{2} \right)^n + \left. \frac{1+\overline{\gamma}_{k,i}}{\overline{\gamma}_{k,i}} \right)^{m-1} \times \left[e^{-\frac{\lambda_{i,k}}{2(1+\overline{\gamma}_{k,i})}} - e^{-\frac{\lambda_{i,k}}{2}} \sum_{n=0}^{m-2} \frac{1}{n!} \left. \frac{\lambda_{i,k}\overline{\gamma}_{k,i}}{2(1+\overline{\gamma}_{k,i})} \right)^n \right], \quad (1)$$

where $\lambda_{i,k}$ is the energy detection threshold selected by the i-th SBS for sensing the k-th channel, m is the time bandwidth product. $\overline{\gamma}_{k,i}$ is the average SNR of the received signal from the k-th PU and is given by $\overline{\gamma}_{k,j} = \frac{P_k g_{k,i}}{\sigma^2}$, where P_k is the transmit power of the k-th PU, $g_{ki} = \frac{1}{d^{\mu}_{k,i}}$ is the path loss between the k-th PU and the i-th SBS, d_{ki} is the distance between the k-th PU and the i-th SBS, σ^2 is the Gaussian noise variance.

Thus, as was shown in [5] the false alarm probability perceived by the i-th SBS $i \in \mathscr{N}$ over the k-th channel, $k \in \mathscr{K}$, belonging to PU, is given by

$$P^i_{fal,k} = P_{fal} = \frac{\Gamma(m, \frac{\lambda_{i,k}}{2})}{\Gamma(m)}, \quad (2)$$

where $\Gamma(.,.)$ is the incomplete gamma function and $\Gamma(.)$ is the gamma function.

The non-cooperative false alarm probability depends on the position of SU. Thus, the index k in Eq. (2) could be dropped, and the missing probability perceived by the i-th SBS, is [5], [6]

$$P_{mis,i} = 1 - P_{det,i}. \quad (3)$$

Assuming a non-cooperative collaboration for every i-th SBS, $i \in \mathscr{N}$ the amount of information which is transmitted to the SUs served by it over its control channel can be obtain from

$$v(\{i\}) = \sum_{k \in \mathscr{K}_i} \sum_{j=1}^{L_i} [(1 - P^i_{fal,k})\theta_k \rho_{ji} - \alpha_k (1 - P^i_{det,k}) (1-\theta_k)(\rho_{kr_k} - \rho^j_{kr_k})], \quad (4)$$

where L_i is the number of SUs served by the i-th SBS, α_u is the penalty factor imposed by the k-th PU for the SU that causes the interference. $(1 - P^i_{det,k})$ defines the probability that the i-th SBS treated channel k as available while the PU is actually transmitting. The probability $(\rho_{kr_k} - \rho^j_{kr_k})$ indicates the reduction of a successful transmission at its receiver r_k of the k-th PU at its receiver r_k caused by the transmission from the j-th SU over k-th channel. It means the probability that the SNR received by the i-th SBS is given by [22]

$$\rho_{ji} = e^{-\frac{v_0}{\overline{\gamma}_{j,i}}}, \quad (5)$$

where v_0 is the target SNR for all PUs, SUs, SBSs, $\overline{\gamma}_{j,i}$ is the average SNR received by the i-th SBS from all SUs with the transmit power P_j of the j-th SU. It is defined as

$$\overline{\gamma}_{ji} = \frac{P_i g_{kr_k}}{\sigma^2 + g_{ji} P_j}, \quad (6)$$

where P_i gives PU i's probability of successful transmission at its receiver r_i, g_{ji} is the channel gain between the j-th SU and the i-th SBS.

Assuming Rayleigh fading and BPSK modulation within each coalition, the probability of reporting error between the i-th SBS and the j-th SU [7] is given by

$$P_{e,i,j} = \frac{1}{2} \left. 1 - \sqrt{\frac{\overline{\gamma}_{j,i}}{2+\overline{\gamma}_{j,i}}} \right). \quad (7)$$

Inside a coalition C by a collaborative sensing, the missing and the false alarm probabilities of a coalition is given by:

$$Q_{mis,C} = \prod_{i \in C} [P_{mis,i}(1 - P_{e,i,j}) + (1 - P_{mis,j})P_{e,i,j}], \quad (8)$$

$$Q_{fal,C} = 1 - \prod_{i \in C} [(1 - P_{fal})(1 - P_{e,i,j}) + P_{fal}P_{e,i,j}]. \quad (9)$$

3. The Coalition Formation Among Secondary Base Stations with Incomplete Information in the CR Networks

In this section the SBS game with incomplete information in CR network is presented.

The problem can be formulated with the help of using a cooperative game theory [16]. More formally, we have a (Ω, u) coalition game, where Ω is the set of players (the SBSs) and u is the utility function or the value of the coalition.

Following the coalition game of Harsanyi [18], a possible definition for a Bayesian game [17] is as follows.

Definition 1 (Bayesian game)

A Bayesian game $\mathscr{G}$ is a strategic-form game with incomplete information, which can be described as follows

$$\mathscr{G} = \langle \Omega, \{\mathscr{T}_k, \mathscr{A}_k, \rho_k, u_k\}_{k \in \mathscr{K}} \rangle \quad (10)$$

which consists of:

- a player set: $\Omega = \{1, \ldots, N\}$,
- a type set: $\mathscr{T}_n (\mathscr{T} = \mathscr{T}_1 \times \mathscr{T}_2 \times \cdots \times \mathscr{T}_N)$,
- an action set: $\mathscr{A}_n (\mathscr{A} = \mathscr{A}_1 \times \mathscr{A}_2 \times \cdots \mathscr{A}_N)$,
- a probability function set: $\rho_n : \mathscr{T}_n \to \mathscr{F}(\mathscr{T}_{-n})$,
- a payoff function set: $u_n : \mathscr{A} \times \mathscr{T} \to \mathscr{R}$, where $u_n(a, \tau)$ is the the payoff of player n when action profile is $a \in \mathscr{A}$ and type profile is $\tau \in \mathscr{T}$.

The set of strategies depends on the type of the player. Additionally, it is assumed that the type of the player is relevant to his decision. The decision is dependent on information which it possesses. A strategy for the player is a function mapping its type set into it action set. The probability function ρ_k represents the conditional probability $\rho_k(-\tau_k|\tau_k)$ that is assigned to the type of profile $\tau_{uk} \in \mathscr{T}_{-k}$ by the given τ_k.

The payoff function of player k is a function of strategy profile $s(.) = \{s_1(.), \ldots, s_K(.)\}$ and the type profile $\tau = \{\tau_1, \ldots, \tau_K\}$ of all players in the game and is given by

$$u_k(s(\tau), \tau) = u_k(s_1(\tau_1), \ldots, s_N(\tau_N), \tau_1, \ldots, \tau_N). \quad (11)$$

In a strategic-form game with complete information, each player chooses one action. In a Bayesian game each player chooses a set or collection of actions, strategy $s_k(.)$.

A definition for a payoff of player in the Bayesian game as follows:

Definition 2 (The player's payoff)

The player's payoff in a Bayesian game is given by

$$u_k(\tilde{s}_k(\tau_k), s_{-k}(\tau_{-k}), \tau) = u_k(s_1(\tau_1), \ldots, \tilde{s}_k(\tau_k), s_{k+1}(\tau_{k+1}), \ldots, s_N(\tau_N), \tau), \quad (12)$$

where $\tilde{s}_k(.), s_{-k}(.)$ denotes the strategy profile where all players play $s(.)$ except player k.

Next, we define the Bayesian equilibrium (BE) as follows:

Definition 3 (Bayesian equilibrium)

The strategy profile $s^*(.)$ is a Bayesian equilibrium (BE), if for all $k \in \mathscr{N}$, and for all $s_k(.) \in S_k$ and $s_{-k}(.) \in S_{-k}$

$$E_\tau [u_k(s_k^*(\tau_{-k}), \tau)] \geq E_\tau \left[u_k(s_k(\tau_k), s_{-k}^*(\tau_{-k}), \tau)\right], \quad (13)$$

where

$$E_\tau [u_k(x_k(\tau_k), x_{-k}(\tau_{-k}), \tau)] \triangleq \sum_{\tau_{-k} \in \mathscr{T}_{-k}} \rho_k(\tau_{-k} \mid \tau_k), u_k(x_k(\tau_k), x_{-k}(\tau_{-k}), \tau), \quad (14)$$

is the expected payoff of player k, which is averaged over the joint distribution of all players' types.

For the proposed game the false alarm probabilities for the i-th and j SBSs over channel k are given by $P_{fal,k}^i$ and $P_{fal,k}^j$. Thus, the utility function or the value of the coalition is given by $u(C)$, namely

$$u(C) = (1 - Q_{mis,C}) - Cost(Q_{fal,C}), \quad (15)$$

where $Q_{mis,C}$ is the missing probability of coalition C.

For the cooperation problem the following definition can be provided [16].

Definition 4 (Transferable utility of coalitional game)

A coalitional game (Ω, u) is said to have a transferable utility if value $u(C)$ can be arbitrarily apportioned between the coalition players. Otherwise, the coalitional game has a non-transferable utility and each player will have their own utility within coalition C.

Based on these concerns, it is important to say, that the utility of coalition C is equal to the utility of each SBS in the coalition. Thus, the used (Ω, u) coalitional game model has a non-transferable utility. In the coalitional game the stability of the grand coalition of all the players is generally assumed and the grand coalition maximizes the utilities of the players. Then, player i may to choose the randomized strategy s which maximizes his expected utility. Informally, we could provide a Nash equilibrium here.

Assuming the perfect coalition of SBS C_{per}, the false alarm probability is given by

$$Q_{fal,C_{per}} = 1 - \prod_{i \in C_{per}} (1 - P_{fal}) = 1 - (1 - P_{fal})^{|C_{per}|}. \quad (16)$$

4. The Coalition Formation Algorithm

In this section, an algorithm for the coalition formation of SBS with incomplete information in CR networks is proposed. The algorithm works on two levels: the possible coalition formation and the grand coalition formation. The first level is the basis for all the coalitions formation. Each member of group C cooperates so as to maximize their collective payoff. At this level a maximum number of SBSs per coalition is defined. At the second level the grand coalition is formed. Firstly, the utility function of the formed coalition is calculated. If the utility function of formed coalition reaches the maximum value, the Bayesian equilibrium (BE) is tested for the coaltion. Finding the Bayesian equilibrium (BE) finishes the operation of the algorithm. If two or more coalitions possess the Bayesian equilibrium with the same value of the payoff, the normalized equilibrium introduced by Rosen [20] is proposed here exists, where it is shown that a unique equilibrium exists if the payoff functions satisfy the condition of the diagonal strictly concave.

Algorithm 1 shows the pseudo-code of the proposed algorithm.

Algorithm 1: BSSs coalition formation

Input: False alarm probability for each coalition

```
|C_perf| := 1;
compute Q_fal,C;
while Q_fal,C > Q_fal,C_perf do
    for i = 1 to N do
        compute Q_fal,C;
        if BE exists for given |C_perf| then
            |C_perf| := i; go to 10;
        else
            |C_perf| := i + 1
        end
    end
    label 10;
end
```

5. Simulation Results

A simulation was used to confirm the above given algorithm for the coalition formation among the SBSs with incomplete information. The simulation of the CR network has a four square with the PU at the center. Each square is equal to 1×1 km. In each square 4 SBSs and 8 SUs were randomly deployed. Initially, it was assumed that each SBS is non-cooperative and detects information from its neighbors by means of the common channels. The energy detection threshold $\lambda_{i,k}$ for an i-th SBS over channel k was chosen following the false probability $P^i_{f,k} = 0.05$, $\forall i \in \mathcal{N}$, $k \in K$. The transmit power of all the SU was assumed as equal to to 10 mW, the transmit power of all the PUs was equal to 100 mW, the noise variance $\sigma^2 = -90$ dBm.

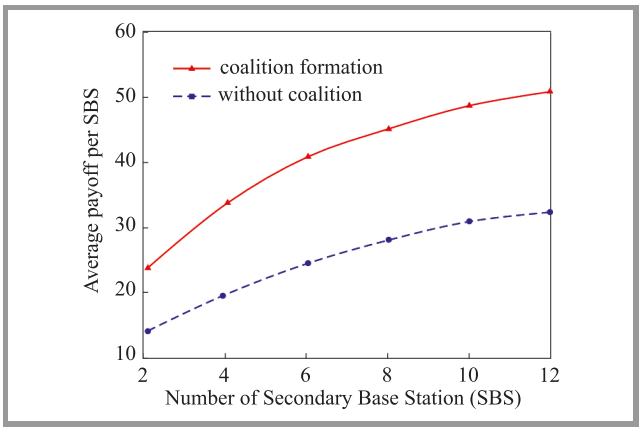


Fig. 1. The average payoff per SBS versus the number of SBSs.

Figure 1 presents the average payoff per SBS versus the number of SBSs for both the organization of the CR network with a coalition of SBSs and the non-cooperation of SBSs. Both results are averaged over random positions of all the nodes (SUs and PUs). In the case of non-cooperation game of SBS the average payoff per SBS has a smaller value than for the cooperation game of SBS. The proposed algorithm significantly increases the average payoff up to 140% relative to the non-cooperative case at the number of 15 SBSs. Figure 2 shows the number of SBSs versus the average maximum coalition number. The graph shows that the number of SBSs increases with the maximum average coalition number. It is due to the fact that as N increases, the number of potential members of the coalition increases. The graph indicates that the typical size of the SBS coalition is proportional to the num-

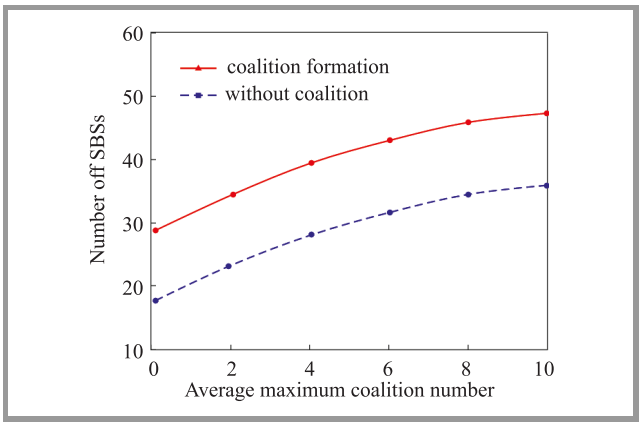


Fig. 2. The number of SBSs versus the average maximum coalition number of SBSs.

ber of SBSs for the certain value. A large number of SBSs does not allow for the formulation of a relatively large coalition.

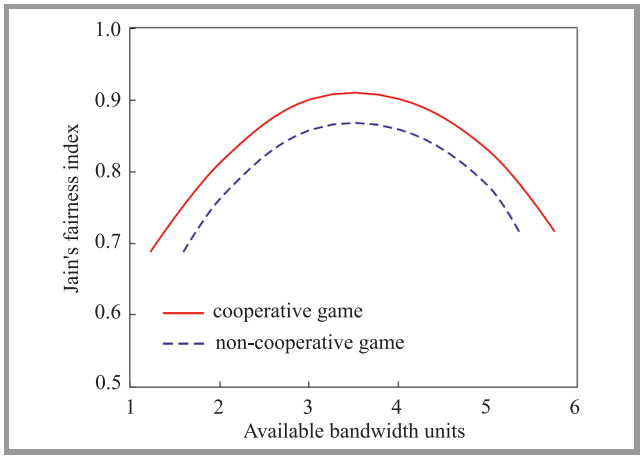


Fig. 3. Jain's fairness index (JFI) for available bandwidth units for cooperative and non-cooperative games.

Figure 3 shows that the coalition among SBSs allows us to obtain a higher value of Jain's fairness index for available bandwidth units. Jain's fairness index is defined as [23]

$$JFI = \frac{(\sum_i^{N_b} x_i)^2}{N_b \sum_{i=1}^{N_b} x_i^2}, \tag{17}$$

where x_i denotes a bandwidth unit and N_b is the number of all bandwidth units. The results show that all cooperation games take the maximum values of the JFI index.

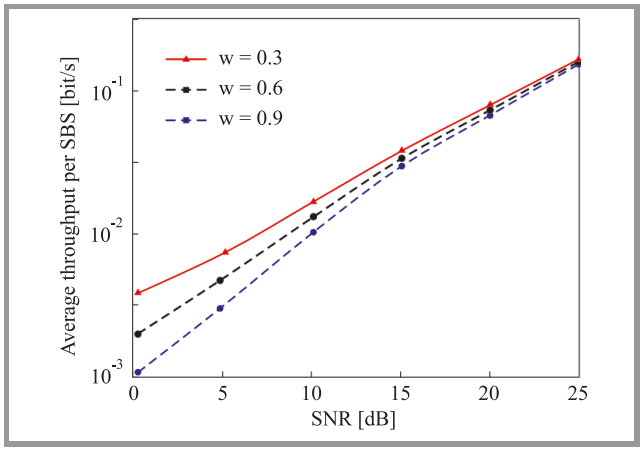


Fig. 4. Maximum achieved throughput for various value of the utility contribution weighting factor w.

Figure 4 shows the results of the average transmissions of SBSs coalition in terms of achieved throughput per SB (the player's achieved throughput in the coalition divided by the total available bandwidth) for various values of the utility-component weighting factor w ($w \in [0.3, 0.6, 0.9]$), and for assumed SNR value. Here, it can be seen that the transmitted power exceeds the power limit for small SNR value and for $w = 0.3$. Thus, a higher throughput is achieved due to the lack of noise. If $w = 0.9$, the opposite results are obtained because the SB is forced to be more power efficient. By assuming $w = 0.6$ in the game, the optimal curve of the achieved throughput and the bandwidth can be obtained.

6. Conclusion

In this paper, a new scheme for the coalition game among the SBSs in CR networks was proposed. The main advantage of the presented solution lies in the coalition formation of the SBS with incomplete information and the conveyed knowledge of the spectrum for all the SUs in the system. The proposed algorithm allows to ensure cooperation among the SBSs. The payoff of every coalition of the SBSs allows to decide to join or leave the coalition. Finally, using showed algorithm, the SBSs can reach a Bayesian equilibrium. The results of the simulation also confirm that proposed algorithm improved the average payoff of the SBSs coalition with incomplete information up to 140% in comparison to the non-cooperative case.
The future work should consider the confrontation of the proposed algorithm with that of an centralized solution.

References

[1] J. Mitola, "Cognitive Radio: an Integrated Agent Architecture for Software Defined Radio", Ph. D. Dissertation, Royal Institute of Technology, 2000.

[2] "Spectrum Policy Task Force", Federal Communications Commission, Tech. rep., Nov. 2002.

[3] "Facilitating Opportunities for Flexible, Efficient, and Reliable Spectrum Use Employing Cognitive Radio Technologies", Federal Communications Commission, *Notice of Proposed Rule Making and Order*, FCC 03-322, 2003.

[4] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "Next generation/dynamic spectru access/cognitive radio wireless networks: a survey, computer networks", vol. 50, no. 13, pp. 2127–2159, 2006.

[5] A. Ghasemi and E. S. Sousa, "Collaborative Spectrum Sensing for Opportunistic Access in Fading Environments", in *Proc. 1st IEEE Symp. New Frontiers Dynam. Spec. Access Netw. IEEE DySPAN 2005*, Baltimore, MA, USA, 2005.

[6] F. F. Digham, M. S. Alouini, and M. K. Simon, "On the energy detection of unknown signals over fading channels", in *Proc. Int. Conf. Commun.*, Alaska, USA, 2003, pp. 3575–3579.

[7] W. Zhang and K. B. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks", *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, 2008.

[8] P. Houz, S. B. Jemaa, and P. Cordier, "Common pilot channel for network selection", in *Proc. 63rd IEEE Veh. Technol. Conf VTC 2006*, Melbourne, Australia, 2006.

[9] M. Filo, A. Hossain, A. R. Biswas, and R. Piesiewicz, "Cognitive pilot channel: enabler for radio systems coexistence", in *Proc. 2nd Int. Worksh. Cogn. Radio Adv. Spectrum Manag.*, Aalborg, Denmark, May 2009.

[10] O. Sallent, J. Perez-Romero, R. Agusti, and P. Cordier, "Cognitive pilot channel enabling spectrum awareness", in *Proc. IEEE Int. Conf. Commun. Worksh. ICC 2009*, Dresden, Germany, 2009.

[11] W. Saad, Z. Han, T. Basr, A. Hjorungnes, Ju Bin Song, "Hedonic coalition formation games for secondary base station cooperation in cognitive radio networks", in *Proc. IEEE Wirel. Commun. Netw. Conf. WCNC 2010*, Sydney, Australia, 2010, pp. 1–6.

[12] W. Saad, Z. Han, M. Debbah, and A. Hjorungnes, "Coalitional games for distributed collaborative spectrum sensing in cognitive radio networks, in *Proc. 28th Conf. Comp. Commun. IEEE IFOCOM 2009*, Rio de Janeiro, Brazil, 2009, pp. 2114–2122.

[13] S. M. Perlaza, S. Lasaulce, M. Debbah, and J.- M. Chaufray, "Game Theory for Dynamic Spectrum Sharing", in *Cognitive Radio Networks: Architecture, Protocols and Standards*, Y. Zhang, J. Zheng, AND H. Chen, Eds., Taylor and Francis Group, Auerbach Publications, Boca Raton, FL 2010.

[14] S. Mathur and L. Sankaranarayanan, "Coalitional games in gaussian interference interference channels", in *Proc. IEEE Int. Symp. Inf. Theory ISIT 2006*, Seattle, WA, USA, 2006, pp. 2210–2214.

[15] G. Scutari, D. P. Palomar, and S. Barbarossa, "Competitive Design of Multiuser MIMO Systems Based on Game Theory: A Unified View", *IEEE J. Selec. Areas in Commun.*, vol. 26, no. 7, pp. 1089–115, 2008.

[16] R. B. Myerson, *Game Theory, Analysis of Conflicy*. Cambridge, USA: Harvard University Press, 1991.

[17] D. Fudenberg, J. Tirole, *Game Theory*. Cambridge, USA: MIT Press, 1991.

[18] J. C. Harsanyi, R. Selten, "A generalized nash solution for two-person bargaining games with incomplete information", *Manag. Sci.*, vol. 18, pp. 80–106, 1972.

[19] R. B. Myerson, "Cooperative games with incomplete information", *Int. J. Game Theory*, vol. 13, pp. 69–86, 1994.

[20] J. B. Rosen, "Existence and Uniquess of Equilibrium Points of Equilibrium Points for Concave N-person Games", *Econometrica*, vol. 33, pp. 520–534, 1965.

[21] D. Niyato, E. Hossein, and Z. Han, *Dynamic Spectrum Access and Management in Cognitive Radio Networks.* Cambridge, UK: Cambridge University Press, 2009.

[22] J. Proakis, *Digital Communications*. 4th edition. McGraw-Hill, 2000.

[23] S. Vassaki, A. Panagopoulos, and Ph. Constantinou, "Game-theoretic approach of fair bandwidth allocation in DVB-RCS networks", in *Proc. Int. Worksh. Satel. Space Commun. IWSSC 2009*, Siena-Tuscany, Italy, 2009, pp. 321–325, 2009.

Jerzy Martyna received M.Sc. degree in Telecommunications and Ph.D. degree in Information Engineering both from the AGH University of Science and Technology, Cracow, Poland, in 1976, and 1985, respectively. Since 1976, he has been with the Faculty of Mathematics and Computer Science at the Jagiellonian University in Krakow. He spent some times at the TU Dortmund University as a research fellow of Alexander von Humboldt-Stiftung and DAAD. His general research interests cover computer networks and distributed systems, mobile and wireless communications systems with emphasis on queueing systems, real-time systems, modeling and performance evaluation of computer systems and artificial intelligence systems first of all in telecommunications. His current research are focused on wireless ad hoc and sensor networks, cognitive radio networks, opportunistic networks and multicarrier (orthogonal frequency-division multiplexing) systems. He is the author of more than 160 papers, which have been presented at national and international conferences. He has published 2 handbooks in the area of computer science and computer networks.

E-mail: martyna@softlab.ii.uj.edu.pl
Institute of Computer Science
Faculty of Mathematics and Computer Science
Jagiellonian University
Prof. S. Łojasiewicza st 6
30-348 Krakow, Poland

Paper

Quasi-Offline Fair Scheduling in Third Generation Wireless Data Networks

Jerzy Martyna

Institute of Computer Science, Faculty of Mathematics and Computer Science, Jagiellonian University, Krakow, Poland

Abstract—In 3G wireless data networks, network operators would like to balance system throughput while serving users fairly. This is achieved through the use of fair scheduling. However, this approach provides non-Pareto optimal bandwidth allocation when considering a network as a whole. In this paper an optimal offline algorithm that is based on the decomposition result for a double stochastic matrix by Birkhoff and von Neumann is proposed. A utility max-min fairness is suggested for the derivation of a double stochastic matrix. Using a numerical experiment, new approach improves the fairness objective and is close to the optimal solution.

Keywords—Birkhoff-von Neumann, max-min fairness, statistical optimization, wireless scheduling.

1. Introduction

Next generation wireless communication is based on a system of wireless mobile services that are transportable across different network backbones. Third generation networks such as the CDMA [1], and the Universal Mobile Telecommunications System (UMTS) [2] standardised by the European Telecommunications Standards Institute (ETSI) promise heterogeneous services to users that may be moved across various regions and networks. Recent 3G releases, often denoted 3.5G and 3.75G, also provide mobile broadband access of several Mbit/s to smartphones and mobile modems in laptops.

The 3G standard, called the 3G1X Evolution or High Data Rate (HDR) was designed for bursty packet data applications. It provides a peak downlink data rate of 2 Mbps and an average downlink data rate of 600 kbit/s within one 1.25 MHz CDMA carrier. HDR is commercially available, and HDR downlinks have a much higher peak data rate (2.4 Mbit/s) than others. Users share the HDR downlink using time multiplying with time slots of 1.67 ms each. Data frames can be transmitted to a specified user at any moment in time, and the data rate is determined by the user's channel condition. Users monitor pilot bursts in the downlink channel to estimate channel conditions in terms of Signal to Noise Ratio (SNR). Then, the SNR is mapped into a supported data rate. The data rate request channel information is transmitted using feedback to the base station. The duration of transmission to each user is determined by the downlink scheduling algorithm.

Several wireless scheduling have been proposed. The scheduling algorithm to satisfy the so-called proportional fairness was proposed by Jalali *et al.* [3]. The scheduling algorithm given by Borst *et al.* [4] provides dynamic control for fair allocation of HDRs. The 3G standard uses a scheduling algorithm [5]. Unfortunately, these algorithms give relative fairness to users rather than guarantee the required QoS performance.

Currently, some scheduling techniques to be used at the Medium Access Control (MAC) layer for high data rate Wireless Personal Area Networks (WPANs) were presented by Fantacci and Tarchi [6]. An efficient heuristic scheduling algorithm for MPEG-4 traffic in high data rate WPANs has been presented by Yang *et al.* [7]. However, these solutions have problems, such as computational complexity and rate granularity limitation.

The main objective of this paper is to introduce a new wireless scheduling algorithm that provides predetermined user throughputs. In addition, the paper will show that proposed scheduling algorithm is quasi-offline. It allows to remove the complexity of on-line scheduling. The main idea of presented scheduling the connection patterns is the Birkhoff decomposition [8] and von Neumann methodology [9].

In this paper, a statistical approach for Birkhoff-von Neumann methodology is used in which traffic demands are captured as statistical traffic distribution. For the derivation of a double stochastic matrix is proposed a utility max-min fair algorithm. In opposition to the von Neumann algorithm [9], the cumulative distribution functions that correspond to the given statistical profile is presented.

The remainder of the paper is organized as follows: In Section 2, a Birkhoff-von Neumann decomposition is presented which offers a quasi-offline scheduling strategy. Section 3 provides proposed wireless scheduling algorithm. In Section 4, some numerical experiments which were performed to examine the properties of the proposed algorithm are described. Some concluding remarks are given in Section 5.

2. Preliminaries

2.1. Birkhoff-von Neumann Decomposition

To explain the idea of Birkhoff-von Neumann decomposition, let $\overline{r} = (r_{i,j})$ be the rate matrix with $r_{i,j}$ being the rate allocated to the traffic from input i to output j for $N \times N$

permutation matrix. The traffic is admissible if and only if the following inequalities are satisfied, namely

$$\sum_{i=1} r_{i,j} \leq 1, \quad j = 1,2,\ldots,N \tag{1}$$

and

$$\sum_{j=1}^{N} r_{i,j} \leq 1, \quad i = 1,2,\ldots,N. \tag{2}$$

There exists a set of positive number ϕ_k and permutation matrix P_k, $k = 1,2,\ldots,K$ for some $K \leq N^2 - 2N + 2$ that satisfies

$$\overline{r} \leq \sum_{k=1}^{K} \phi_k p_k \tag{3}$$

and

$$\sum_{k=1}^{K} \phi_k = 1. \tag{4}$$

Generally, the Birkhoff-von Neumann methodology is performed offline. Unfortunately, the computational complexity of the decomposition is $O(N^{4.5})$.

Several method have been used to decreasing of the computational complexity. Among others, the Weighted Fair Queueing (WFQ) scheme as on-line algorithm has been proposed by Demers *et al.* [10]. Thus, the complexity of the on-line scheduling algorithm is $O(log\ N)$ as one needs to sort the $O(N^2)$ virtual finishing times in the WFQ-like algorithm.

2.2. Max-min Fairness

Maximizing aggregate utility is able to approach max-min fairness, if the utility function has a particular form. Max-min fairness an important requirement for wireless networks, such as multi-hop WANETs, MANETs, etc. [11]. To explain the idea of max-min fairness, let x be a vector, $x \in R^n$. Given a non-empty set $S \subseteq R^n$, a fairness concept supplies a way of designating some vector as the "best" one in S. A vector $x \in S$ is max-min fair if one cannot increase one of its components without decreasing another of its components that is already smaller or equal, while remaining in S.

In the following, an algorithm to obtain a utility max-min fairness is presented. In contrast to approaches in which the utility is defined with respect to service quality parameters [12], [13], this utility max-min fair algorithm is used to the Birkhoff-von Neumann decomposition problem under statistical traffic distribution.

In the proposed algorithm it was assumed that each row and column as being fixed or free. Initially, all rows and columns are free. Let ρ_i be the available capacity on row i and η_j be the available capacity on column j. Both values are initially equal to 1. The algorithm is repeated in the loop and at each iteration of the algorithm are consider only free columns and rows. Algorithm 1 shows the pseudocode of the proposed algorithm to obtain a utility max-min fairness.

Algorithm 1: A utility max-min fairness

```
Input: Flow rates specified as N × N matrix
while exists free column or row do
    for each free flow i, j on row i temporarily do
        allocate k_ij such that
        Σ_{j∈T} k_ij = ρ_i,  u_ij(k_ij) = θ  ∀j ∈ T
        where T is set of all columns
    end
    for each free flow i, j on column j temporarily do
        allocate k_ij such that
        Σ_{j∈S} k_ij = η_j,  u_ij(k_ij) = θ  ∀i ∈ S
        where S is set of all rows
    end
    find the minimum maximum common utiliy u_ij that
    could be achieved
    if this minimum θ corresponds to row i
    then remove row i from S and fix the rate
    allocations:
            S = S − {i}, λ_ij = k_ij;
    end if;
    find the minimum maximum common utiliy u_ij that
    could be achieved
    if this minimum θ corresponds to column j
    then remove column j from T and fix the rate
            allocations: T = T − {j}, λ_ij = k_ij;
    end if;
end
update the corresponding row and column capacities
that are affected by the fixing of the λ_ij;
```

3. Statistical Approach for Birkhoff-von Neumann Decomposition

In this section, a statistical approach for Birkhoff-von Neumann decomposition is presented. In this approach traffic requirements are described as statistical traffic distributions rather a vector fixed average rates.

The flow rates is specified as $N \times N$ matrix of probability density functions:

$$F = (f_0(x)) \tag{5}$$

where $f_0(x)$ is the probability distribution of traffic requirements for flow (i,j) and x be the rate allocation.

That long-term average throughput for $n, n = 1,\ldots,N$, is given by

$$T_n = \frac{\mu_n}{N} + \frac{\sigma_n}{N} G_N \tag{6}$$

and

$$G_N = N \int_0^1 u^{N-1} Q^{-1}(1-u)du, \tag{7}$$

where μ_n is the mean, σ_n is the variance $n = 1,2,\ldots,N$, $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t/2} dt$.

Note that the Cumulative Distribution Function (CDF) of a Gaussian random variable depends only on the mean and the variance, because the average traffic flow does not depend on other traffic flows distribution.
For each $f_{i,j}(x)$ corresponding to the probability distribution of traffic requirements for flow (i, j), the CDF function is defined that describes the probability distribution of a random variable X that represents the actual traffic requirement. Thus, the CDF function of X is given by

$$\phi_{i,j}(x) = Pr[X \leq x], \tag{8}$$

where $\phi_{i,j}$ is the probability that the rate allocation x is sufficient to cover the actual traffic requirement.
To maximize the probability $\phi_{i,j}$, the CDF functions as utility functions can be used and derive the final rate allocation matrix.

4. Numerical Example

The traffic distribution is modeled as a Gaussian distribution. Each probability density function $f_{i,j}(x)$ can be defined with a given mean $\mu_{i,j}$ and a standard deviation $\sigma_{i,j}$. Consider 3×3 rate matrix with the following means and standard deviations:

$$\begin{pmatrix} \mu_{1,1} & \mu_{1,2} \\ \mu_{2,1} & \mu_{2,2} \end{pmatrix} = \begin{pmatrix} 0.4 & 0.4 \\ 0.4 & 0.4 \end{pmatrix}$$
$$\begin{pmatrix} \sigma_{1,1} & \sigma_{1,2} \\ \sigma_{2,2} & \sigma_{2,2} \end{pmatrix} = \begin{pmatrix} 0.05 & 0.1 \\ 0.1 & 0.05 \end{pmatrix}. \tag{9}$$

The decomposition algorithm originally proposed by von Neumann [9] is used to a double stochastic rate matrix Λ with the average traffic requirement matrix $\{\mu_{i,j}\}$ as the starting point. The von Neumann algorithm aims to increase the rate to make all row and column sums equal to 1, namely

$$\Lambda = \begin{pmatrix} 0.6 & 0.1 \\ 0.1 & 0.6 \end{pmatrix}. \tag{10}$$

The actual traffic requirement would be

$$\begin{pmatrix} 99.45\% & 50\% \\ 50\% & 99.45\% \end{pmatrix}. \tag{11}$$

By using the utility max-min fair algorithm to construct the doubly stochastic rate matrix $\Lambda = (\lambda_{ij})$ with the given Gaussian distribution function as utility functions, the following matrix can be obtained

$$\Lambda = \begin{pmatrix} 0.5 & 0.2 \\ 0.2 & 0.5 \end{pmatrix}. \tag{12}$$

Obtained result is better than the 50% probability that would be achieved by the von Neumann algorithm, namely

$$\Lambda = \begin{pmatrix} 99.64\% & 99.64\% \\ 99.64\% & 99.64\% \end{pmatrix}. \tag{13}$$

Summing up, the expected traffic can be modeled by the probability distribution for flow rates of HDR system. The utility max-min fair allocation algorithm can be used to construct the double stochastic rate matrix $\Lambda = (\lambda_{ij})$ with the CDF functions as the utility function.

5. Conclusion

It has been provided an algorithm to packet scheduling in wireless networks and has been formulated the problem in which traffic demands are captured as statistical traffic distribution. A utility max-min fair algorithm was used for the derivation of a double-stochastic matrix. This quasi-offline scheduling is attractive as it also largely removes the complexity of online wireless packet scheduling. Finally, the numerical results demonstrate that proposed solution achieves the required system performance.

References

[1] "Mobile Station-base Station Compatibility Standard for Dual-mode Wideband Spread Spectrum Cellular Systems", EIA/TIA Interim Standard, Washington: Telecommunication Industry Association, 1999.

[2] "UMTS, Release 11. 3G Partnership Project", 2013 [Online]. Available: http://www.3gpp.org/Release-11

[3] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of CDMA-HDR a high efficiency – high data rate personal communication wireless system", in *Proc. IEEE Veh. Technol. Conf. Proc. VTC 2000 (Spring)*, Tokyo, Japan, 2000, pp. 1854–1858.

[4] S. Borst and P. Whiting, "Dynamic rate control algorithms for HDR throughput optimization", in *Proc. IEEE INFOCOM 2001*, Anchorage, Alaska, USA, 2001, pp. 976–985.

[5] 1xEV: 1x EVolution IS-856 TIA/EIA Standard – Airlink Overview, QUALCOMM Inc., White Paper, Nov. 2001.

[6] R. Fantacci and D. Tarchi, "Efficient scheduling techniques for high data-rate wireless personal area networks", *Int. J. Sensor Netw.*, vol. 2, no. 1–2, pp. 128–134, 2007.

[7] G. Yang, H. Hu, and L. Rong, Y. Du, "Performance of an efficient heuristic scheduling algorithm for MPEG-4 traffic in high data rate wireless personal area network", *IET Commun.*, vol. 5, no. 8, pp. 1090–1095, 2011.

[8] G. Birkhoff, "Tres observations sobre el algebra lineal", *Univ. Nac. Tucumán Revue*. Ser. A 5 pp. 147–151, 1946.

[9] J. von Neumann, *A Certain Zero-sum Two-person Game Equivalent to the Optimal Assignment Problem, Contributions to the Theory of Games*, vol. 2. Princeton, NJ: Princeton University Press, 1953, pp. 5–12.

[10] A. Demers, S. Keshav, and S. Shenkar, "Analysis and simulation of a fair queueing algorithm", in *Proc. ACM Symp. Commun. Arch. Prot. SIGCOMM '89*, Austin, TX, USA, pp. 1–12.

[11] L. Tassiulas and S. Sarkar, "Maxmin fair scheduling in wireless networks", in *Proc. IEEE INFOCOM 2002*, New York, NY, USA, 2002, pp. 763–772.

[12] Z. Cao and E. W. Zegara, "Utility max-min an application-oriented bandwidth allocation scheme", in *Proc. IEEE INFOCOM 1999*, New York, NY, USA, 1999, vol. 2, pp. 793–801.

[13] S. Sarkar and L. Tassiulas, "Bandwidth layers in multicast networks", in *Proc. IEEE INFOCOM 2000*, Tel-Aviv, Israel, 2000, vol. 3, pp. 1491–1500.

[14] R. Maheshwari *et al.*, "Adaptive channelization for high data rate wireless networks", Tech. Rep, Stony Brook University, Stony Brook, New York, USA, 2009.

Jerzy Martyna – for biography, see this issue, p. 111.

Paper

Performance Tests of Xen-based Node for Future Internet IIP Initiative

Grzegorz Rzym and Krzysztof Wajda

Department of Telecommunications, AGH University of Science and Technology, Kraków, Poland

Abstract—In this paper the authors intend to report results for performance evaluation of Xen-based node providing both transport and computational functionalities. This node design is proposed as the implementation platform developed for the Polish Initiative of Future Internet called System IIP. In particular, we search for mutual dependence among transport and computational performance parameters of the node. Our investigations show that there is significant dependence among performance indices, such as virtual link bandwidth and node's processing power, strongly depending on frame size. The tests and measurements were done according to fundamental methodology designed for network devices, described in RFC 2544. The goal of those investigations is to design a provisioning module allocating both transport and computational resources.

Keywords—benchmarking, Future Internet, measurements, virtualization, Xen.

1. Introduction

We are witnessing growing efforts in recent years aiming at designing a new architecture of the Internet, not based exlusively on classical IP protocols family. The general ideal is to design new networking environment being strongly oriented towards emerging applications and networking paradigms, such as Internet of Things (IoT) and Content Aware Networking (CAN). Besides mentioned above goals are also modern mechanisms such as virtualization and parallelization of networks, complemented by new (or revisited) approaches to data, control and management planes.

Most promising and pioneering initiatives towards Next Generation Internet comprise Japanese Akari [1], US-based GENI [2] and European project 4WARD [3]. Involvement of EU-promoted Future Internet Assembly (FIA), supported by US-based National Scence Foundation (NFS), also ETSI, and ITU-T has given a significant and fruitful impact on research and experimental works in this area.

This paper is organized as follows: Section 2 presents Future Internet initiative, Section 3 outlines the important aspects of Xen relative to virtual forwarding, bridging and routing, in Section 4 implemented node model is invited, Section 5 presents the experimental setup that authors use to perform our evaluation, and Section 6 presents results. The paper is concluded in Section 7.

2. Related Work

Evaluating of transport features and capabilities of Xen nodes can be done experimentally due to lack of theoretical models.

Initial tests of Xen 1.0 presented in [4] have shown that performance of Xen platform is practically equivalent to the performance of native Linux system. The developers from University of Cambridge have shown that Xen can be widely used to proceed transmitted data. Running simultaneously 128 virtual machines caused only 7.5% loss of total throughput compared to Linux with 5 ms maximum scheduling "slice". When scheduling "slice" were set to 50 ms (the default value used by ESX Server) the throughput curve was very close to the native Linux. Mean response time in such configured system was 5.4 ms.

N. Egi *et al.* in [5] presented the forwarding performance of driver domain (dom0) and virtual machines (domU). They configured Xen network with tree types of mechanisms to transmit packets between physical network devices and virtual interfaces and inside VMs: bridging, routing and hybrid method (combination of routing with bridging). Observations included in [5] have shown that the performance of hybrid connection inside Xen node is 30% lower than using only bridging or routing. Since CPU is reported as the bottleneck of PC-based virtual router the authors suggested that domU should only host the control path and the forwarding path should be located and served in dom0. This solution leads to avoidance of context switching overheads. The authors also compared network performance for driver domain and native Linux system. Their results have shown very close performance in forwarding of short frames.

In [6] three techniques for network optimization in the Xen were proposed:

– redefinition of the virtual network interfaces which allows guest domains to incorporate high-level network offload features (i.e., TCP segmentation offloading, scatter-gather I/O and TCP checksum offloading) in physical NICs,

– new data path between dom0 and domU avoids data remapping operations and proposal of usage of superpages and global page memory mapping.

All these techniques provide 35% growth of data transmission performance in driver domain and 18% growth in guest's domains.

Pujolle *et al*. presented in [7] evaluation of Xen transmission performance. The authors believed that forwarding should be accomplished inside virtual machines (virtual routers). Since the forwarding performance of dom0 is better than domU they propose several optimization methods. They provide a mechanism to guarantee the required system throughput and latency by:

– optimized CPU allocation, prioritizing packets inside dom0 before switching them to the target virtual machine,

– appropriate configuration of the Xen Credit scheduler and finally dedicated virtual routers to carry flows with different priorities.

This allows to forward quality-sensitive flows with acceptable delay and throughput.

Further experimental results applicable to IIP project are recently investigated and analyzed in [8]. Adamczyk and Chydziński studied isolation between virtual machines across virtual network adapters. Their investigations have shown lack of proper performance isolation among virtual machines. Xen Netback driver that is responsible for the scheduling work uses simple round-robin algorithm. In addition several network adapters can be mapped to the same Netback kernel thread. The authors propose additional two parameters (priority and min rate) for every virtual network adapter to improve the network performance isolation.

Recent works focus only on throughput and forwarding performance of PC-based virtual routers under different configurations. The mutual dependence among transport and node performance parameters have not been investigated in any work yet. The obtained results can be used to design a provisioning module combining both transport and processing resources.

3. Future Internet Initiative

Designing and experimenting with novel architecture of Future Internet are ongoing as a target of a Polish initiative called System IIP. The structure of the System IIP is based on implementation of four levels of the architecture (bottom-up description): physical infrastructure (L1), virtualization (L2), Parallel Internets (PIs, L3) and virtual networks (L4).

The Future Internet Engineering project assumes existence of three Parallel Internets (PIs): IPv6 Quality of Service (IPv6 QoS), Content Aware Network (CAN) and Data Stream Switching (DSS). More details can be found in [9] and [10].

Xen is one of three system virtualization environment used in IIP project. Others are NetFPGA and EZchip NP-3 [11]. The advantage of Xen is that it can be run on each computer that supports virtualization, it is an open source and fully programmable environment.

4. Xen as Virtualization Platform with Bridging/Routing Features

Xen [4] is an open source virtualization platform. It consist of the *Hypervisor* and *Domains*.

Hypervisor is an abstraction software layer running directly above the hardware. This layer is an interface between operations running on the hardware (CPU, I/O devices, etc.) and the guest's operating system. Its main tasks are CPU scheduling, memory management and forwarding of interrupts (hypercalls in Xen).

Xen runs virtual machines in separate environments known as domains. There are two kinds of domains: privileged Domain0 and unprivileged DomainU. The former, the driver Domain0 is responsible for the communication between guest domains DomainU and the devices. It implements all device drivers and has direct connection through *event channel* with these devices. DomainU is an unprivileged domain started by Domain0 and runs guest operating systems.

Xen implements two types of drivers: back-end and front-end. Back-end drivers are used by Domain0 to communicate from one side with hardware, from the other – with front-end implementation in DomainU. This concept of distributed drivers are used by block and networks devices.

There are three types of mechanisms to transmit frames between physical network devices and virtual interfaces: bridging, routing and additionally Network Address Translation (NAT) mechanism.

4.1. Bridging in Xen

Bridging is the default network configuration in Xen. It uses standard Linux bridging mechanism for packets forwarding. All guest domains can share the same Network Interface Card (NIC). Back-end devices are connected to the physical NIC and are represented as a pethX devices in the system.

When packet arrives to NIC Hypervisor it is notified by physical interrupt. Then Hypervisor forwards that information to the Domain0 through the event channel and copies the packet to its address space. When guest domain receives its scheduled time slot it sees the notification and looks for the packet in the common memory page. Next, the guest domain can start processing the packet. The return route is similar. We can observe two way communication: the communication between the Hypervisior and the driver domain and the communication between the driver domain and the guest domain.

4.2. Routing in Xen

In a routed network configuration in Xen a point-to-point link is created between the driver domain Domain0 and each guest virtual network interface. Is it necessary for each guest domain have an IP address. Packets forwarded between physical interfaces and other virtual network cards are routed like in native Linux.
The use of NAT allows us to create many private Local Areas Networks (LANs) inside one physical computer. System configuration is the same as in the case of network routing. In order to address translation it is necessary to use standard Linux tools such as iptables (also used to traffic filtering).

5. The Node Model

The IIP node consists of following elements: frame classifier, scheduler and three virtual machines corresponding to three various Parallel Internets presented in the Fig. 1.

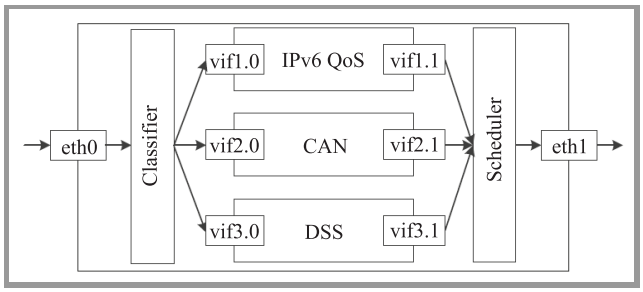


Fig. 1. The node model.

5.1. Frames Classifier

The IIP System Classifier is a mechanism that allows for demultiplexing of incoming traffic to the one of three virtual machines. Therefore, the IIP System defines a new Parallel Internet Header (PIH). It has 8 bits and is located next to the Ethernet frame header.
The Classifier is located in the main virtual machine – Domain0. This enables the PDU classification by using modified *ebtables* rules [12].

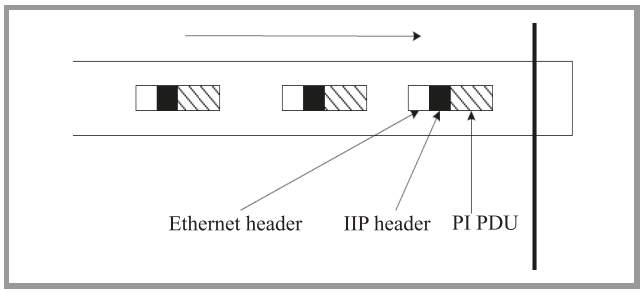


Fig. 2. IIP incoming data stream.

In the Fig. 2 an incoming data stream with distinguished order of the headers can be observed [13].

5.2. Virtual Node

As a virtual router we use a separated virtual machines. As guest operating system we run Gentoo Linux with software specified to perform different tasks for each PI. Such solution significantly simplifies the implementation and allows for future expansion without major modifications of the existing system.

5.3. Scheduler

The main goal of the scheduling mechanism is the division of physical link resources (its capacity C) between each Parallel Internets [13]. In the case of three previously mentioned PIs the following condition 1 should be satisfied:

$$C_{IPv6QoS} + C_{CAN} + C_{DSS} \leqslant C \tag{1}$$

In the first level the scheduling mechanism is based on a cycle. A duration o each phase of the cycle should be set in the provisioning (i.e. dimensioning) phase. The cycle is composed of fixed number of phases. It is worth mentioning that as a result of using Scheduler the link data rate is lost – it can not be used by other virtual machines (*non-work conserving* algorithm). Scheduling mechanism for the second level is different and suitable for each PI. In the Fig. 3 the two-level scheduling mechanism is shown [13].
In the case of the Xen platform the *Netback* driver is responsible for scheduling process of outgoing traffic. Used standard FIFO queue and *Credit Scheduler* allows us to specify the maximum outgoing throughput.

6. Experimental Settings

Evaluating of Xen-based node was performed by using HP ProLiant DL360 G6 server with 2 Intel Xeon X5660 2.80 GHz processors, 6×4 GB DDR3 1333 MHz RAM memory, $4 \times$ Intel 82571EB NIC and 500 GB HP SCSI hard drive.

6.1. Test Topology

General scheme of network performance testing Xen-based node is presented in the Fig. 4.
As a Traffic Generator and Traffic Sink a Spirent TestCenter STC-2000 platform was used. This hybrid (hardware and software) device provides wide variety of network test configurations for functions ranging from 2nd to 7th OSI-ISO layers. It also offers high performance traffic generation and measurements.
Network tests were performed according to recommendations defined in RFC2544 [14] and RFC5180 [15]. Generated bandwidth was in the range from 10 Mbit/s to 330 Mbit/s for each StreamBlock – three simultaneous streams from three Parallel Internets (that gives altogether 990 Mbit/s of maximum throughput). Tests were

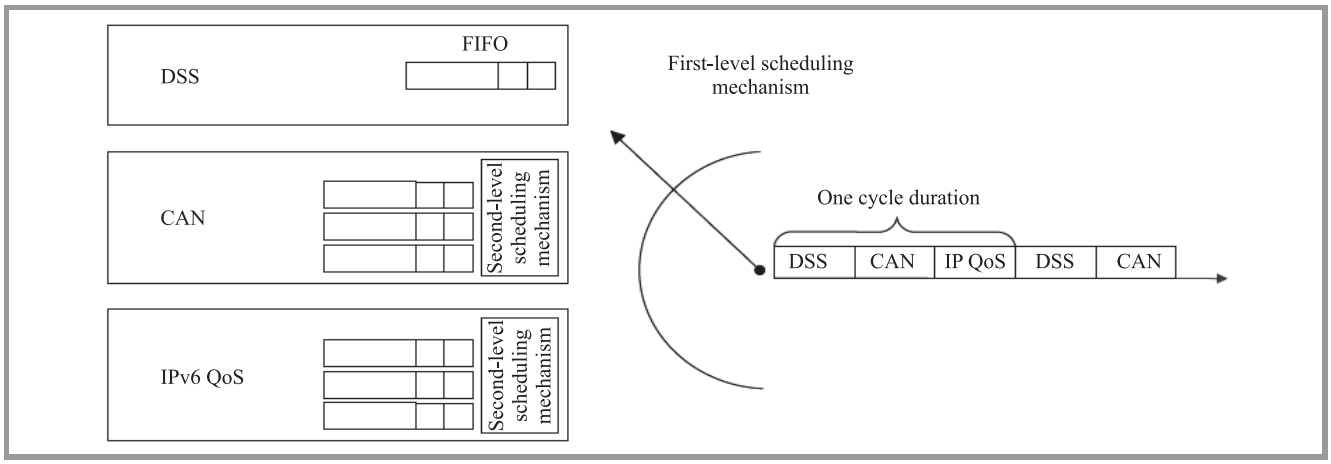


Fig. 3. Scheduling mechanism.

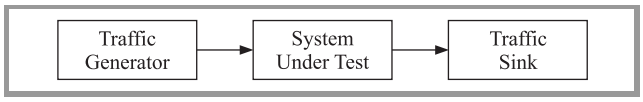


Fig. 4. Network topology used during tests.

run with the following frame sizes: 79, 128, 256, 512, 1024, 1280, and 1518 bytes since this parameter has significant impact on the performance. One measurement run took 110 seconds.

6.2. Test Description

Since we did not find any recommendation for virtual machines performance measurement we proposed the following algorithm:

1. Start the server.
2. Run simultaneously 3 IIP virtual machines.
3. Wait 5 minutes for system stabilization.
4. Run test on Spirent TestCenter platform according to scenarios proposed in recommendations.
5. Save results.
6. Restart server and go back to the first point.

First tests to evaluate the native performance of our system is run. These results will help in evaluation of the overhead imposed by the use of virtualization and IIP System implementation.
Next, we run simultaneously 3 virtual machines providing data stream transmission within the different Parallel Internets. Each guest domain had allocated (pinned) 2 processors, 4 GB RAM memory, 5 GB virtual hard drive as an image file. The driver domain had pinned 8 cores. When network performance was measured we run a script that monitors the usage of computational power of processors. Each test were performed with 100 μs phase size for each virtual machine.

7. Results

In this section the results of experiments carried in two settings (conditions) are presented. For the first test set there were carried experiments for fixed frame sizes, according to RFC 2544 (i.e., each run for fixed and different size of frame: 79, 128, 256, 512, 1024, 1280 and 1518 bytes). Next, in second test set we make measurements with random frame size. Results are presented in two versions: X axis is expressed either in Mbit/s (suitable for general purposes of dimensioning) or kpps (kilo packets per second), reflecting transport efficiency of the node.

7.1. Fixed Frame Size

We can observe a strong relation between maximum throughput and frame size. For a given throughput maximum number of small frames (i.e., 128 bytes) that can be processed by the node is much lower than for the larger frames (e.g., 1024 bytes) because number of headers that the node must analyze in the latter case is lower. In the Fig. 5a node throughput for each tested frame size is presented. As we can observe in the Fig. 5b maximum number of handled packets for small frames is variable.
As can be observed in the Fig. 6, the average frame latency is strongly correlated with maximum throughput. The latency is measured by Spirent device as the difference in time between the instant of receiving frame and sending the frame. Until the frames are not dropped, the mean latency remains low (about 150 μs). Above the maximum throughput the average frame delay increases significantly and then remains at a constant level – for frames from 79 to 512 bytes of size the average latency above the maximum throughput tends to 20 ms, for bigger frames latency can be even higher.
Our investigations show that for the required bandwidth the processing power utilization for each virtual machine is larger when handling smaller frames, i.e., for 150 Mbit/s about 50% for 128 bytes and 16% for 1024 bytes frames. In the Figs. 7–8 the impact of traffic load on the utilization

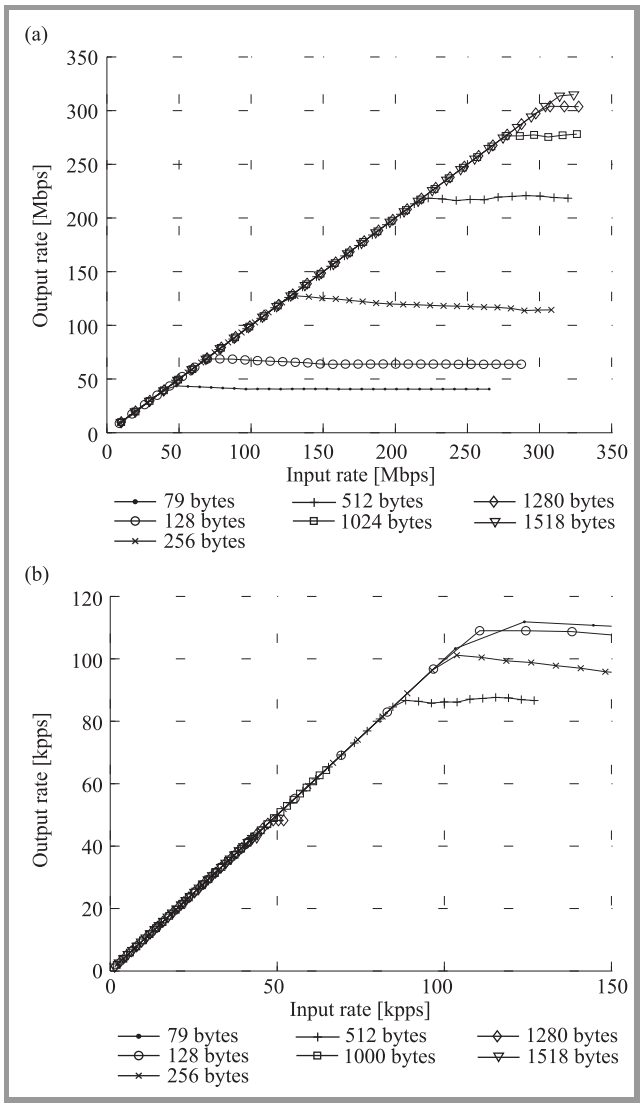


Fig. 5. Node throughput for a specified frame size.

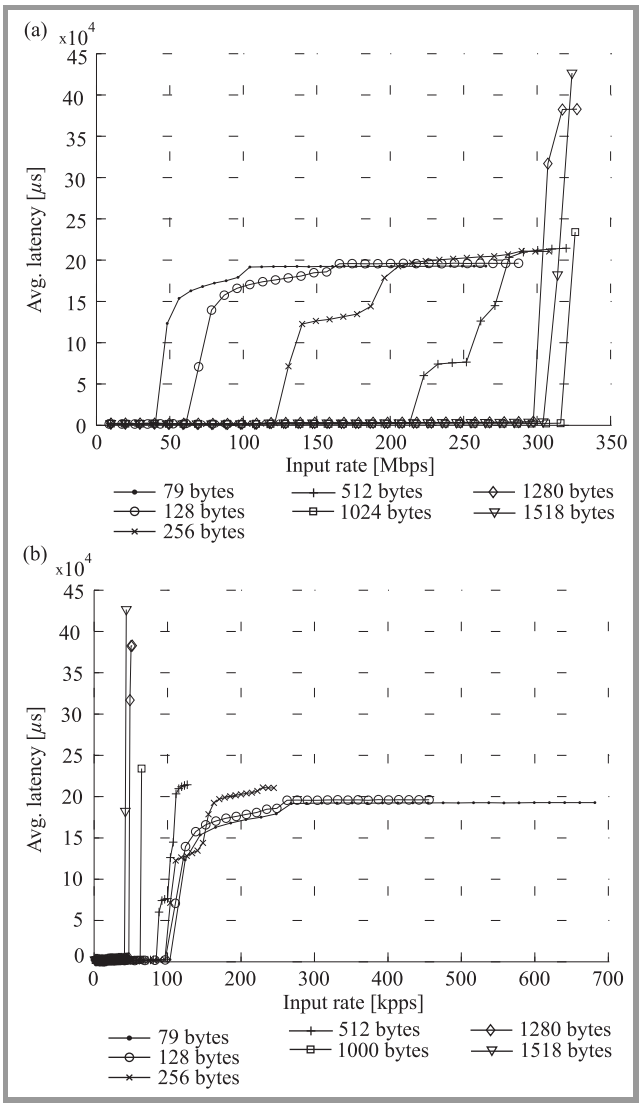


Fig. 6. Node latency for a specified frame size.

of processing power is shown, allocated to each virtual machines for 128 and 1024 bytes frame size, respectively. Presented results show large differences in the use of node processing power depending on the frame size. For bigger frames (i.e., 1024 bytes) system load increases almost linearly with the size of incoming traffic rate in the whole range. For smaller frames system load is linear until frames are lost, then system utilization is maintained at a constant level.

As we expected, the processors assigned to the CAN virtual machine are most utilized. This result is caused by necessary analysis of data contained in the PDU frame by machine kernel after removing of PIH header according to the specifications presented in [13]. Virtual machine serving traffic from DSS Parallel Internet uses the least amount of the server computational power.

It is also visible uniformity of load for each processor assigned to virtual machines. First processors are used much harder than the other in each virtual machine. This difference increases with the increase of the bandwidth. Also

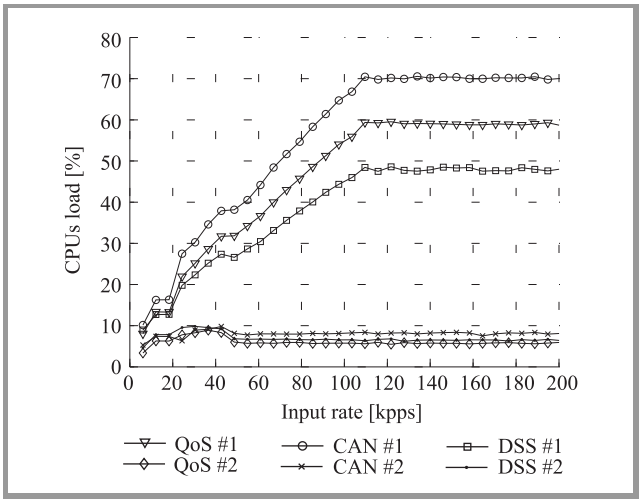


Fig. 7. System load for 128 bytes frame size.

important is the fact that when frame size increases, the difference in the use of processing power in each machine

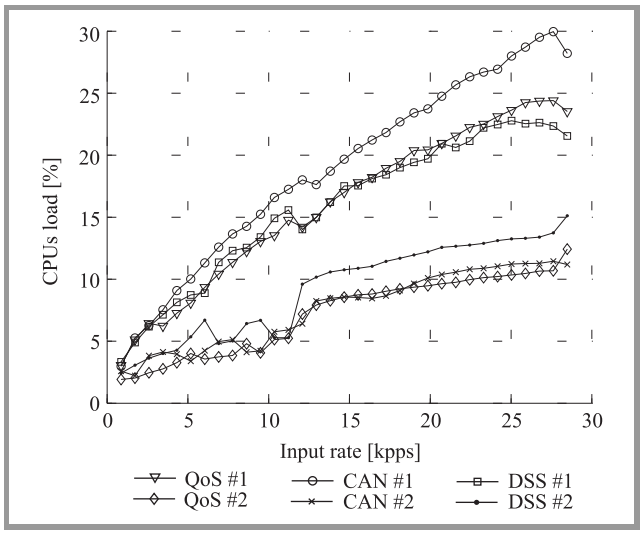


Fig. 8. System load for 1024 bytes frame size.

becomes lower. Larger fluctuations in load are noticeable for larger incoming throughput.

7.2. Random Frame Size

Next, we tested our system with random frame size. We used continuous linear distribution in the range from 79 to 1518 bytes to generate random frames (the only possible distribution in the Spirent TestCenter application). Tests were repeated 10 times and average results are presented.

Performance characteristics suggest a linear increase in computational load for each processor with increasing throughput. Only for the largest bandwidth the breakdown in the plots can be observed suggesting problems with handling so high volume of traffic (Fig. 9).

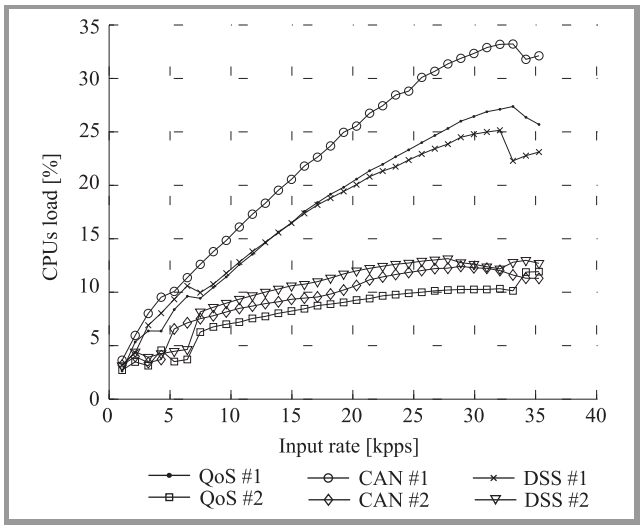


Fig. 9. System load for a random frame size.

As before, the greatest demand for computing power is imposed on VM handled traffic from CAN Parallel Internet. This machine performs the largest number of operations on the received header. There is no significant difference between the use of processor power for QoS IPv6 and DSS machines.

In the Figs. 10–11 capabilities of traffic passing through the system can be observed. These charts confirm relations that can be seen in the Fig. 9. When frames are dropped the average latency increases strongly from several microseconds to tens of milliseconds.

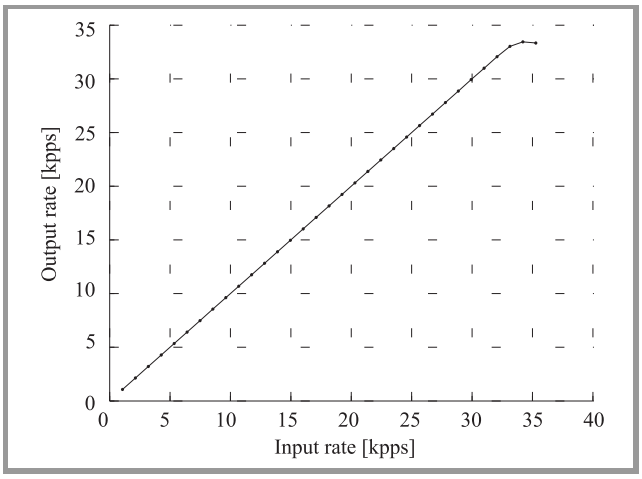


Fig. 10. Node throughput for a random frame size.

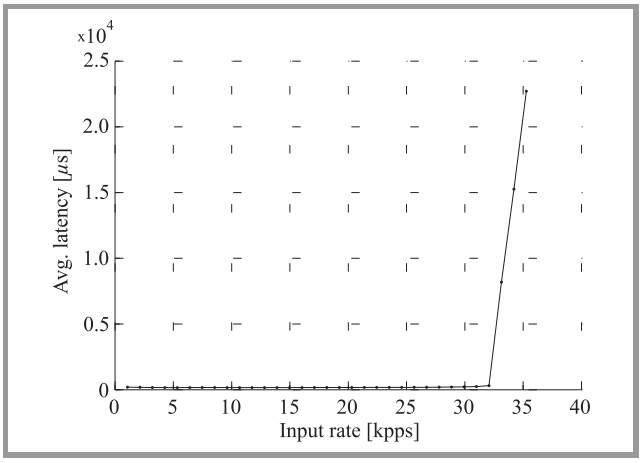


Fig. 11. Node latency for a random frame size.

Continuous uniform distribution is not the best to approximate the actual Internet traffic characterization. Unfortunately, only the linear distribution of traffic generation was possible in the Spirent TestCenter device. However, occurring randomness can in some way simulate Internet traffic. It should be noted that each Parallel Internet has individual traffic specification that it supports. In a case of CAN it will be a lot of queries searching for published content, and in IPv6 QoS the traffic will be served with accordance with specified classes.

7.3. Fitting of Linear Coefficients

Our investigations have shown strong correlation between indices, such as virtual link bandwidth and node's pro-

Table 1
Coefficients of linear equation

Frame size	QoS			CAN			DSS		
	a	b	R-square	a	b	R-square	a	b	R-square
79	3.2580	17.4400	0.9859	3.8780	20.7800	0.9827	2.5440	15.9600	0.9885
128	2.6810	8.4820	0.9859	3.1700	10.7200	0.9850	2.0680	9.1430	0.9832
256	1.4670	9.5110	0.9832	1.7810	12.1500	0.9882	1.1080	10.5500	0.9626
512	0.9354	7.4550	0.9561	1.0680	8.9720	0.9614	0.7271	9.0230	0.9182
1024	0.6521	4.4250	0.9842	0.7717	5.1020	0.9808	0.5987	5.2490	0.9637
1280	0.3863	4.8230	0.9646	0.4427	4.9360	0.9117	0.4262	4.5550	0.8367
1518	0.3717	3.9690	0.9614	0.3769	4.7130	0.8977	0.4678	3.3700	0.9452

cessing power, being strongly depending on frame size. This relation is linear, so we propose to find best coefficients of the linear Eq. 2. From Eq. 2 it is possible to calculate required processing power (the percentage of processor usage). Since first processor in each virtual node during our tests were more utilized than the second one, we focus on the calculation of coefficients only the first CPU.

$$CPU_{usage} = a \cdot x + b \tag{2}$$

where: a, b – coefficients of linear equation, CPU_{usage} – observed processor utilization, x – given throughput in kpps.

We have calculated best coefficients of this linear equation for each tested frame size. Values of these coefficients and coefficients of determination (R-square) [16] are shown in the Table 1. Sample fit for the CAN virtual machine handling 256 bytes size of frames is shown in the Fig. 12.

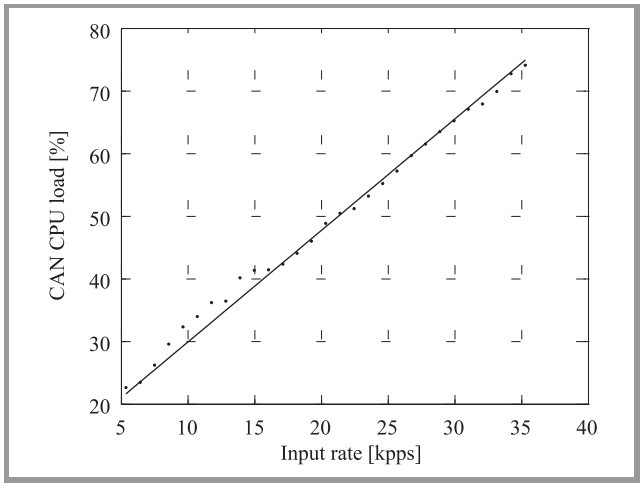


Fig. 12. Sample fit.

As we can observe in Table 1 there is strong dependance of adjusted coefficients from the frame size. Calculated coefficients can be used to estimate required computational power of the virtual node for given throughput value in provisioning phase of IIP System [17].

8. Conclusion

In this paper the preliminary results for performance evaluation of Xen-based node possessing both transport and computational functionalities were presented.

Performed tests indicate the rightness of the Xen implementation for the IIP System. Maximum use of CPU computing power for random frame sizes for each virtual machines does not exceed 35%. This allows for another or more complex program implementations associated with the processing of network traffic being performed inside each virtual machine. It is also possible to run programs not directly related with such procedures.

It is worth mentioning that the use of cyclic scheduler imposes a significant reduction in network performance compared the solution without it. This is due to the fact that the division of the total time slots causes larger or smaller periods of inactivity, in which the server can not send data. For example, when the frame 1518 comes to the scheduler it may be not possible to send it, because the remaining time slot can be too short. Then this frame must be queued and can be sent only after obtaining access to the link in the next time slot.

Our investigations provide linear model that can be used to estimate required processing power of the virtual machine for required throughput in dimensioning phase of IIP System.

Also it should be noted that the tested node constitutes the transport part of larger IIP System node.

The IIP project is now within implementation phase and this enables for running tests showing behavior of transport platforms and also to infer into inside the system performance in order to obtain measures quantifying both transport and processing aspects of served streams.

Acknowledgments

This work has been supported in part by the Polish Ministry of Science and Higher Education under the European

Regional Development Fund, Grant no. POIG.01.01.02-00-045/09-00 Future Internet Engineering. The authors would like to thank Błażej Adamczyk for valuable discussions.

References

[1] Akari project [Online]. Available: http://akari-project.nict.go.jp/eng/index2.htm/

[2] GENI project [Online]. Available: http://www.geni.net/

[3] 4WARD project [Online]. Available: http://www.4ward-project.eu/

[4] P. Barham *et al.*, "Xen and the art of virtualization", in *Proc. 19th ACM Symp. Oper. Sys. Princip. SOSP 2003*, New York, USA, 2003, pp. 164–177.

[5] N. Egi *et al.*, "Evaluating Xen for router virtualization", *Proc. 16th Int. Conf. Comp. Commun. Netw. ICCCN 2007*, Honolulu, Hawaii USA, 2007, pp. 1256–1261.

[6] A. Menon, A. L. Cox, and W. Zwaenepoel, "Optimizing network virtualization in xen", *in Proc. Ann. Tech. Conf. USENIX'06*, Boston, MA, USA, 2006, pp. 15–28.

[7] M. Bourguiba, K. Haddadou, and G. Pujolle, "Evaluating and enhancing Xen-based virtual routers to support real-time applications", in *Proc. IEEE Consum. Commun. Netw. Conf.*, Las Vegas, USA, 2010, pp. 1–5.

[8] B. Adamczyk and A. Chydziński, "On the performance isolation across virtual network adapters in Xen", in *Proc. 2nd Int. Conf. Cloud Comput. GRIDs Virtual. CLOUD COMPUTING 2011*, Rome, Italy, 2011, pp. 222–227.

[9] W. Burakowski *et al.*, "Architektura Systemu IIP", *Krajowe Sympozjum Telekomunikacji i Teleinformatyki*, Łódź, Poland, 2011, pp. 720–722 (in Polish).

[10] IIP project [Online]. Available: http://www.iip.net.pl/

[11] P. Zwierko *et al.*, "Platformy wirtualizacji dla Systemu IIP", *Krajowe Sympozjum Telekomunikacji i Teleinformatyki*, Łódź, Poland, 2011, pp. 824–831 (in Polish).

[12] B. De Schuymer *et al.*, "Ebtables" [Online]. Available: http://ebtables.sourceforge.net/

[13] W. Burakowski *et al.*, "Idealne urządzenie umożliwiające wirtualizację infrastruktury sieciowej w Systemie IIP", *Krajowe Sympozjum Telekomunikacji i Teleinformatyki*, Łódź, Poland, 2011, pp. 818–823 (in Polish).

[14] S. Bradner and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", Request for Comments: RFC 2544, 1999.

[15] C. Popoviciu, A. Hamza, G. Van de Velde, and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", Request for Comments: RFC 5180, 2008.

[16] C. A. Colin *et al.*, "An R-squared measure of goodness of fit for some common nonlinear regression models", *J. Econometrics*, vol. 77, no. 2, pp. 1790–1792, 1997.

[17] J. Gozdecki, M. Kantor, K. Wajda, and J. Rak, "A flexible provisioning module optimizing utilization of resources for the Future Internet IIP initiative", in *Proc. XVth Int. Telecommun. Netw. Strat. Plan. Symp. NETWORKS 2012*, Rome, Italy, 2012, pp. 1–6.

Grzegorz Rzym received his M.Sc. in Electronics and Telecommunications in 2012 and B.Sc. in Acoustic Engineering in 2013, both from AGH University of Science and Technology, Poland. Currently, he is a Ph.D. student at the Department of Telecommunications. His research interests cover management system design and implementation and virtualization.
E-mail: rzym@kt.agh.edu.pl
Department of Telecommunications
AGH University of Science and Technology
A. Mickiewicza av. 30
30-059 Kraków, Poland

Krzysztof Wajda received his M.Sc. in Telecommunications in 1982 and Ph.D. in 1990, both from AGH University of Science and Technology, Kraków, Poland. In 1982 he joined AGH-UST, where he was responsible for laboratory of switching technology. He spent a year at Kyoto University and half year in CNET (France). Dr. Wajda is currently an Assistant Professor at AGH University of Science and Technology. He was involved in a few international projects: COST 242, Leonardo da Vinci (JOINT and ET-NET), Copernicus ISMAN, ACTS 038 BBL, TEMPUS JEP No. 0971, IST LION, IP NOBEL, NoE e-photon/ONe(+), BONE, SmoothIT. At present he works in Future Internet Engineering project, Polish initiative towards NG Internet. He participated also in a few grants supported by National Science Foundation. He has been a consultant to private telecommunication companies. Main research interests: traffic management for broadband networks, performance evaluation, network reliability, control plane, management systems. K. Wajda is the author (or coauthor) of 6 books (in Polish) and over 100 technical papers. He is a member of IEEE.
E-mail: wajda@kt.agh.edu.pl
Department of Telecommunications
AGH University of Science and Technology
A. Mickiewicza av. 30
30-059 Kraków, Poland

Information for Authors

Journal of Telecommunications and Information Technology **(JTIT)** is published quarterly. It comprises original contributions, dealing with a wide range of topics related to telecommunications and information technology. **All papers are subject to peer review**. Topics presented in the JTIT report primary and/or experimental research results, which advance the base of scientific and technological knowledge about telecommunications and information technology.

JTIT is dedicated to publishing research results which advance the level of current research or add to the understanding of problems related to modulation and signal design, wireless communications, optical communications and photonic systems, voice communications devices, image and signal processing, transmission systems, network architecture, coding and communication theory, as well as information technology.

Suitable research-related papers should hold the potential to advance the technological base of telecommunications and information technology. Tutorial and review papers are published only by invitation.

Manuscript. TEX and LATEX are preferable, standard Microsoft Word format (.doc) is acceptable. The author's JTIT LATEX style file is available:
http://www.nit.eu/for-authors

Papers published should contain up to 10 printed pages in LATEX author's style (Word processor one printed page corresponds approximately to 6000 characters).

The manuscript should include an abstract about 150–200 words long and the relevant keywords. The abstract should contain statement of the problem, assumptions and methodology, results and conclusion or discussion on the importance of the results. Abstracts must not include mathematical expressions or bibliographic references.

Keywords should not repeat the title of the manuscript. About four keywords or phrases in alphabetical order should be used, separated by commas.

The original files accompanied with pdf file should be submitted by e-mail: redakcja@itl.waw.pl

Figures, tables and photographs. Original figures should be submitted. Drawings in Corel Draw and PostScript formats are preferred. Figure captions should be placed below the figures and can not be included as a part of the figure. Each figure should be submitted as a separated graphic file, in .cdr, .eps, .ps, .png or .tif format. Tables and figures should be numbered consecutively with Arabic numerals.

Each photograph with minimum 300 dpi resolution should be delivered in electronic formats (TIFF, JPG or PNG) as a separated file.

References. All references should be marked in the text by Arabic numerals in square brackets and listed at the end of the paper in order of their appearance in the text, including exclusively publications cited inside. Samples of correct formats for various types of references are presented below:

[1] Y. Namihira, "Relationship between nonlinear effective area and mode field diameter for dispersion shifted fibres", *Electron. Lett.*, vol. 30, no. 3, pp. 262–264, 1994.

[2] C. Kittel, *Introduction to Solid State Physics*. New York: Wiley, 1986.

[3] S. Demri and E. Orłowska, "Informational representability: Abstract models versus concrete models", in *Fuzzy Sets, Logics and Knowledge-Based Reasoning*, D. Dubois and H. Prade, Eds. Dordrecht: Kluwer, 1999, pp. 301–314.

Biographies and photographs of authors. A brief professional author's biography of up to 200 words and a photo of each author should be included with the manuscript.

Galley proofs. Authors should return proofs as a list of corrections as soon as possible. In other cases, the article will be proof-read against manuscript by the editor and printed without the author's corrections. Remarks to the errata should be provided within one week after receiving the offprint.

Copyright. Manuscript submitted to JTIT should not be published or simultaneously submitted for publication elsewhere. By submitting a manuscript, the author(s) agree to automatically transfer the copyright for their article to the publisher, if and when the article is accepted for publication. The copyright comprises the exclusive rights to reproduce and distribute the article, including reprints and all translation rights. No part of the present JTIT should not be reproduced in any form nor transmitted or translated into a machine language without prior written consent of the publisher.
For copyright form see: http://www.nit.eu/for-authors

A copy of the JTIT is provided to each author of paper published.

Journal of Telecommunications and Information Technology has entered into an electronic licencing relationship with EBSCO Publishing, the world's most prolific aggregator of full text journals, magazines and other sources. The text of *Journal of Telecommunications and Information Technology* can be found on EBSCO Publishing's databases. For more information on EBSCO Publishing, please visit www.epnet.com.

(*Contents Continued from Front Cover*)

Call and Connections Times in ASON/GMPLS Architecture

S. Kaczmarek, M. Młynarczuk, and P. Zieńko — *Paper* — *80*

Single Hysteresis Model for Limited-availability Group with BPP Traffic

M. Sobieraj, M. Stasiak, J. Weissenberg, and P. Zwierzykowski — *Paper* — *89*

Interworking and Cross-layer Service Discovery Extensions for IEEE 802.11s Wireless Mesh Standard

K. Gierłowski — *Paper* — *97*

Cooperative Games with Incomplete Information for Secondary Base Stations in Cognitive Radio Networks

J. Martyna — *Paper* — *106*

Quasi-Offline Fair Scheduling in Third Generation Wireless Data Networks

J. Martyna — *Paper* — *112*

Performance Tests of Xen-based Node for Future Internet IIP Initiative

G. Rzym and K. Wajda — *Paper* — *115*

INSTYTUT ŁĄCZNOŚCI
PAŃSTWOWY INSTYTUT BADAWCZY

Editorial Office

National Institute
of Telecommunications
Szachowa st 1
04-894 Warsaw, Poland

tel. +48 22 512 81 83
fax: +48 22 512 84 00
e-mail: redakcja@itl.waw.pl
http://www.nit.eu

JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY 2/2013