The Timeliness of the Reserved Service in the Cluster with the Regulation of the Time of Destruction of Overdue Requests in the Node Queues

Vladimir Bogatyrev [1,2], Stanislav Bogatyrev [1,3] and Anatoly Bogatyrev [3]

[1] *ITMO University, Kronverksky Pr. 49, bldg. A, Saint-Petersburg, 197101, Russia*
[2] *Saint-Petersburg State University of Aerospace Instrumentation, 67, Bolshaya Morskaia str. St Petersburg, Russia*
[3] *JSC NEO Saint Petersburg Competence Center, 1-Ya Sovetskaya, house 6 str. St. Petersburg, Russia*

Abstract

With the increasing complexity of distributed control tasks based on their intellectualization, there are problems of insufficient time and computing resources for functioning in real time. In this regard, there is a need to develop methods for organizing distributed real-time computer systems, based on the consolidation of distributed computing resources with their integration into clusters. The possibilities of increasing the probability of timely servicing of waiting-critical requests in the cluster as a result of query replication and controlling the time of destruction of potentially expired replicas in node queues are investigated. The cluster is represented as a group of queuing systems with infinite queues with a limited average waiting time. The effectiveness of the reserved service of a real-time request is determined by the probability of executing at least one of the generated copies of the request in the maximum allowable time without losing it due to errors and waiting time limits in the queues of cluster nodes. It is shown that there is an optimal multiplicity of query replication with a significant influence of the choice of restrictions on the waiting time for requests in queues before they are destroyed.

Keywords

Replication, redundant service, timeliness, real-time, cluster

1. Introduction

One of the key trends in the development of modern automated control systems is their intellectualization and construction on the basis of distributed computer systems for storing, transmitting and processing data. Ensuring the high quality of distributed management is based on the concept of multi-agent systems. In multi-agent systems, decision-making in changing operating conditions involves the interaction of agents characterized by autonomy, intelligence, purposefulness and activity of behavior. The strategy of interaction of agents for making coordinated decisions, especially when solving difficult-to-formalize tasks, is currently increasingly based on neural network technology [1-3].

With the increasing complexity of automated control tasks, there are problems of insufficient time and computing resources for the functioning of a neural network in real time. In this regard, there is a need to develop methods for organizing distributed computer systems in real time, based on the consolidation of distributed computing resources with their integration into clusters [4-6].

For distributed real-time computer systems, it is fundamental to support the reliability and timeliness of the computing process, which can be considered as a key condition for their operability. [7- 10].

The reliability and timeliness of executing requests that are critical to delays in computer systems and networks can be increased as a result of increasing the computing resources of their consolidation

GraphiCon 2021: 31st International Conference on Computer Graphics and Vision, September 27-30, 2021, Nizhny Novgorod, Russia
EMAIL: vabogatyrev@corp.ifmo.ru (V. Bogatyrev); stanislav@nspcc.ru (S. Bogatyrev); anatoly@nspcc.ru (A. Bogatyrev)
ORCID: 0000-0003-0213-0223 (V. Bogatyrev); 0000-0003-0836-8515 (S. Bogatyrev); 0000-0001-5447-7275 (A. Bogatyrev)

and load balancing and priority distribution. Additional opportunities to increase the reliability and probability of timely execution of requests in real time are provided by their replication with redundant service [11-13]. Real-time redundant service is successful when at least one replica (copy) of the request is completed in a time less than the established limits [11-14]. The organization of redundant service in the cluster, of course, allows you to increase the reliability of calculations, but it is associated with the need to resolve a technical contradiction. Indeed, replication (copying) of requests executed in different nodes of the cluster can lead to both a decrease in the time until the first result is received, and to its increase due to an increase in the total load of the cluster.

The efficiency and expediency of redundant query execution in a cluster represented by a group of queuing systems with infinite queues is shown in [11-15]. The possibilities of combining the reserved service of waiting-critical requests with the regulation of the number of places in the queues of cluster system nodes are analyzed in [16]. The effect analyzed in [16] is achieved as a result of reducing the load achieved by destroying requests received when the queue length exceeds the established limits. In this case, the queue increases above the set limit when the system load increases. The boundary length of the queue at which requests are not accepted for service at the node is set based on maximizing the probability of timely servicing at least one of the copies of requests. At the same time, [16] shows the existence of an optimal multiplicity of reservation requests and an adjustable queue length, at which the maximum probability of timely error-free service of requests is achieved. Potentially, the effect of increasing the probability of timely service can also be achieved by combining the reservation of requests with the regulation of the average waiting time for requests before they are destroyed in queues. Moreover, the specified average time before the destruction of requests in queues may in principle not coincide with the maximum allowable time t_0 of waiting for requests in queues, set based on the requirements of application tasks performed in real time.

The purpose of the work is to study the possibilities of increasing the probability of timely servicing of waiting-critical requests in the cluster as a result of query replication and regulating the time of destruction of potentially expired replicas in node queues.

The effectiveness of the reserved service of a real-time request is determined by the probability of executing at least one of the generated copies of the request in the maximum allowable time without losing it due to errors and waiting time restrictions in the queues of cluster nodes.

2. The probability of servicing requests in the cluster when adjusting the time of destruction of expired replicas of requests

A cluster of *m* nodes is represented as a group of *m* queuing systems (QMS) with infinite queues with a limit on the average waiting time *t* (QMS with impatient customers) [17-19]. If the waiting time for a request in the queue exceeds the set time, it leaves the system without maintenance (it is destroyed). It is assumed that the time of destruction of overdue requests has an exponential distribution with an average t. Note that individual requests can be in the queue for both more and less time *t*. It should also be noted that the time t in general does not coincide with the maximum allowable waiting time t_0, which is set based on the requirements for solving real-time application problems.

Let's consider the service process in some one of the m n-channel QMS with a waiting time limit [17-19]. In each of the *m* QMS, a decrease in the number of requests occurs as a result of either the completion of their service with an intensity of $\mu=1/v$, or as a result of their leaving the queue with an intensity of $\tau=1/t$. At the same time, *v* is the average time of their execution,

The transition from the state with *k* requests in the S_k system to the state with *k*-1 requests S_{k-1} is performed at. k<n, with intensity $\lambda_{k,k-1}=k\mu$. If there are *r* applications in the queue ($k=n+r$), then the transition from the S_k state to S_{k-1} occurs either when one of the *n* applications is completed, or one of the *r* applications is destroyed in the queue. For the studied QMS, the average queue length is calculated as [17-19]

$$L = \frac{\alpha^n P_0}{n!} \sum_{i=1}^{\infty} i\alpha^i \left[\prod_{j=1}^{i} (n + j\beta)^{-1} \right],$$

where

$$\beta = \tau/\mu,$$

$$P_0 = \left[\sum_{k=0}^{n} \frac{\alpha^k}{k!} + \frac{\alpha^n}{n!} \sum_{i=1}^{\infty} \alpha^i \left[\prod_{j=1}^{i} (n + j\beta)^{-1} \right] \right]^{-1}, \quad (1)$$

At the same time, the loading of one of the *m* QMS of the group is carried out. Without replication of requests, loading one of the *m* nodes of the cluster, and when creating *K* replicas of each request, while λ is the intensity of the input stream of requests to the cluster, while α is loading one of the m QMS of the group. Without query replication, loading one of the m nodes of the cluster, and when creating *K* replicas of each request, while λ is the intensity of the input stream of requests to the cluster

$$\beta = \tau/\mu,\ \alpha = \lambda K/\mu m.$$

Each of the *L* requests in the node queue can leave it with an intensity of τ=1/*t*, that is, on average, *Lv* requests leave the queue per unit of time. Thus, the absolute and relative throughput of a cluster node is calculated as:

$$q_a = \lambda - \tau L,$$

$$q = q_a / \lambda = 1 - \tau L / \lambda = 1 - \frac{\beta L}{\alpha}. \quad (2)$$

The probability of a replica being denied service by a cluster node is calculated as:

$$Q = 1 - q = \frac{\beta L}{\alpha},$$

and the probability of serving a request without destroying it in the queue is calculated as

$$p = q = \frac{\beta L}{\alpha}.$$

The probability of executing (without destroying in queues) at least one copy of the reserved request in the cluster is calculated as:

$$P = 1 - Q^K.$$

The dependence of the probability of servicing at least one of the reserved requests in the cluster on the intensity of their receipt when setting the request waiting limits is shown in Figure 1. The multiplicity of replication 1, 2, 3 corresponds to curves 1, 2, 3 with an average waiting time to destruction in queues *t*=1 s, and curves 4, 5, 6 with *t*=5 s. The calculation is performed for a cluster of *m*=8 nodes represented by single-channel QMS (*n*=1) with an unlimited queue at *v*=0.1 s. The figure shows the expediency of setting the multiplicity of query replication depending on the intensity of queries λ. Thus, for the variants presented by the graphs, as λ increases, it is advisable to consistently switch the replication multiplicity *K*=3, 2, 1. The presented graphs show the significance of the impact on the probability of servicing requests of choosing the waiting time in the queue before destroying the replicas of requests. The dependence of the probability of servicing at least one of the reserved requests in the cluster on the multiplicity of request replication is shown in Figure 2. In the figure of the intensity of request receipt λ=20, 30.25 1/s, curves 1, 2 ,3 at *t*=5 s and curves 4, 5, 6 at *t*=2 s correspond to the curves. Fig. 2 shows the existence of an optimal multiplicity of query replication with a significant influence of the choice of restrictions on the waiting time for requests in queues before they are destroyed.

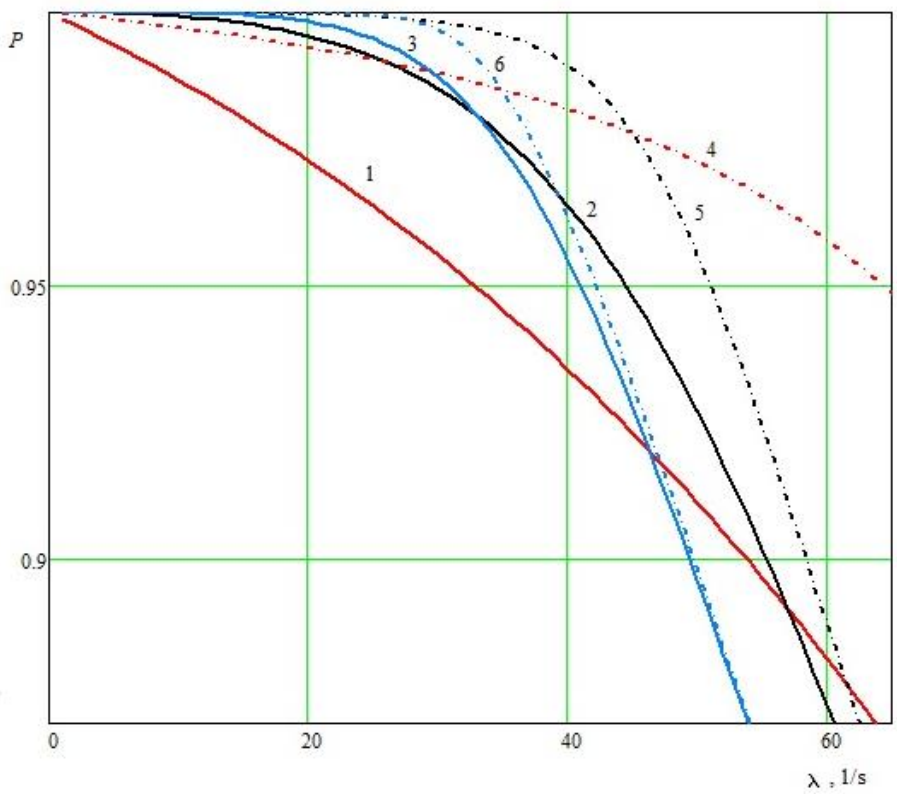

Figure 1: The dependence of the probability of servicing at least one of the reserved requests in the cluster on the intensity of their receipt when setting request waiting limits

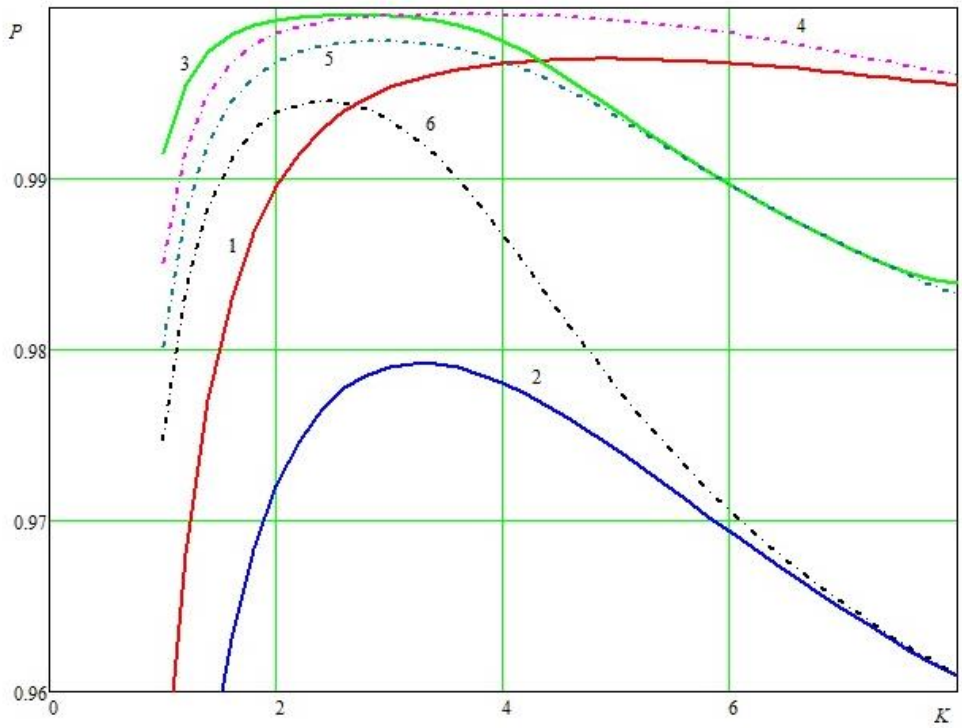

Figure 2: The dependence of the probability of servicing requests by the cluster on the multiplicity of their replication

3. The probability of timely maintenance when replicating requests with the regulation of waiting time for replicas in queue

Let's analyze the effect of limiting the average waiting time t in the queue before destroying requests on the probability of servicing them for a time less than the maximum allowable value t_0. The probability of not exceeding the waiting time of the maximum allowable waiting time value t_0 in each of the m CMOS representing the cluster node is calculated based on a modification of the well-known formula [17-19] for systems with a limited waiting time for requests t is calculated as:

$$F(t_0) = 1 - \frac{1}{q}\sum_{k=n}^{\infty}\frac{\alpha^k P_0}{k!}e^{-(n\mu+(k-n+1)\frac{1}{t})t_0}\sum_{j=0}^{k-n}\frac{\left[n\mu+(k-n+1)\frac{1}{t}t_0\right]^j}{j!},$$

where q and P_0 are determined by the formulas (1) and (2).

The probability of timely maintenance when the waiting time is less than t_0 with the destruction of requests in the cluster node queue from the waiting time limit t is defined as

$$F_1 = qF(t_0).$$

In case of redundant servicing of requests with the formation of K replicas and the conditions for timely execution of at least one replica in at least one of the m nodes of the cluster, it is calculated as:

$$F_2 = 1-(1-F_1)^K.$$

To evaluate the effectiveness of regulating the waiting time limits in nodes t, as a prototype, we consider a cluster in whose nodes the destruction of requests does not occur. When representing the cluster nodes of the prototype CFR type M/M/1 [20, 21], the probability of redundant request service during the waiting time less than the maximum allowable value t_0 in a cluster with the formation of K replicas of requests is calculated as:

$$F_3 = 1-\left[\frac{\lambda K v}{m}e^{(\frac{\lambda K}{m}-\frac{1}{v})t_0}\right]^K.$$

Let's first consider a cluster without reserving requests. The dependence of the probability of timely servicing (for a time less than t_0=0.5 s) of requests in the cluster on the intensity of their receipt when setting the request waiting limits t= 0.5, 1, 2, 5 s is represented by curves 1, 2, 3, 4 in Fig. 3. Curve 5 represents the case without limiting the request waiting time in the cluster nodes. The calculation is performed for a cluster containing m=8 computer nodes represented by single-channel QMS (n=1) with an unlimited queue at v=0.1 s.

The presented dependencies show the effectiveness of reserving requests with a significant impact on the probability of timely servicing requests by choosing the waiting time in the queue before destroying the replicas of requests.

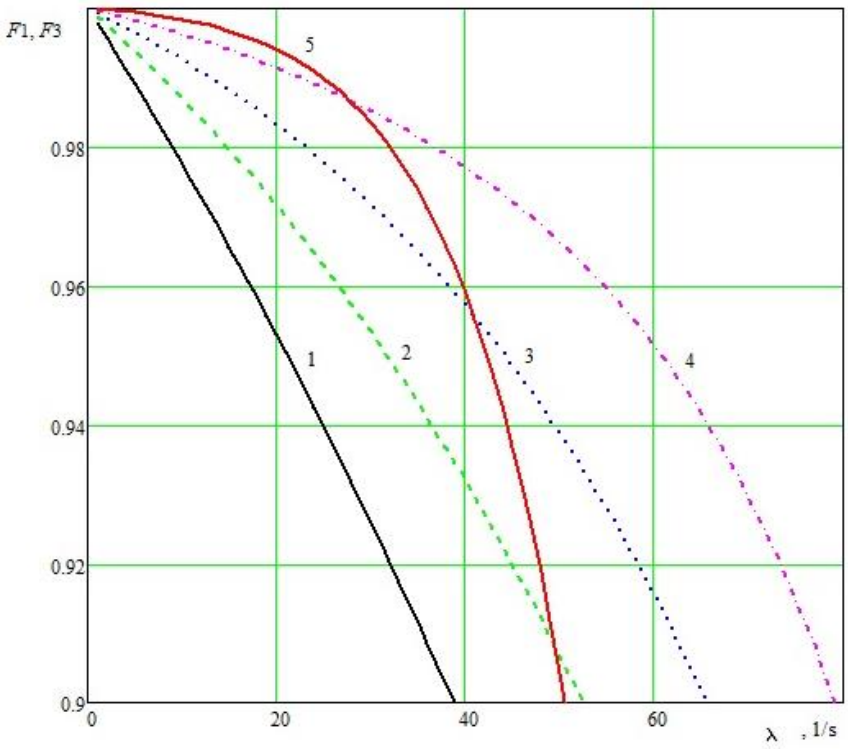

Figure 3: Dependence of the probability of timely unserved servicing of requests in the cluster on the intensity of their receipt

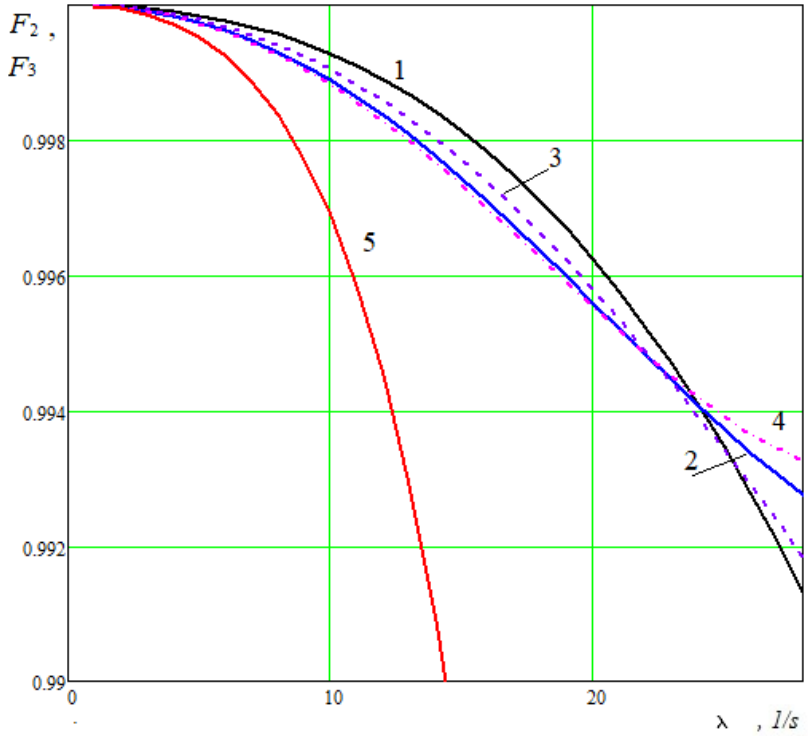

Figure 4: The dependence of the probability of timely duplicated servicing of requests in the cluster on the intensity of their receipt

The conducted studies confirm the possibility of controlling the probability of timely error-free service of requests as a result of regulating the multiplicity of replication and the time of destruction of potentially expired replicas in node queues.

The proposed models and technical solutions for ensuring the reliability and timeliness of redundant services that are critical to queue delays are supposed to be adapted for use within the framework of the concept of Ultrareliable and Low-Latency Wireless Communication [22-24] for organizing communication in distributed control systems.

4. Conclusion

Analytical models are proposed and the possibilities of increasing the probability of timely servicing of waiting-critical requests in a real-time cluster as a result of query replication and controlling the time of destruction of potentially expired replicas in node queues are shown.

The influence of the regulation of limiting the average waiting time t in the queue before the destruction of requests on the probability of their timely service for a time less than the maximum allowable value t0 is shown.

It is shown that there is an optimal multiplicity of replication of requests with a significant impact on the probability of their timely maintenance by choosing restrictions on the waiting time for requests in queues before they are destroyed.

5. References

[1] K. Valogianni, W. Ketter, J. Collins Multiagent Approach to Variable-Rate Electric Vehicle Charging Coordination. Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (Istanbul, Turkey, May, 4-8, 2015). – NewYork: ACM, 2015. – P. 1131-1139.

[2] M. Lanctot, A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning / Advances in Neural Information Processing Systems. – 2017. – P. 4190-4203.

[3] S. Samarasinghe, Neural Networks for Applied Sciences and engineering: from Fundamentals to Complex Pattern Recognition. Boston: Auerbach publications, 2016. – 570 p.

[4] S. Kim, Y. Choi, Constraint-aware VM placement in heterogeneous computing clusters. Cluster Comput 23, 71–85 (2020). doi.org/10.1007/s10586-019-02966-6.

[5] M. Siddiqi1, H. Yu, J. Joung, 5G Ultra-Reliable Low-Latency Communication Implementation Challenges and Operational Issues with IoT Devices Electronics 2019, 8, 981; www.mdpi.com/journal/electronics. doi:10.3390/electronics8090981.

[6] I. Koren, Fault tolerant systems. Morgan Kaufmann publications, San Francisco 2009 378 p.

[7] H. Aysan, Fault-tolerance strategies and probabilistic guarantees for real-time systems Mälardalen University, Västerås, Sweden. 2012. 190 p.

[8] T. Astakhova, N.Verzun, M. Kolbanev, A. Shamin, A model for estimatingenergy consumption seen when nodes of ubiquitous sensor networks communicate information to each other. In Proceedings of the 10th Majorov International Conference on Software Engineering and Computer Systems, Saint Petersburg, Russia, December 20-21 (2018).

[9] D.A. Zakoldaev, A.G. Korobeynikov, A.V. Shukalov, I.O. Zharinov, O.O. Zharinov, Industry 4.0 vs Industry 3.0: the role of personnel in production. IOP Conference Series: Materials Science and Engineering, 2020, Vol. 734, No. 1, pp. 012048. doi 10.1088/1757-899X/734/1/012048.

[10] S.B. Ya, T.M. Tatarnikova, E.D. Poymanova, Organization of multi-level data storage (2019) Informatsionno-Upravliaiushchie Sistemy, 2019 (2), pp. 68-75. doi: 10.31799/1684-8853-2019-2-68-75.

[11] V.A. Bogatyrev, A.V. Bogatyrev, S.V. Bogatyrev, Redundant Servicing of a Flow of Heterogeneous Requests Critical to the Total Waiting Time During the Multi-path Passage of a Sequence of Info-Communication Nodes. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2020. Vol. 12563. pp. 100-112. doi 10.1007/978-3-030-66471-89.

[12] V.A. Bogatyrev, A.V. Bogatyrev, S.V. Bogatyrev, Multipath Redundant Transmission with Packet Segmentation. Wave Electronics and its Application in Information and Telecommunication Systems (WECONF 2019). 2019. pp. 8840643. doi:10.1109/WECONF.2019.8840643.
[13] S.A Arustamov, V.A. Bogatyrev, V.I. Polyakov, Back up data transmission in real-time duplicated computer systems / Advances in Intelligent Systems and Computing, 2016, 451, pp. 103–109. doi: 10.1007/978-3-319-33816-3_11.
[14] V.A. Bogatyrev, A.V. Bogatyrev, S.V. Bogatyrev, The probability of timeliness of a fully connected exchange in a redundant real-time communication system. Wave Electronics and its Application in Information and Telecommunication Systems (WECONF 2020). https://ieeexplore.ieee.org/document/9131517. doi:10.1109/WECONF48837.2020.9131517.
[15] V.A. Bogatyrev, S.V. Bogatyrev, A.N. Derkach, Timeliness of the Reserved Maintenance by Duplicated Computers of Heterogeneous Delay-Critical Stream. CEUR Workshop Proceedings. 2019. Vol. 2522. pp. 26-36.
[16] V.A. Bogatyrev, S.V. Bogatyrev, A.V. Bogatyrev, Redundant multi-path service of a flow heterogeneous in delay criticality with defined node passage paths. Journal of Physics: Conference Series, Volume 1864, 13th Multiconference on Control Problems (MCCP 2020) 6-8 October 2020, Saint Petersburg, Russia 2021 J. Phys.: Conf. Ser. 1864 012094 - 2021, Vol. 1864, 012094, No. 1, pp. 012094. doi 10.1088/1742-6596/1864/1/012094.
[17] L.A. Ovcharov, Applied problems of the theory of queuing. - M.: Mechanical Engineering, 1969 .-- 324
[18] E. S. Wentzel, Operations research M.: Soviet Radio, 1972. - 552 p
[19] A. P. Kirpichnikov, Methods of the applied theory of queuing. 2018. 224 p. ISBN 978-5-9710-4916-6.
[20] L. Kleinrock, Queueing Systems: Volume I. Theory. New York: Wiley Interscience.1975 p. 417. ISBN 978-0471491101.
[21] L. Kleinrock, Queueing Systems: Volume II. Computer Applications. New York:Wiley Interscience. 1976 p. 576. ISBN 978-0471491118.
[22] M. Bennis, M. Debbah, H.V. Poor, Ultrareliable and Low-Latency Wireless Communication: Tail, Risk and Scale. Proc. IEEE 2018, *106*, 1834–1853. doi: 10.1109/JPROC.2018.2867029.
[23] H.Ji,; S. Park, J. Yeo, Y. Kim, J. Lee, B. Shim, Ultra-Reliable and Low-Latency Communications in 5G Downlink: Physical Layer Aspects. IEEE Wirel. Commun. 2018, *25*, 124–130. doi :10.1109/MWC.2018.1700294.
[24] J. Sachs, G. Wikström, T. Dudda, R. Baldemair, K. Kittichokechai, 5G Radio Network Design for Ultra-Reliable Low-Latency Communication. *IEEE Netw.* 2018, *32*, 24–31. doi:10.1109/MNET.2018.1700232.