Modeling a QoS classified communication in a multiuser Wi-Fi environment

Attila Kuki, Béla Almási, Tamás Bérczes, János Sztrik

Abstract—**The aim of the present paper is to propose a finite source queueing model in order to include the random backoff feature of the wireless communication. Two classes of sources (high priority and low priority traffic) are included. The random backoff feature is implemented by using retrial queues for each traffic class. Supposing exponentially distributed inter-event times the MOSEL software tool is used to develop the special software to calculate the most important steady-state performance characteristics of the system, such as utilizations, mean orbit sizes, mean waiting times, that is mean time spent in the orbit. It is showed how the retrial discipline effects the mean waiting times (compared to the FIFO discipline): not only the values, but also the form of the curve is quite different in the case of packet reordering.**

I. Introduction

The investigation of the users' connection to a Wi-Fi access point is a focused research area today.

For the mathematical analysis, queueing models are widely used to create stochastic models which can be used to calculate the most important performance characteristics of the communication systems (e.g. utilizations, response times) (see e.g. [3], [4], [5], [6], [7], [8], [9]). Using the classic FIFO service discipline in these models are not suitable for modeling the wireless network environment, because the random backoff feature of the wireless access is completely omitted. Although the queueing model technology can be used efficiently to model and study the influence of the priorities on the QoS performance between the traffic classes (see [10]), but it can be hardly used to model the wireless access feature.

In [1] the authors created a queueing model to evaluate a sensor network environment with two quality classes of sources. The Emergency class represents the sources of very important communication (e.g. fire alarm), and the Normal class represents the standard communication (e.g. temperature data). In the model of [1] the Emergency class is served by a FIFO queue. The FIFO discipline absolutely excludes the ability of request reordering. The random backoff feature of the Wi-Fi access can not be precisely described by the FIFO discipline.

In this paper we would like to introduce a new element in the queueing modeling of the QoS performance. The basic idea of our solution appeared in [2], where the authors used two orbits (retrial queues) in an infinite queueing model. The requests staying in the orbit are randomly retrying the transmission, so opening the possibility of describing the random backoff feature of the wireless network access. We establish a finite source queueing model containing two QoS traffic classes. For the service of the requests in each class we use retrial systems, random backoff may occur inside the classes too. In this paper we use finite source queueing system to model small sized network environments, where the number of users may not be considered as infinite.

The main contribution of the present paper is to introduce a finite source queueing system with two orbits, which is more suitable for modeling small environments (e.g. a picocell or femtocell sized radio networks, where the number of sources may not be considered as infinite). We would like to study the system parameters of the multiuser Wi-Fi access environment (queue length, service time etc).

The rest of the paper is organized as follows. Section 2 describes the precise mathematical model when a multi-dimensional continuous time Markov-chain is defined for describing the system's dynamics. Formulas of the most important performance measures are also discussed here. For presentation of numerical results the MOdeling Specification and Evaluation Language (MOSEL see [11]) tool is used. In Section 3 we investigate a concrete environment, and we study the effect of different parameters on the systems performance. Finally, the conclusion closes the paper.

II. System Model

For modeling the effect of the wireless access problem, let's see a queueing model with a single server unit, where the jobs come from two classes of finite sources. These sources represent the incoming packets of the Wi-Fi connected users. The first class of sources corresponds to the high priority sessions (eg. interactive voice or video stream), and the second class of sources refers to the low priority sessions (eg. file transfer or database transaction). The number of sources of the high priority class is denoted by N, and the number of sources of the low priority class is denoted by K. The sources of both classes may send a new service request (ie. a new packet is sent through the communication channel). The distribution of the inter-request times is exponential with parameter λ_1 for the high priority packets and with parameter λ_2 for the low priority class.

Because there are no queues (buffers) for either high priority class or low priority class, there can be at most one request in the service area. So, the server can be engaged with a request from the high priority class, or with a request from the low priority class (or it can be idle).

Manuscript received June 13, 2014, revised August 31, 2014.

The authors are with the Department of Informatics Systems and Networks, Faculty of Informatics, University of Debrecen, Hungary, e-mail: {berczes.tamas,almasi.bela,sztrik.janos,kuki}@inf.unideb.hu

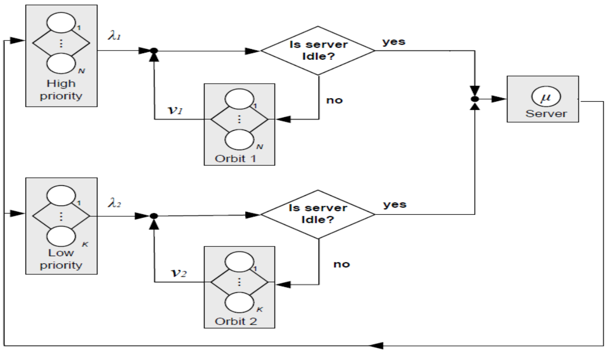

Fig. 1. A retrial queue with components

For a request generated from the high priority sources there are two possibilities. In case of the idle server the request is transmitted directly to the server, where the request will be served with an exponentially distributed service time. The service rate is denoted by μ. When the request is served, it goes back to the high priority source. In case of the busy server, the request is sent to the orbit from the high priority sources. From the orbit the request will retry to be served after an exponentially distributed retrial time, the retrial rate is denoted by ν_1. Based on these random times, the order of high priority packets arriving to the orbit may differ from the order of packets leaving the orbit.

For a request generated from the low priority sources there are these two possibilities, as well. In case of the idle server the request is transmitted directly to the server, where the request will be served with an exponentially distributed service time, with the same service rate μ. When the request is served, it goes back to the low priority source. In case of the busy server, the request is sent to the orbit from the low priority sources. From the orbit the request will retry to be served after an exponentially distributed retrial time, the retrial rate is denoted by ν_2. Again, based on these random times, the order of low priority packets arriving to the orbit may differ from the order of packets leaving the orbit.

The functionality of this communication network is presented on Fig. 1.

To create a stochastic process describing the behavior of the system the following notations are introduced (Table I contains the overview of parameters of the network):

- $k_1(t)$ is the number of active sources in the high priority class at time t,
- $k_2(t)$ is the number of active sources in the low priority class at time t,
- $o_1(t)$ is the number of requests in the orbit for high priority requests at time t,
- $o_2(t)$ is the number of requests in the orbit for low priority requests at time t,
- $y(t) = 0$ if there is no request in the server at time t. The server is available and ready to receive a job. $y(t) = 1$ if the server is engaged with a request coming from the high priority class, and $y(t) = 2$ if the server is engaged with a job coming from the low priority class at time t.

It is easy to see that:

$$k_1(t) = \begin{cases} N - o_1(t), & y(t) = 0, 2 \\ N - o_1(t) - 1, & y(t) = 1 \end{cases}$$

and

$$k_2(t) = \begin{cases} K - o_2(t), & y(t) = 0, 1 \\ K - o_2(t) - 1, & y(t) = 2 \end{cases}$$

TABLE I
LIST OF NETWORK PARAMETERS

Parameter	Maximum	Value at t	Unit
Active high priority s.	N	$k_1(t)$	-
Active low priority s.	K	$k_2(t)$	-
High priority gen. rate		λ_1	1/s
Low priority gen. rate		λ_2	1/s
Total gen. rate	$\lambda_1 N + \lambda_2 K$	$\lambda_1 k_1(t) + \lambda_2 k_2(t)$	1/s
Requests in high pr. orbit	N	$o_1(t)$	-
Requests in low pr. orbit	K	$o_2(t)$	-
Ret. rate in high pr. orbit		ν_1	1/s
Ret. rate in low pr. orbit		ν_2	1/s
Service rate		μ	1/s

In order to obtain the steady-state probabilities and performance measures within the Markovian framework, the mathematical tractability of the proposed model should be preserved. Therefore, we follow the classical approach frequently applied in the theory of retrial queues for the performance evaluation of infocommunication systems, namely, the distributions of inter-event times (i.e., request generation times for low and high priority packets, service time, retrial times) presented in the system are assumed to be exponentially distributed and totally independent of each other.

Consequently, the state of the network at a time t can be described by a Continuous Time Markov Chain (CTMC) with 3 dimensions:

$$X(t) = (y(t); o_1(t); o_2(t))$$

The steady-state distributions are denoted by

$$P(y, o_1, o_2) = \lim_{t\to\infty} P(y(t) = y, o_1(t) = o_1, o_2(t) = o_2)$$

Note that in the present case, the unique stationary distribution always exists, because the underlying CTMC is irreducible and the state space of the CTMC is finite. For computing the steady-state probabilities and the system characteristics, the MOSEL-2 software tool is used. These computations are similar to the ones described in, for example [12], [13].

As soon as we have calculated the distributions defined above, the most important steady-state system performance measures can be obtained in the following way:

- *Utilization of the Server with respect to high priority packets*

$$U_{S_1} = \sum_{o_1=0}^{N-1} \sum_{o_2=0}^{K} P(1, o_1, o_2)$$

- *Utilization of the Server with respect to low priority packets*

$$U_{S_2} = \sum_{o_1=0}^{N} \sum_{o_2=0}^{K-1} P(2, o_1, o_2)$$

- *Overall utilization of the Server*

$$U_S = U_{S_1} + U_{S_2}$$

- *Mean number of jobs in the orbit for high priority requests*

$$\overline{O_1} = E(o_1(t)) = \\ = \sum_{y=0}^{2} \sum_{o_1=0}^{N} \sum_{o_2=0}^{K} o_1 P(y, o_1, o_2)$$

- *Mean number of jobs in the orbit for low priority requests*

$$\overline{O_2} = E(o_2(t)) = \\ = \sum_{y=0}^{2} \sum_{o_1=0}^{N} \sum_{o_2=0}^{K} o_2 P(y, o_1, o_2)$$

- *Mean number of high priority jobs in the network*

$$\overline{M_1} = \overline{O_1} + U_{S_1}$$

- *Mean number of low priority jobs in the network*

$$\overline{M_2} = \overline{O_2} + U_{S_2}$$

- *Mean number of jobs in the network*

$$\overline{M} = \overline{M_1} + \overline{M_2}$$

- *Mean number of active high priority sources*

$$\overline{\Lambda_1} = N - \overline{M_1}$$

- *Mean number of active low priority sources*

$$\overline{\Lambda_2} = K - \overline{M_2}$$

- *Mean generation rate of high priority sources*

$$\overline{\lambda_1} = \lambda_1 \overline{\Lambda_1}$$

- *Mean generation rate of low priority sources*

$$\overline{\lambda_2} = \lambda_2 \overline{\Lambda_2}$$

- *Mean waiting time in orbit for high priority requests*

$$\overline{W_1} = \frac{\overline{O_1}}{\overline{\lambda_1}}$$

- *Mean waiting time in orbit for low priority requests*

$$\overline{W_2} = \frac{\overline{O_2}}{\overline{\lambda_2}}$$

- *Mean response time of high priority requests*

$$\overline{T_1} = \frac{\overline{M_1}}{\overline{\lambda_1}}$$

- *Mean response time of low priority requests*

$$\overline{T_2} = \frac{\overline{M_2}}{\overline{\lambda_2}}$$

III. NUMERICAL RESULTS

Investigating the functionality and the behavior of the system several numerical calculations were performed. From the steady-state probabilities computed by MOSEL-2 tool the most interesting performance characteristics were obtained, which are graphically presented in this section. The numerical values of the model parameters are described in Table I and in Table II. In the calculations a single variable λ is used for the generation rates. The high priority generation rate is $\lambda_1 = \lambda$, and the low priority generation rate is $\lambda_2 = 2\lambda$.

On the Figures 2 - 8 the three different lines represent the effects of different retrial rates in the orbit for the high priority requests ($\nu_1 = 2$: blue lines, dotted with rhombus, $\nu_1 = 4$: red lines, dotted with squares, $\nu_1 = 8$: green lines, dotted with triangles). On Figure 9 the two lines correspond to the different functionality of the server (described in [1]).

TABLE II
NUMERICAL VALUES OF MODEL PARAMETERS

Case studies							
No.	N	K	λ	ν_1	ν_2	μ	y-axis
Fig. 2	50	50	$x-axis$, [0.1..1]	$2,4,8$	2	20	$\overline{O_1}$
Fig. 3	50	50	$x-axis$, [0.1..1]	$2,4,8$	2	20	$\overline{O_2}$
Fig. 4	50	50	$x-axis$, [0.1..4.6]	$2,4,8$	2	20	$\overline{W_1}$
Fig. 5	50	50	$x-axis$, [0.1..4.6]	$2,4,8$	2	20	$\overline{W_2}$
Fig. 6	50	50	$x-axis$, [0.1..1]	$2,4,8$	2	20	U_s
Fig. 7	50	50	$x-axis$, [0.1..1]	$2,4,8$	2	20	U_{s_1}
Fig. 8	50	50	$x-axis$, [0.1..1]	$2,4,8$	2	20	U_{s_2}
Fig. 8	50	50	$x-axis$, [0.1..4.6]	2	2	20	$\overline{O_2}$

On Figure 2 the mean orbit size is displayed for high priority requests. When the generation rate is increased, the size of the orbit will be larger. The size of the orbit depends on the retrial rate, as well. In case of lower retrial rate, the size of orbit will be significantly larger.

Figure 3 shows the size of the orbit for low priority requests. Compared to Figure 2, a reverse tendency can be observed here. As we increase the retrial rate of the orbit for high priority requests, the low priority requests will have difficulties reaching the server (higher value of ν_1 implies larger $\overline{O_2}$). In addition, the size of orbit for low priority request fills up faster than the size of the other orbit. This is a straight consequence of the larger generation rate for low priority requests.

Figure 4 displays the Overall Utilization of the Server as function of λ. The utilization of the server (U_s) rises dramatically at the beginning: at value of $\lambda = 0.3$ the utilization equals to 85 percent.

On Figures 5 and 6 the utilization of the server with respect to high and low priority requests are shown. In cases of higher values of ν_1 the U_{s_1} values will be greater. Reverse effect stands for low priority requests, because for higher values of ν_1 (high priority retrial rate), larger number of high priority service demand will be present in the system. If $\nu_2 = \nu_1 = 2$, the server utilization curves for the two priority classes are the same (the blue lines. When the retrial rate of ν_1 is greater than ν_2 (red and green lines), the lines have maximum points.

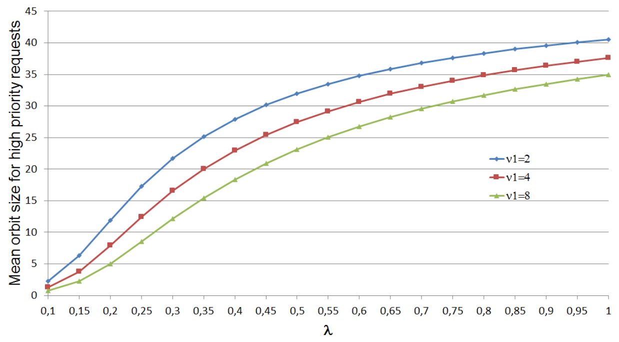

Fig. 2. Mean orbit size for high priority requests vs λ

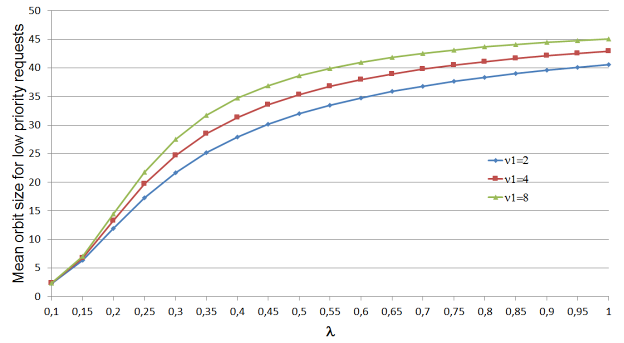

Fig. 3. Mean orbit size for low priority requests vs λ

For small values of generation rates the probabilities of CPU serves low priority requests are increasing, and higher values of generation rates the large number of high priority retrials will lower the considered probabilities. On Figure 4 it can be seen, that for higher values of generation rate (greater than 0.4) the utilization is almost constant (0.9). As U_{s_1} increases with the higher generation rates, too, consequently U_{s_2} has to be decreased.

In the Figures 7 and 8 we would like to investigate the effect of the random backoff feature on the mean waiting times (i.e. the mean times spent in the orbits). The parameters of these performance measures were the same like it was in [1]. Obviously, the larger generation rates will cause more time spent in the orbits. Because of the larger generation rate, for low priority requests $\overline{W_2}$ increases faster than $\overline{W_1}$ for high priority requests. Comparing the Figure 8 to the one presented in [1] (see Figure 9) we can see big differences caused by replacing the FIFO queue with an orbit: the values of the mean waiting times in the case of using FIFO discipline for the high priority requests are quite larger than in the case of using two orbits (the case for modeling the random backoff feature of the wireless access). On the other hand, it can be stated, that the functionalities of orbits connected with an other orbit and connected with a FIFO queue are different. In model of [1] the curves are first concave, then turn into convex, in recent model they are first convex, then change into concave. Thus the presence of the wireless access issue in the high priority class (which are not present in model of [1]) changes significantly the working conditions for the low priority class.

IV. CONCLUSION

In this paper a finite source queueing model was created in order to include the wireless network access feature of the communication. Two classes of sources (high priority and low priority traffic) were investigated. The random backoff feature of the Wi-fi access was implemented by using retrial queues for each traffic class. The conceptual working scheme of the model was described by a multidimensional Markov Chain, and the MOSEL software tool was used to develop the special software in order to calculate the most important steady-state performance characteristics of the system. At the end of the

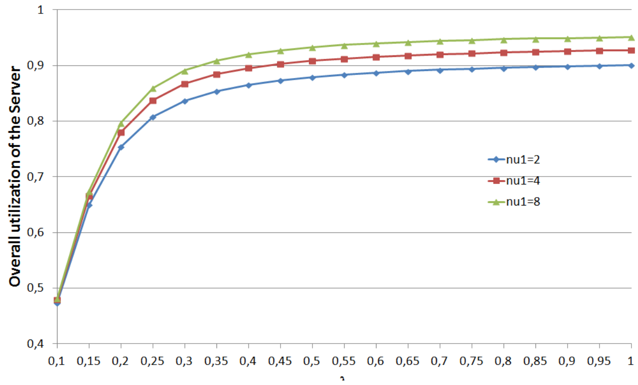

Fig. 4. Overall utilization of Server vs λ

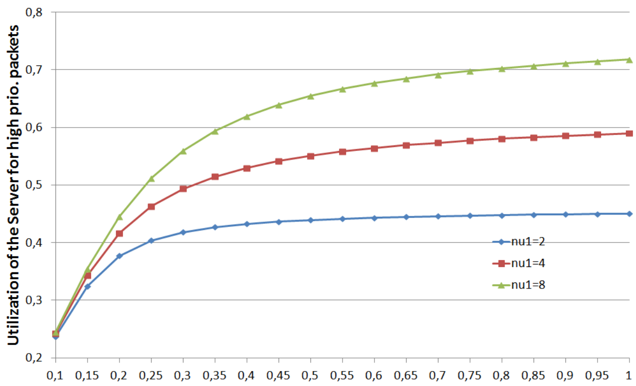

Fig. 5. Utilization of server for high priority packets vs λ

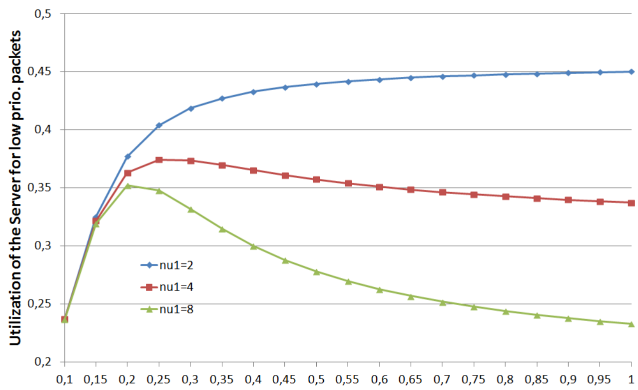

Fig. 6. Utilization of server for low priority packets vs λ

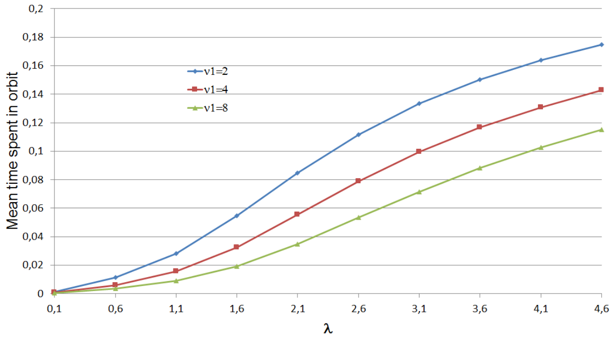

Fig. 7. Mean time spent in orbit for high priority requests vs λ

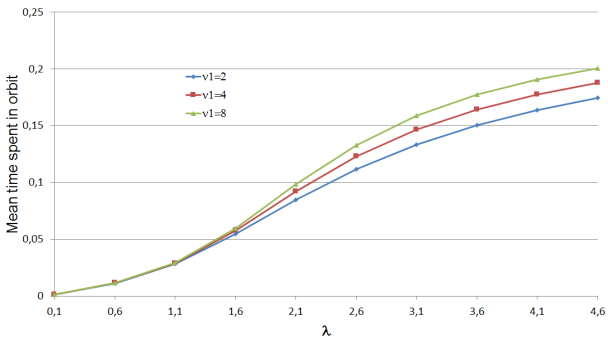

Fig. 8. Mean time spent in orbit for low priority requests vs λ

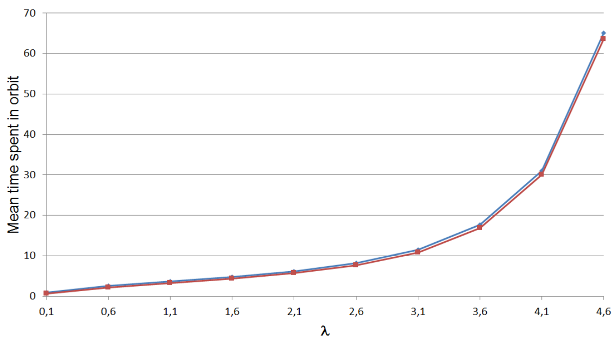

Fig. 9. Mean time spent in orbit for low priority requests vs λ (FIFO for high priority requests)

paper we showed how the feature of the random backoff effects the mean waiting times (compared to the FIFO discipline): not only the values, but also the form of the curve is quite different in the case of the considered wireless communication environment.

ACKNOWLEDGMENT

The research of A. Kuki and B. Almási was supported by the TÁMOP 4.2.2. C-11/1/KONV-2012-0001 project. The project has been supported by the European Union, co-financed by the European Social Fund.

The work of János Sztrik was realized in the frames of TÁMOP 4.2.4. A/2-11-1-2012-0001 National Excellence Program - Elaborating and operating an inland student and researcher personal support system. The project was subsidized by the European Union and co-financed by the European Social Fund.

The work of Tamás Bérczes was realized in the frames of TÁMOP 4.2.4. A/2-11-1-2012-0001 National Excellence Program - Elaborating and operating an inland student and researcher personal support system. The project was subsidized by the European Union and co-financed by the European Social Fund.

REFERENCES

[1] Bérczes, T., Almási, B., Sztrik, J., Kuki, A. ”A contribution to modeling sensor communication networks by using finite-source queueing systems,” Proc. 8th IEEE International Symposium on Applied Computational Intelligence and Informatics, Timisoara, Romania pp. 89-93, 2013.

[2] Avrachenkov, K., Nain, P., Yechiali, U. ”A retrial system with two input streams and two orbit queues,” Research Report no. 7999, Research Centre SOPHIA ANTIPOLIS MDITERRANE France, June, 2012.

[3] H. T. Tran, T. V. Do, László Pap ”Analysis of a queue with two priority classes and feedback controls,” Vietnam J. Computer Science 1(2): 71-78 (2014).

[4] Artalejo, J., and Gomez-Corral A. ”Retrial Queueing Systems,” Springer, Berlin, 2008

[5] Basharin, G. P., Samouylov, K. E., Yarkina, N. V. and Gudkova, I. A. ”A new stage in mathematical teletraffic theory,” Automation Remote Control, vol. 70, no. 12 pp. 1954-1964, 2009.

[6] Feng, W., and Kowada, M. ”Performance analysis of wireless mobile networks with queueing priority and guard channels,” International Transactions on Operational Research, no. 15, pp. 481- 508, 2008.

[7] Lakatos, L., Szeidl, L., and Telek, M. ”Introduction to queuing systems with telecommunication applications,” Springer, Heidelgerg, 2013.

[8] Ponomarenko, L., Kim, C. S., and Melikov, A. ”Performance analysis and optimization of multi-traffic on communication networks,” Springer, Heidelgerg, 2010.

[9] Stasiak, M., Glabowski, M., Wishniewski, A., and Zwierzykowski, P. ”Modeling and dimensioning of mobile networks,” John Wiley, Chichester, 2011.

[10] Szilágyi, Sz. Almási, B. ”A Review of Congestion Management Algorithms on Cisco Routers,” Journal of Computer Science and Control Systems vol. 5, no. 1, pp. 103-107, 2005.

[11] Begain, K., Bolch, G., and Herold, H. ”Practical Performance Modeling, Application of the MOSEL Language,” Kluwer Academic Publisher, Boston, 2001.

[12] Wüchner, P., Sztrik, J., and de Meer, H. ”Modeling wireless sensor networks using finite-source retrial queues with unreliable orbit,” Springer Lecture Notes in Computer Science, no. 6821, pp. 275-285, 2011.

[13] Bolch, G., Greiner, S., de Meer, H., and Trivedi, K. ”Queueing Networks and Markov Chains,” John Wiley & Sons, New York, 2nd edition, 2006.

Tamás Bérczes received his MSc Degree in Mathematics in 2000. at the University of Debrecen, Hungary. He is currently assistant professor at the Department of Informatics Systems and Networks of the same university. He received his Ph.D. in 2011. His primary research interests are performance analysis of retrial queues and their application.

Béla Almási is a associate professor at the Department of Informatics Systems and Networks at the Faculty of Informatics, University of Debrecen, Hungary. He received his Ph.D. in 1998. at the University of Debrecen, Hungary and Habilitation from University of Debrecen in 2006. His primary research interests are network systems, performance analysis of retrial queues and their application.

Attila Kuki He is currently an assistant professor at the Department of Informatics Systems and Networks at the Faculty of Informatics, University of Debrecen. H e received his Ph.D. in 1997 in Mathematics. His primary research interests are performance analysis of retrial queues and their application.

János Sztrik is a Full Professor at the Faculty of Informatics, University of Debrecen, Hungary. He obtained the Candidate of Mathematical Sciences Degree in Probability Theory and Mathematical Statistics in 1989 from the Kiev State University, Kiev, USSR, Habilitation from University of Debrecen in 1999, Doctor of the Hungarian Academy of Sciences, Budapest, 2002. His research interests are in the field of production systems modeling and analysis, queueing theory, reliability theory, and computer science.