

Multiserver Multichannel Real-Time System with Limited Maintenance Facilities under Maximum Load

Edward Ianovsky¹, Joseph Kreimer²

¹Raval ACS Ltd., Beer-Sheva, Israel

²Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, Beersheba, Israel

Email: e.ianovsky@hotmail.com, kreimer@bgu.ac.il

Received 23 June 2015; accepted 21 July 2015; published 24 July 2015

Copyright © 2015 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

We consider a multi server and multichannel real-time system with identical servers (e.g. unmanned aerial vehicles, machine controllers, etc.) that provide services for requests of real-time jobs arriving via several different channels (e.g. surveillance regions, assembly lines, etc.) working under maximum load regime. Each channel has its own constant numbers of jobs inside at any instant. Each channel has its own specifications, and therefore different kinds of equipment and inventory are needed to serve different channels. There is a limited number of identical maintenance teams (less than the total number of servers in the system). We compute analytically steady-state probabilities of this system, its availability, loss penalty function and other performance characteristics, when both service and maintenance times are exponentially distributed.

Keywords

Availability, Performance, Real-Time System, Steady-State

1. Introduction

Real-time systems (RTS) are defined as those for which correctness depends not only on the logical properties of the computed results, but also on their temporal properties. In RTS, a calculation that uses temporally invalid data may be useless and sometimes harmful—even if such a calculation is functionally correct. Examples include industrial automation, traffic control, aerospace, robotic, intelligence and defense system, telecommunication and distributed process control, just to name a few.

Several researchers ([1]-[3]) have proposed a number of classic priority algorithms for scheduling real-time

tasks on a single processor. The problem of minimizing the number of processors in multiprocessor computer system executing real-time tasks is studied in [4].

Different scientific communities are treating RTS problems. During last three decades several meta-heuristic methods, such as Tabu Search [5], Simulated Annealing [6] and Greedy Randomized Adaptive Search Procedure [7] are developed. Good surveys on Artificial Intelligence and real-time decision problems (RTDP) are given in references [8]-[11].

The use of analytical methods of queueing theory [12] and stochastic processes [13] has significant benefits in developing RTS.

We will focus on RTS with a zero deadline for the beginning of job processing. In these RTS, jobs are processed immediately upon arrival, if there are available servers. That part of the job which is not processed immediately is lost forever, since queueing of jobs or their parts are not allowed. The particular interest in such RTS is aroused by military intelligence problems involving unmanned aerial vehicles (UAV), which demonstrate a very high efficiency in many local conflicts. It is proved that the non-mix policy of never relieving an operative server maximizes the availability of a multiserver single-channel RTS involving preventive maintenance and working in general regime with any arrival pattern under consideration and constant services and maintenance times ([14] [15]). This policy appears to be optimal for any finite time interval, and not only for infinite horizon. Multiserver (identical servers) and multichannel (identical channels) RTS [16] [17] (with limited and unlimited number of maintenance facilities respectively), working under maximum load regime were treated as finite source queues. Two-dimensional birth-and-death processes [18] [19] were applied in analysis of a multiserver RTS (with ample and limited number of maintenance teams respectively) with two different channels operating under a maximum load regime, when both service and maintenance times are exponentially distributed. In [20] the results of [18] for $r \geq 2$ different channels operating under a maximum load regime were extended. Optimal assignment probabilities maximizing availability of RTS (with ample and limited number of maintenance teams respectively) with large number of servers and two channels were obtained [21] [22]. Amultiserver and multichannel RTS with identical servers and channels and ample maintenance facilities, working in general regime with exponentially distributed service, maintenance, jobs inter-arrival and duration times were treated as Markov chains [23] in order to obtain various performance characteristics. The researchers [24] [25] obtained moments of RTS with ample and limited maintenance facilities respectively. In [26] RTS with preemptive priorities policy were considered. In [27] the RTS with exactly one job in each channel was studied. The researchers [28] computed analytically (for exponentially distributed service times) and via Cross Entropy [29]-[31] simulation approach (for generally distributed service times) optimal routing probabilities for RTS with ample maintenance facilities.

This work extends the models [27] [28] as follows. The RTS under consideration assumes the constant number of jobs, specific for each channel. We compute analytically steady-state probabilities of this system, its availability, loss penalty function and other performance characteristics.

The paper is organized as follows: in Section 2 describes the model; Section 3 provides steady state probabilities; Section 4 presents various performance measures; Finally, Section 5 summarizes our results.

2. Description of the Model

We consider a multiserver RTS consisting of N identical servers that provide service for requests of real-time jobs arriving via r different channels required to be under nonstop surveillance. There are exactly r_k ($r_k \geq 1$), $k = 1, \dots, r$ requests of real-time jobs in k -th channel at any instant (there are no additional job arrivals to the busy channel), and therefore r_k servers at most are used to serve the i -th channel (with others being on stand-by or providing the service to another channel or in maintenance or waiting for maintenance) at any given time. Thus,

the total number of operating servers in the system is at most $\sum_{k=1}^r r_k$.

Each channel has its own specifications and conditions, etc., and therefore different kinds of equipment and inventory are needed to serve different channels.

A server providing service for the k -th channel is operative for a period of time S_k before requiring R_k hours of maintenance. S_k and R_k are independent exponentially distributed random variables with parameters μ_k ($k = 1, \dots, r$) and λ respectively.

It is assumed that there are K ($K \leq N$) maintenance teams available to repair (with repair times R_k being i.i.d.r.v.) the servers. Thus, a shortage of maintenance facilities is possible. In that case the server waits for

maintenance. This server is assigned to the k -th channel with probability p_k ($k = 1, \dots, r$). It receives the appropriate kind of maintenance (equipment, programming, etc.), and therefore cannot be sent to another channel. Assignment probabilities p_k may depend upon inventory conditions. They also can be used as control parameters. The duration R_k of repair is exponentially distributed with parameter λ , and does not depend on the channel. After maintenance, the server will either be on stand-by or serving the channel it was assigned to.

The system works under a maximum load (worst case) of nonstop data arrival to each one of r channels. This kind of operation is typical in high performance data acquisition and control systems, such as self-guided missiles, space stations, satellites, etc.

If, during some period of time of length T , there is no available server to serve one of the jobs, we will say that the part of the job of length T is lost.

Queues of jobs or their parts cannot exist in RTS, nevertheless they can be fitted into a framework of finite source queues, while using a dual approach of changing the roles between jobs and servers.

3. Steady State Probabilities

Denote: $\lambda_k = \lambda p_k$ and $\rho_k = \lambda_k / \mu_k$, (n_1, n_2, \dots, n_r) the state of the system, where n_k ($k = 1, \dots, r$) is a number of fixed servers assigned to the k -th channel (obviously $\sum_{k=1}^r n_k \leq N$), and p_{n_1, n_2, \dots, n_r} the corresponding

steady state probability. There are $\binom{N+r}{r}$ states in total.

The above RTS can be presented as a closed Jackson queuing network (Figure 1) consisting of N customers (N servers of the RTS), $r + 1$ nodes (r channels and maintenance station of the RTS) with r_k servers at k -th node (r_k jobs in the k -th channel of the RTS) ($k = 1, \dots, r$) and K servers at $r + 1$ -th station (K maintenance teams of the RTS). The network has transition probabilities p_k (assignment probabilities of the RTS) from $r + 1$ -th node to k -th one ($k = 1, \dots, r$), and transition probabilities from k -th station to $r + 1$ -th are equal to 1. Other transition probabilities are equal to 0. Service times at the network stations (operating and maintenance times of the RTS) are exponentially distributed. The customers of the network cannot leave it and cannot come to it from outside. Thus, the network is a closed Jackson network by definition.

From the description of the RTS as a closed Jackson network we obtain ([12] [32]) that the steady-state probabilities p_{n_1, n_2, \dots, n_r} are given by the following:

Theorem 1: A real-time system with N servers, K ($K < N$) maintenance crews, r ($r \geq 2$) different channels operating under a maximum load regime with r_k jobs in k -th channels at any instant, and exponentially

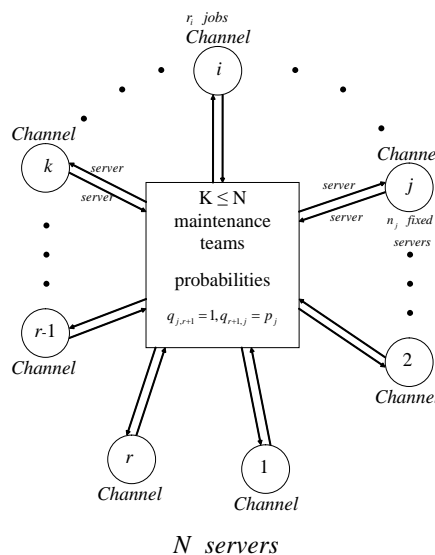


Figure 1. Closed Jackson network.

distributed operating and maintenance times (with parameters μ_k (for the k -th channel ($k = 1, \dots, r$)) and λ respectively) has following values of the steady-state probabilities

$$P_{n_1, n_2, \dots, n_r} = \frac{K^{\min\left(\sum_{i=1}^r n_i, N-K\right)} K!}{\min\left(N - \sum_{i=1}^r n_i, K\right)!} \prod_{j=1}^r \frac{\rho_j^{n_j}}{\min(n_j, r_j)! r_j^{\max(n_j - r_j, 0)}} P_{0, \dots, 0}, \quad (3.1)$$

$$P_{0, \dots, 0} = \left[\sum_{\substack{n_1, \dots, n_r \geq 0 \\ n_1 + \dots + n_r \leq N}} \frac{K^{\min\left(\sum_{i=1}^r n_i, N-K\right)} K!}{\min\left(N - \sum_{i=1}^r n_i, K\right)!} \prod_{j=1}^r \frac{\rho_j^{n_j}}{\min(n_j, r_j)! r_j^{\max(n_j - r_j, 0)}} \right]^{-1}, \quad (3.2)$$

where (n_1, n_2, \dots, n_r) is a state of the RTS, n_k is number of fixed servers of the RTS in the k -th channel and $\rho_k = \lambda_k / \mu_k$ ($k = 1, \dots, r$).

We will use the following notations:

$$\begin{aligned} 1) \quad Q_N^r(\rho_1, \dots, \rho_r) &= \sum_{\substack{n_1, \dots, n_r \geq 0 \\ n_1 + \dots + n_r \leq N}} \frac{K^{\min\left(\sum_{i=1}^r n_i, N-K\right)} K!}{\min\left(N - \sum_{i=1}^r n_i, K\right)!} \prod_{j=1}^r \frac{\rho_j^{n_j}}{\min(n_j, r_j)! r_j^{\max(n_j - r_j, 0)}} \\ &= \sum_{n_1=0}^N \sum_{n_2=0}^{N-n_1} \sum_{n_3=0}^{N-n_1-n_2} \dots \sum_{n_r=0}^{N-\sum_{i=1}^{r-1} n_i} \frac{K^{\min\left(\sum_{i=1}^r n_i, N-K\right)} K!}{\min\left(N - \sum_{i=1}^r n_i, K\right)!} \prod_{j=1}^r \frac{\rho_j^{n_j}}{\min(n_j, r_j)! r_j^{\max(n_j - r_j, 0)}} \end{aligned}$$

and

2) $P_N^{(k)}(n_k, \rho_1, \dots, \rho_r)$ the probability that n_k fixed servers are assigned to the k -th channel ($k = \overline{1, r}$).

Then, using Theorem 1, we obtain:

$$\begin{aligned} P_{0, \dots, 0} &= [Q_N^r(\rho_1, \dots, \rho_r)]^{-1}, \\ P_N^{(k)}(n_k, \rho_1, \dots, \rho_r) &= \frac{\sum_{n_1=0}^{N-n_k} \sum_{n_2=0}^{N-n_k-n_1} \dots \sum_{n_{k-1}=0}^{N-n_k-\sum_{i=1}^{k-2} n_i} \sum_{n_{k+1}=0}^{N-\sum_{i=1}^k n_i} \dots \sum_{n_r=0}^{N-\sum_{i=1}^{r-1} n_i} \frac{K^{\min\left(\sum_{i=1}^r n_i, N-K\right)} K!}{\min\left(N - \sum_{i=1}^r n_i, K\right)!} \prod_{j=1}^r \frac{\rho_j^{n_j}}{\min(n_j, r_j)! r_j^{\max(n_j - r_j, 0)}}}{Q_N^r(\rho_1, \dots, \rho_r)} \\ &= \frac{(K\rho_k)^{n_k}}{\min(n_k, r_k)! r_k^{\max(n_k - r_k, 0)} Q_N^r(\rho_1, \dots, \rho_r)} \\ &= \frac{\sum_{n_1=0}^{N-n_k} \sum_{n_2=0}^{N-n_k-n_1} \dots \sum_{n_{k-1}=0}^{N-n_k-\sum_{i=1}^{k-2} n_i} \sum_{n_{k+1}=0}^{N-\sum_{i=1}^k n_i} \dots \sum_{n_r=0}^{N-\sum_{i=1}^{r-1} n_i} \frac{K^{\min\left(\sum_{i=1, i \neq k}^r n_i, N-n_k-K\right)} K!}{\min\left(N - n_k - \sum_{i=1, i \neq k}^r n_i, K\right)!}}{\min(n_k, r_k)! r_k^{\max(n_k - r_k, 0)} Q_N^r(\rho_1, \dots, \rho_r)} \\ &= \frac{(K\rho_k)^{n_k} Q_{N-n_k}^{r-1}(\rho_1, \dots, \rho_{k-1}, \rho_{k+1}, \dots, \rho_r)}{\min(n_k, r_k)! r_k^{\max(n_k - r_k, 0)} Q_N^r(\rho_1, \dots, \rho_r)} \end{aligned} \quad (3.3)$$

where $n_k = \overline{0, N}$, $k = \overline{1, r}$.

4. Performance Characteristics

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your journal for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

In this section we show how to obtain some useful performance characteristics of the RTS under consideration.

Each server can be in one of following states:

- (i) busy (serving a job of one of the channels);
- (ii) in maintenance;
- (iii) on stand-by (for one of the channels);
- (iv) waiting for maintenance.

Each job of k -th channel ($k = \overline{1, r}$) can be in one of two positions:

- (i) in service;
- (ii) out of service.

Keeping in mind that n_k is a number of fixed servers assigned to the k -th channel ($k = \overline{1, r}$), we can represent the number of channels and servers in different positions in terms of n_k , namely:

Number of fixed servers is $\sum_{k=1}^r n_k$;

Number of fixed servers serving the k -th channel (also a number of jobs served in the k -th channel) is

$$\min(n_k, r_k), \quad k = \overline{1, r};$$

Number of fixed servers on stand-by for the k -th channel is $\max(n_k - r_k, 0)$, $k = \overline{1, r}$;

Number of broken servers is $N - \sum_{k=1}^r n_k$;

Number of broken servers in maintenance is $\min\left(N - \sum_{k=1}^r n_k, K\right)$;

Number of broken servers waiting for maintenance is $\max\left(N - \sum_{k=1}^r n_k - K, 0\right)$.

Now we can obtain corresponding average values (performance characteristics):

$$L_k = E[n_k] = \sum_{n_1=0}^N \sum_{n_2=0}^{N-n_1} \cdots \sum_{n_r=0}^{N-\sum_{j=1}^{r-1} n_j} n_k p_{n_1, \dots, n_r}$$

for average number of fixed servers assigned to the k -th channel, $k = \overline{1, r}$;

$$L = \sum_{k=1}^r L_k$$

for average number of fixed servers in the system;

$$L_{k,b} = E[\min(n_k, r_k)] = \sum_{n_1=0}^N \sum_{n_2=0}^{N-n_1} \cdots \sum_{n_r=0}^{N-\sum_{j=1}^{r-1} n_j} \min(n_k, r_k) p_{n_1, \dots, n_r}$$

for average number of busy servers serving the k -th channel (of jobs served in the k -th channel), $k = \overline{1, r}$;

$$L_{k,ns} = E[\max(r_k - n_k, 0)] = r_k - L_{k,b}, \quad k = \overline{1, r}$$

for average number of nonserved jobs in the k -th channel;

$$L_b = \sum_{k=1}^r L_{k,b}$$

for average number of busy servers in the system (average number of jobs in service);

$N - L$ for average number of broken servers;

$$\begin{aligned} L_m &= E \left[\min \left(N - \sum_{k=1}^r n_k, K \right) \right] = \sum_{n_1=0}^N \sum_{n_2=0}^{N-n_1} \cdots \sum_{n_r=0}^{N-\sum_{j=1}^{r-1} n_j} \min \left(N - \sum_{k=1}^r n_k, K \right) p_{n_1, \dots, n_r} \\ &= K \sum_{n_1=0}^{N-K} \sum_{n_2=0}^{N-K-n_1} \cdots \sum_{n_r=0}^{N-K-\sum_{j=1}^{r-1} n_j} p_{n_1, \dots, n_r} + \sum_{n_1=0}^N \sum_{n_2=0}^{N-n_1} \cdots \sum_{n_r=\max \left(N-K+1-\sum_{j=1}^{r-1} n_j, 0 \right)}^{N-\sum_{j=1}^{r-1} n_j} \left(N - \sum_{k=1}^r n_k \right) p_{n_1, \dots, n_r} \end{aligned}$$

for average number of broken servers in maintenance;

$$p_k L_m$$

for average number of broken servers in maintenance assigned to the k -th channel, $k = 1, \dots, r$;

$$L_w = E \left[\max \left(N - \sum_{k=1}^r n_k - K, 0 \right) \right] = \sum_{n_1=0}^{N-K} \sum_{n_2=0}^{N-K-n_1} \cdots \sum_{n_r=0}^{N-K-\sum_{j=1}^{r-1} n_j} \left(N - \sum_{k=1}^r n_k - K \right) p_{n_1, \dots, n_r}$$

for average number of broken servers waiting for maintenance;

$$L_{k, sb} = E \left[\max (n_k - r_k, 0) \right] = \sum_{n_1=0}^N \sum_{n_2=0}^{N-n_1} \cdots \sum_{n_r=0}^{N-\sum_{j=1}^{r-1} n_j} \max (n_k - r_k, 0) p_{n_1, \dots, n_r}$$

for average number of fixed servers on stand-by assigned to the k -th channel, $k = \overline{1, r}$;

$$L_{sb} = \sum_{k=1}^r L_{k, sb}$$

for average number of fixed servers on stand-by in the system;

$$\lambda_{av} = \lambda L_m$$

for average rate of servers arriving from maintenance;

$$\lambda_{av, k} = \lambda p_k L_m$$

for average rate of servers arriving from maintenance and assigned to the k -th channel, $k = \overline{1, r}$;

$$W_{k, sb} = L_{k, sb} / \lambda_{av, k}$$

for average time spent on stand-by by server assigned to the k -th channel, $k = \overline{1, r}$ (Little Theorem);

$$W_k = W_{k, sb} + 1/\mu_k$$

for average time of server assigned to the k -th channel being fixed, $k = 1, \dots, r$ (Little Theorem).

$$R_N^{(k)}(\rho_1, \dots, \rho_r) = L_{k, b} = \sum_{n=0}^N \min(n, r_k) P_N^{(k)}(n, \rho_1, \dots, \rho_r)$$

is an average number of jobs served in the k -th channel ($k = 1, \dots, r$) at any given moment.

It follows from (3.3) that

$$R_N^{(k)}(\rho_1, \dots, \rho_r) = \sum_{n=0}^N \frac{\min(n, r_k) (K \rho_k)^n}{\min(n, r_k)! r_k^{\max(n-r_k, 0)}} \frac{Q_{N-n}^{r-1}(\rho_1, \dots, \rho_{k-1}, \rho_{k+1}, \dots, \rho_r)}{Q_N^r(\rho_1, \dots, \rho_r)} \quad (4.1)$$

$$L_{k,ns} = E[\max(r_k - n_k, 0)] = r_k - L_{k,b} = r_k - R_N^{(k)}(\rho_1, \dots, \rho_r)$$

for average number of nonserved jobs in the k -th channel, $k = 1, \dots, r$.

The use of RTS relies on the principle of availability. We will introduce therefore several definitions.

Definition 4.1: For a channel (with r_k jobs inside) operating under maximum load regime, the average availability is given by the following formula

$$Av_k = E[\text{number of served jobs in } k\text{-th channel}] / r_k = R_N^{(k)}(\rho_1, \dots, \rho_r) / r_k.$$

Definition 4.2: For a *multichannel* system (number of channels $r \geq 2$) operating under maximum load regime, the average system availability is given by the formula

$$Av_N(\rho_1, \dots, \rho_r) = \sum_{k=1}^r R_N^{(k)}(\rho_1, \dots, \rho_r) / \sum_{k=1}^r r_k.$$

Another important characteristic is an average loss penalty (or operation cost) of system operation in equilibrium during time unit.

Definition 4.3: Let C_k ($k = \overline{1, r}$) be the cost of time unit during which one job in k -th channel is out of service (*penalty*). Then formula

$$TC_N(\rho_1, \dots, \rho_r) = \sum_{k=1}^r C_k (r_k - R_N^{(k)}(\rho_1, \dots, \rho_r))$$

represents average loss penalty function.

We also have the following important relationship between system availability and its loss penalty function, namely for $C_1 = C_2 = \dots = C_r = 1$

$$Av_N(\rho_1, \dots, \rho_r) = 1 - TC_N(\rho_1, \dots, \rho_r) / \sum_{k=1}^r r_k \quad (4.3).$$

5. Conclusion

We presented a multiserver multichannel real-time system working in maximum load regime with the shortage of maintenance teams. The system consists of r different channels with exactly r_k jobs in k -th channel. We obtained analytically the steady state probabilities and various performance measures, providing complete description of this system. These analytical results could be used immediately without long simulations by researchers and practitioners. Our next goal is to find the optimal routing probabilities, maximizing the system availability.

References

- [1] Liu, C.L. and Layland, J.W. (1973) Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. *Journal of the Association for Computing Machinery*, **20**, 46-61.
- [2] Labetoulle, J. (1974) Some Theorems on Real Time Scheduling. In: Gelenbe, E. and Mahl, R., Eds., *Computer Architecture and Networks*, North Holland Publications, New York, 285-298.
- [3] Serlin, O. (1972) Scheduling of Time Critical Processes. *Proceedings of the Spring Joint Computer Conference*, **40**, 925-932.
- [4] Dhal, S.K. and Liu, C.L. (1978) On a Real-Time Scheduling Problem. *Operations Research*, **26**, 127-140. <http://dx.doi.org/10.1287/opre.26.1.127>
- [5] Glover, F.W. and Laguna, M. (1997) *Tabu Search*. Kluwer Academic Publishers, Dordrecht. <http://dx.doi.org/10.1007/978-1-4615-6089-0>
- [6] Kirkpatrick, S., Gelatt Jr., C.D. and Vecchi, M.P. (1983) Optimization by Simulated Annealing. *Science*, **220**, 671-680. <http://dx.doi.org/10.1126/science.220.4598.671>
- [7] Feo, T.A., Venkatraman, K. and Bard, J.F. (1991) A GRASP for a Difficult Single Scheduling Problem. *Computers and Operations Research*, **18**, 635-643. [http://dx.doi.org/10.1016/0305-0548\(91\)90001-8](http://dx.doi.org/10.1016/0305-0548(91)90001-8)
- [8] Garvey, A. and Lesser, V. (1993) A Survey of Research in Deliberative Real-Time Artificial Intelligence. Report, Department of Computer Science, University of Massachusetts, 84-93.

- [9] Strosnider, J.K. and Paul, C.J. (1993) A Structured View of Real-Time Problem Solving. *AI Magazine*, **14**, 45-66.
- [10] Laffey, T.J., *et al.* (1988) Real-Time Knowledge System. *AI Magazine*, **9**, 27-45.
- [11] Seguin, R., Potvin, J.Y., Gendreau, M., Crainic, T.G. and Marcotte, P. (1997) Real-Time Decision Problems: An Operational Research Perspective. *Journal of the Operational Research Society*, **48**, 162-174.
<http://dx.doi.org/10.1057/palgrave.jors.2600341>
- [12] Gross, D., Shortle, J.F., Thompson, J.M. and Harris, C.M. (2008) Fundamentals of Queueing Theory. 4th Edition, John Wiley, New York. <http://dx.doi.org/10.1002/9781118625651>
- [13] Karlin, S. and Taylor, H.M. (1975) A First Course in Stochastic Processes. 2nd Edition, Academic Press, New York.
- [14] Kreimer, J. and Mehrez, A. (1993) An Optimal Operation Policy for Real-Time N -Server Stand-By System Involving Preventive Maintenance. *European Journal of Operational Research*, **69**, 50-54.
[http://dx.doi.org/10.1016/0377-2217\(93\)90089-6](http://dx.doi.org/10.1016/0377-2217(93)90089-6)
- [15] Kreimer, J. and Mehrez, A. (1994) Optimal Real-Time Data Acquisition and Processing by a Multiserver Stand-By System. *Operations Research*, **42**, 24-30. <http://dx.doi.org/10.1287/opre.42.1.24>
- [16] Kreimer, J. and Mehrez, A. (1998) Computation of Availability of a Real-Time System Using Queueing Theory Methodology. *Journal of the Operational Research Society*, **49**, 1095-1100.
<http://dx.doi.org/10.1057/palgrave.jors.2600610>
- [17] Kreimer, J. (1999) Real-Time Multiserver and Multichannel Systems with Shortage of Maintenance Crews. *Mathematical and Computer Modelling*, **30**, 169-176. [http://dx.doi.org/10.1016/S0895-7177\(99\)00206-X](http://dx.doi.org/10.1016/S0895-7177(99)00206-X)
- [18] Kreimer, J. (1999) Performance of Real-Time Multiserver System with Two Different Channels in Equilibrium. *Communications in Dependability and Quality Management*, **2**, 16-23.
- [19] Kreimer, J. (2000) Real-Time Multiserver System with Two Non-identical Channels and Limited Maintenance Facilities. *Mathematics and Computers in Simulation*, **53**, 85-94. [http://dx.doi.org/10.1016/S0378-4754\(00\)00171-3](http://dx.doi.org/10.1016/S0378-4754(00)00171-3)
- [20] Kreimer, J. (2002) Real-Time System with Homogeneous Servers and Nonidentical Channels in Steady State. *Computers and Operations Research*, **29**, 1465-1473.
- [21] Ianovsky, E. and Kreimer, J. (2001) Optimization of Real-Time Multiserver System with Two Different Channels. *Communications in Dependability and Quality Management*, **4**, 16-23.
- [22] Ianovsky, E. and Kreimer, J. (2003) Optimization of Real-Time Multiserver System with Two Different Channels and Shortage of Maintenance Facilities. *Mathematics and Computers in Simulation*, **63**, 615-627.
[http://dx.doi.org/10.1016/S0378-4754\(03\)00092-2](http://dx.doi.org/10.1016/S0378-4754(03)00092-2)
- [23] Kreimer, J. (2002) Effectiveness Analysis of Real-Time Data Acquisition and Processing Multichannel Systems. *IEEE Transactions on Reliability*, **51**, 91-99. <http://dx.doi.org/10.1109/24.994922>
- [24] Ianovsky, E. and Kreimer, J. (2005) Moments of Real-Time Systems with Ample Maintenance Facilities. *Communications in Dependability and Quality Management*, **8**, 15-26.
- [25] Ianovsky, E. and Kreimer, J. (2007) Moments of Real-Time Systems with Shortage of Maintenance Teams. *Communications in Dependability and Quality Management*, **10**, 87-98.
- [26] Bassan, E. and Kreimer, J. (2008) Multiserver and Multichannel Real-Time Systems with Separate Queues and Preemptive Priorities. *Computer Modelling and New Technologies*, **12**, 7-15.
- [27] Ianovsky, E. and Kreimer, J. (2009) Multiserver Real-time System with Shortage of Maintenance Teams. *Communications in Dependability and Quality Management*, **12**, 5-18.
- [28] Ianovsky, E. and Kreimer, J. (2011) An Optimal Routing Policy for Unmanned Aerial Vehicles (Analytical and Cross-Entropy Simulation Approach). *Annals of Operations Research*, **189**, 215-253.
<http://dx.doi.org/10.1007/s10479-009-0609-1>
- [29] Rubinstein, R.Y. and Kroese, D.P. (2008) Simulation and the Monte Carlo Method. 2nd Edition, John Wiley & Sons, New York.
- [30] Rubinstein, R.Y., Ridder, A. and Vaisman, R. (2014) Fast Sequential Monte Carlo Methods for Counting and Optimization. John Wiley & Sons, New York.
- [31] Kroese, D.P., Taimre, T. and Botev, Z.I. (2011) Handbook of Monte Carlo Methods. John Wiley & Sons, New York.
<http://dx.doi.org/10.1002/9781118014967>
- [32] Ianovsky, E. (2005) Analysis and Optimization of Real-Time Systems. Ph.D. Thesis, Ben-Gurion University of the Negev, Beer-Sheva.