Numerical Model of Generelized Multi-Priority Queuing System

Eimutis Valakevicius, Mindaugas Snipas

Department of Mathematical Research in Systems

Kaunas University of Technology

Kaunas, Lithuania

E-mails: {eimval@ktu.lt, minsnip@ktu.lt}

Abstract—**The paper presents a novel approach of semi-automatic technique for creation numerical models of stochastic queuing systems of general type. The general model can be reduced to a queuing system of desired construction: with various numbers of queues, servers, any service time distributions and different priorities. Markov chains are used to model the dynamics of systems with phase–type distributed service times. Event–driven approach is applied to model the change of the system. The calculated stationary probabilities of possible states of the system are used to compute measures of the system performance.**

Keywords-generalized queueing system; numerical model; phase–type distribution; Markov chain; stationary probabilities

I. INTRODUCTION

Queuing models are useful mathematical tools in the study of design problems of a wide variety of stochastic service systems, such as, computer and telecommunication networks, inventory, logistic and other complex systems. There are a lot of publications in scientific literature devoting to the construction of analytical models of Markovian queuing systems [1][2][3][4][5]. Some queuing systems are modeled using numerical approach. Usually, the analytic approach is effective only for Markovian queuing systems with infinite waiting room. However, it is difficult or often impossible to find analytical solutions of non-Markovian multi-class and multi-server priority queues. Some researches restricting themselves analyzing two priority classes or limiting the number of high or low priority jobs [6][7]. In other papers multi-priority, multi-server system is roughly approximated by a single server system [8] or aggregating the multi-priority classes into two classes, as in [9][10]. This article proposes a numerical approach for semi-automatic construction numerical models of complex generalized queuing systems. The numerical technique allows modeling a wide variety of stochastic service systems. The paper is an enhancement of articles published in [11][12] and can be considered as a "work in progress". So, it was not possibility to compare the proposed technique with related work.

The paper is organized as follows. Section 2 gives description of a generalized queuing system under consideration. Section 3 describes the algorithm implemented in to programming tool to create a numerical model. The example is given in Section 4 and the paper is concluded in Section 5.

II. DESCRIPTION OF THE STOCHASTIC MODEL

Consider a queuing system with k identical servers and n priority classes: the highest priority 1 and lowest priority n. Priority rules is non - preemptive. In queuing applications, it is often convenient to approximate the service time by distributions that are built out of a finite sum or a finite mixture of exponentially distributed components. These distributions are called phase-type distributions. The application of such distribution allows constructing Markovian model [13] of the system. The queuing system under consideration assumes Poisson arrival processes [14] of n class's customers with rates $\lambda_i, i = 1, \ldots n$ and phase-type distributed service times with parameters $\mu_i^{(j)}, i = 1, \ldots k; j = 1,2$. Two phase-type distribution will be used in the model. The approximation of general type distribution by two phase-type distribution is discussed in [12]. The scheme of simplified queuing system with $n = 2$ and $k = 1$ is represented in Figure 1:
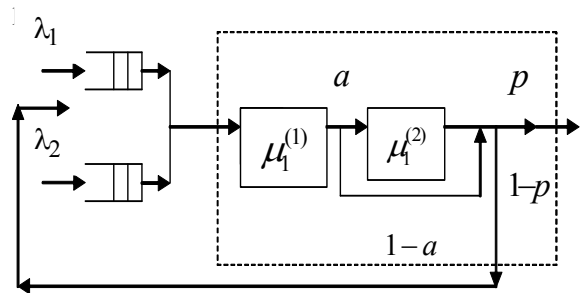

Figure 1. The simplified queuing system

Within each class, service discipline of customers is a First Come First Served (FCFS*)*. Queue length of each class customers is limited. Each served customer can leave the system with probability p or can be returned for repeated service with probability 1-p.

III. PROCESS OF CREATING A NUMERICAL MODEL OF THE QUEUEING SYSTEM

We will present the algorithm of creating the numerical model of a queuing system under consideration. It can be considered as consisting of five parts [15].

1. Define the state vector of the system. All possible combinations of coordinates describe the set of all possible system states.

2. Identify all the possible events which change the state of the system.

3. Describe all the events for generation system states and transition matrix.

4. Compute the stationary probabilities of the states.

5. Calculate the performance measures of the system.

The performance of the generalized queuing system will be described according the suggested algorithm.

The state of the system is given by the vector

$$S = (s_i, i = 1, \ldots, n; s_j, j = n+1, \ldots, n+k; \quad (1)$$
$$s_m, m = n+k+1, \ldots n+2k),$$

here

s_i, $i = \overline{1,n}$ - the number of customers in the ith queue.

$$s_j = \begin{cases} i, & \text{if the } (j-n)\text{th server is busy} \\ & \text{by customer from } i\text{th queue;} \\ 0, & \text{if the server is idle;} \\ & j = n+1, \ldots, n+k; \end{cases} \quad (2)$$

$$s_m = \begin{cases} 0, & \text{if the } (m-n-k)\text{th server is idle;} \\ 1, & \text{if the customer is served} \\ & \text{in the } (m-n-k) \text{ th server in} \\ & \text{the first phase;} \\ 2, & \text{if the customer is served} \\ & \text{in the } (m-n-k)\text{th server in the} \\ & \text{second phase;} \\ & m = n+k+1, \ldots, n+2k; \end{cases} \quad (3)$$

The values of coordinates are bounded so:

$$s_i \leq l_i, i = 1, \ldots, n. \quad (4)$$

The set of transition rates among states is the following:

$$Rates = \{\, \lambda_i, i = 1, \ldots, n, \mu_j^{(1)}, j = 1, \ldots, k,$$
$$\mu_j^{(2)}, j = 1, \ldots, k\}, \quad (5)$$

where

λ_i - the arrival rate of ith class customers to the system;

$\mu_j^{(i)}$ - the service rate of a customer in jth server and ith stage.

The set of possible events in the system:

$$E = \{e_1; e_i, i = 2, \ldots, n;$$
$$e_j, j = n+1, \ldots, n+k;$$
$$e_m, m = n+k+1, \ldots n+2k; \quad (6)$$
$$e_r, r = n+2k+1, \ldots, n+3k;$$
$$e_v, v = n+3k+1, \ldots, n+4k;$$
$$e_t, t = n+4k+1, \ldots, n+5k\},$$

where

e_i - a customer of the ith class arrived to system with rate λ_i ;

e_j - a customer was served in $(j\text{-}n)$th server in the first phase and with rate $\mu_j^{(1)} p \ (1\text{-}a)$ leaves the system;

e_m - a customer was served in $(m\text{-}n\text{-}k)$th server in the first phase and with rate $\mu_{m-n-k}^{(1)} (1\text{-} p)(1\text{-}a)$ returns for repeated service;

e_r - a customer was transferred from the first phase to the second phase in the $(r\text{-}n\text{-}2k)$th server with rate $\mu_{r-n-k}^{(1)} a$;

e_v - a customer was served in $(v-n-3k)$ th server and left the system with rate $\mu_{v-n-k}^{(2)} p$;

e_t - a customer was served in $(t\text{-}n\text{-}4k)$th server and was returned for repeated service with rate $\mu_{t-n-k}^{(2)} (1\text{-}p)$;

The initial state is

$$S_0 = (0, \ldots, 0).$$

The description of the event e_1 is given below.

e_1 :

$$\textbf{if} \qquad s_1 < l_1$$
$$\textbf{then if}$$
$$(s_1 > 0 \ and \ (s_m = 1 \ or \ s_m = 2)),$$
$$m = n + k + 1, \ldots, n + 2k)$$
$$\textbf{then} \qquad s_1 := s_1 + 1$$
$$\textbf{end if}$$
$$\textbf{else}$$
$$j := \min\{p : s_p = 0, \ p = n+1, \ldots, n+k\}$$
$$s_j := 1, \ s_{j+k} := 1$$
$$\textbf{end if}$$
$$\textbf{Return } \text{Rate} := \lambda_1$$

The created programming tool using described events generates the possible set of states, constructs the transition matrix among them, calculates stationary probabilities of system states

$$\pi(s_i, i = 1, \ldots, n; s_j, j = n+1, \ldots, n+k; \\ s_m, m = n+k+1, \ldots n+2k) := \pi(s_i; s_j; s_m) \qquad (7)$$

and computes performance measures of the system. The formulas for computing desired measures must be created. For example, the mean queue length $E(L_i)$ of ith class customers in the system is calculated according the following formula

$$E(L_i) \\ = \sum_{s_i, i=1,\ldots,n} \sum_{s_j, j=n+1,\ldots,n+k} \sum_{s_m, m=n+k+1,\ldots,n+2k} s_i \cdot \pi(s_i; s_j; s_m) \qquad (8)$$

IV.   EXAMPLE

Consider a queuing system with 3 priority classes and 2 identical servers. This queuing system assumes Poisson arrival processes with rates $\lambda_i, i = 1, \ldots, n$ and exponentially distributed service times with parameters $\mu_j^{(1)}$ and $\mu_j^{(2)} = 0$, $j = 1, \ldots, k$ respectively.

The main problem when modelling real systems is a rapid growth of the number of states of a Markov chain. Computation of stationary probabilities can require a large amount of calculations and computer resources. For example, if the set of states is described as

$$S = \{(s_1, s_2, s_s, s_4, s_5)\}, s_1 \leq l_1; s_2 \leq l_2; s_3 \leq l_3; s_4 \leq l_4; s_5 \leq l_5, (9)$$

it is easy to prove that the total number of states equal to

$$(l_1 + 1)(l_2 + 1)(l_3 + 1)(l_4 + 1)(l_5 + 1) \cdot \qquad (10)$$

We chose limitation on summary waiting space

$$l_1 + l_2 + l_3 \leq 7 \cdot \qquad (11)$$

We estimated the mean queue length for each customer class and general loss probability $P(Loss)$ (i.e., the probability that the customer of any class will not be served).

Numerical modeling results were compared with simulation results using ARENA simulation software. During each simulation session total amount of more than 5 500 000 customer arrivals were generated. We used PC with AMD Athlon 64 X2 dual core processor 4000+ 2.10 GHz, 896 MB of RAM physical address extension. Intensity rates, modeling results and calculation times are in Table 1.

TABLE I   MODELING RESULTS

Parameters			Method	$E(L_1)$	$E(L_2)$	$E(L_3)$	$P(Loss)$	Time, s.
$\lambda_1 = 1$	$\lambda_2 = 0.8$	$\lambda_3 = 0.5$	Numerical modeling	0.0722	0.0922	0.0800	0.0012	78
$\mu_1^{(1)} = 2$	$\mu_2^{(1)} = 3$	$\mu_3^{(1)} = 4$	ARENA	0.0719	0.0925	0.0795	0.0012	316
$\lambda_1 = 2$	$\lambda_2 = 1.6$	$\lambda_2 = 1$	Numerical modeling	0.4108	0.7774	1.0709	0.0764	79
$\mu_1^{(1)} = 2$	$\mu_2^{(1)} = 3$	$\mu_3^{(1)} = 4$	ARENA	0.4097	0.7783	1.0684	0.0763	314

The modeling parameters were chosen freely. The obtained results showed that it requires less calculation time and gives higher accuracy than standard simulation software to model certain queuing systems.

V. CONCLUSION AND FUTURE WORK

This paper presented a new programing tool, which was developed in C++ language code to implement a numerical model. The tool realized the semi-automatic technique for creation of numerical models and analysis of stochastic queuing systems. It generates the set of all possible states and transition matrix of the system under consideration, computes the stationary probabilities and the performance measures of the system according the given formulas. The technique offers advanced functionality compared to existing tools from several points of view: 1) the tool allows solving problems of large scale (with thousands or millions of states); 2) the tool allows creating models of non-markovian queuing systems approximating ones by markovian models.

Problems of large dimension require a large amount of calculations and computer resources. Often it is impossible to solve some problems with available resources. The future work will be devoted for creation a special computation algorithm which allows avoiding mentioned problems.

REFERENCES

[1] A. Sleptchenko, A. Harten, and M. Heijden, "An exact solution for the state probabilities of the multi-class, multi-server queue with preemptive priorities, Queuing systems", Queueing Systems: Theory and Applications, vol. 50, 2005, pp. 81-107.

[2] H. R. Gail, S. L. Hantler, and B. A. Taylor, "On preemptive Markovian queue with multiple servers and two priority classes", Mathematics of Operations research, vol. 17, no. 2, 1992, pp. 365-391.

[3] A. Sleptchenko, I .J. B. F. Adan, and G .J. van Houtum, "Joint queue length distribution of multi-class, single-server queues with preemptive priorities" <http://alexandria.tue.nl/repository/books/581962.pdf> 25.12.2012

[4] M. Harchol-Balter, T. Osogami, A. Scheller-Wolf, and A. Wierman, "Multi-server queueing systems with multiple priority classes Queueing Systems", Theory and Applications, vol. 51, 2005, pp. 331-360.

[5] E. P. C. Kao and S. D. Wilson, "Analysis of non - preemptive priority queues with multiple servers and two priority classes", European Journal of Operational Research, vol. 118, 1999, pp. 181-193.

[6] E. P. C. Kao and K. S. Narayanan, "Computing steady-state probabilities of a nonpreeptive priority multiserver queue", Journal on Computing, vol. 2, no 3, 1990, pp. 211 – 218.

[7] E. Kao and K. Narayanan, "Modeling a multiprocessor system with preemptive priorities", Management Science, vol. 2, 1991, pp.185–97.

[8] A. Bondi and J. Buzen, "The response times of priority classes under preemptive resume in M/G/m queues", In ACM Sigmetrics, Aug. 1984, pp. 195–201.

[9] I. Mitrani and P. King, "Multiprocessor systems with preemptive priorities", Performance Evaluation, vol.1, 1981, pp. 118–125.

[10] T. Nishida, "Approximate analysis for heterogeneous multiprocessor systems with priority jobs", Performance Evaluation, vol. 15, 1992, pp. 77–88.

[11] M. Snipas and E. Valakevicius, "Numerical-analytic model of multi-class, multi-server queue with nonpreemptive priorities", Innovations and Advances in Computer Sciences and Engineering. Dordrecht: Springer, 2010, ISBN 9789048136575, pp. 413-415.

[12] E. Valakevičius, "The application of phase type distributions for modelling queuing systems", Journal of Systemics, Cybernetics and Informatics, vol. 5, no. 6, 2009, pp. 28-32.

[13] W-K. Ching and M. K. Ng, "Markov Chains: Models, Algoritms and Applications", Springer, 2005.

[14] H. C. Tijms, Stochastic Models: An Algorithmic Approach. John Wiley & Sons, 1994.

[15] H. Pranevičius and E. Valakevičius, Numerical Models of Systems Specified by Markovian processes. Kaunas: Technologija, 1996.