Journal of Mathematical Sciences, Vol. 237, No. 8, March, 2019

COMPARISON OF TWO OPERATION MODES OF FINITE-SOURCE RETRIAL QUEUEING SYSTEMS WITH COLLISIONS AND A NON-RELIABLE SERVER BY USING SIMULATION

Á. Tóth[1] , T. Bérczes[1] , J. Sztrik[1] , and A. Kuki[1]

In this paper a finite-source retrial queuing system with collision of the customers is investigated by means of computer simulation. The server is not reliable; it is subjected to breakdowns, and the repairs depend on whether the state of the server is idle. The random variables used in the model are jointly independent and are exponential and gamma distributed. Two operation modes are considered in the case of busy breakdown. The novelty of the investigation is a comparison of the performance measures of these modes, and estimations obtained by the simulation are graphically illustrated showing the influence of the difference of the working modes on the performance measures such as mean and variance of the response time, mean and variance of the number of customers in the system, mean and variance of the sojourn time in the orbit, mean and variance of time time a customer spent in service.

1. Introduction

For modeling and investigating real life situations arising in the telecommunication systems such as telephone switching systems, call centers, computer networks, and computer systems, the retrial queues are effective and commonly used tools. It is important to keep in mind that in many practical situations a reverse behavior can be experienced between the rate of generation of new calls and the number of customers in the system. An increasing number of customers will cause a decreasing rate of call generations. This result can be demonstrated by means of finite-source or quasi-random input models. The models of quasi-random input in retrial queues were recently considered for the description of cellular mobile communication networks, computer networks, and local-area networks with random access protocols, with multiple-access protocols, and smart city networks. See, for example, [3, 8].

In real life these types of systems usually cannot be assumed reliable. Thus it is very important to investigate the retrial queueing systems with random server breakdowns and repairs. The nonreliable operation of the systems has a great influence on system characteristics and performance measures. Finite-source retrial queues with server breakdowns have been investigated in several recent papers; see, for example, [2, 5, 7, 13, 14].

In a communication session where there are only limited number of communication channels or other facilities the users (sources) usually fight for these resources. In many cases there is a significant possibility of a conflict. Several sources launching uncoordinated attempts can produce collisions leading to the loss of the transmission and, consequently, the necessity for retransmission. It is very important to build up efficient procedures to prevent the conflict and corresponding message delay. Retrial queues with collision were recently studied in [1, 4, 9–12].

The aim of the present paper is to investigate a system including a nonreliable server with finite number of sources where collisions can take place. In this paper we develop simulation models using SimPack, a collection of C/C++ libraries and executable programs for computer simulation [6], to obtain an estimate for the desired performance measures. In this collection various algorithms are supported, connected with simulation including discrete event simulation, continuous simulation, and combined

[1] University of Debrecen, Debrecen, Hungary, e-mail: adamtoth102@gmail.com, sztrik.janos@inf.unideb.hu, berczes.tamas@inf.unideb.hu, kuki.attila@inf.unideb.hu

Proceedings of the XXXIV International Seminar on Stability Problems for Stochastic Models, Debrecen, Hungary.

(multi-model) simulation. The aim of this work is to provide two different operation modes of the server breaking down in the busy mode.

2. The model of the system

A finite source retrial queueing system is considered, where the number of sources is denoted by N. Each source can generate requests with rate λ/N, that is, the distribution of the source time is exponential with parameter λ/N. There is a queue for waiting; thus, in the case of an idle server, the incoming customer enters into service instantly. The service times are supposed to be gamma distributed with parameters α and β. If the server is busy, that is, a request is under service, an arriving customer, either from the orbit or from the source, will cause a collision with the customer under service, and both requests are directed into the orbit. The customers in the orbit are able to retry reaching the server after an exponentially distributed time with parameter σ/N. The server is nonreliable, so it is supposed to break down after an exponentially distributed time interval. In the case of a busy server the parameter is γ_0, otherwise (in the case of an idle server) it is γ_1. The repair process starts immediately upon the breakdown. The repair time is also an exponentially distributed random variable with parameter γ_2. It is supposed that when the server is unavailable, every source is eligible to generate a customer and sends it to the unit, and these requests are forwarded to the orbit. The customers from the orbit may also retry to reach the server. In the case where a busy server breaks down, two operation modes are considered:

- the interrupted request gets into the orbit instantaneously;
- the service of the interrupted request is suspended and it continues after the server is repaired.

When the submission (the service of a request) is successful, the request goes back to the source. All the random variables involved in the model construction are assumed to be jointly independent.

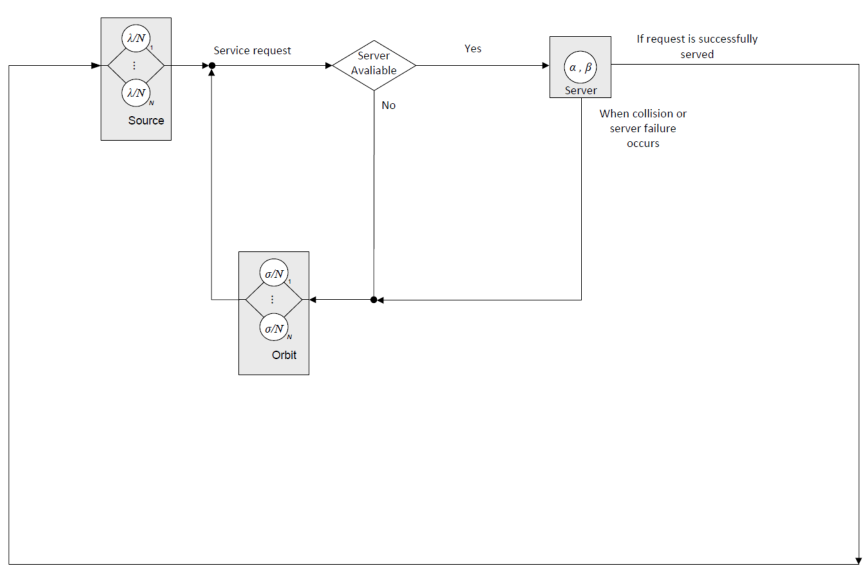

Fig. 1. System model.

3. Simulation results

3.1. Scenario A

The following table shows the input parameters of Scenario A (see Table 1).

Table 1. Numerical values of model parameters

Case	N	λ/N	γ_0	γ_1	γ_2	σ/N	α	β
1	100	0.01	0.1	0.1	1	0.01	0.5	0.5
2	100	0.01	0.1	0.1	1	0.01	1	1
3	100	0.01	0.1	0.1	1	0.01	2	2

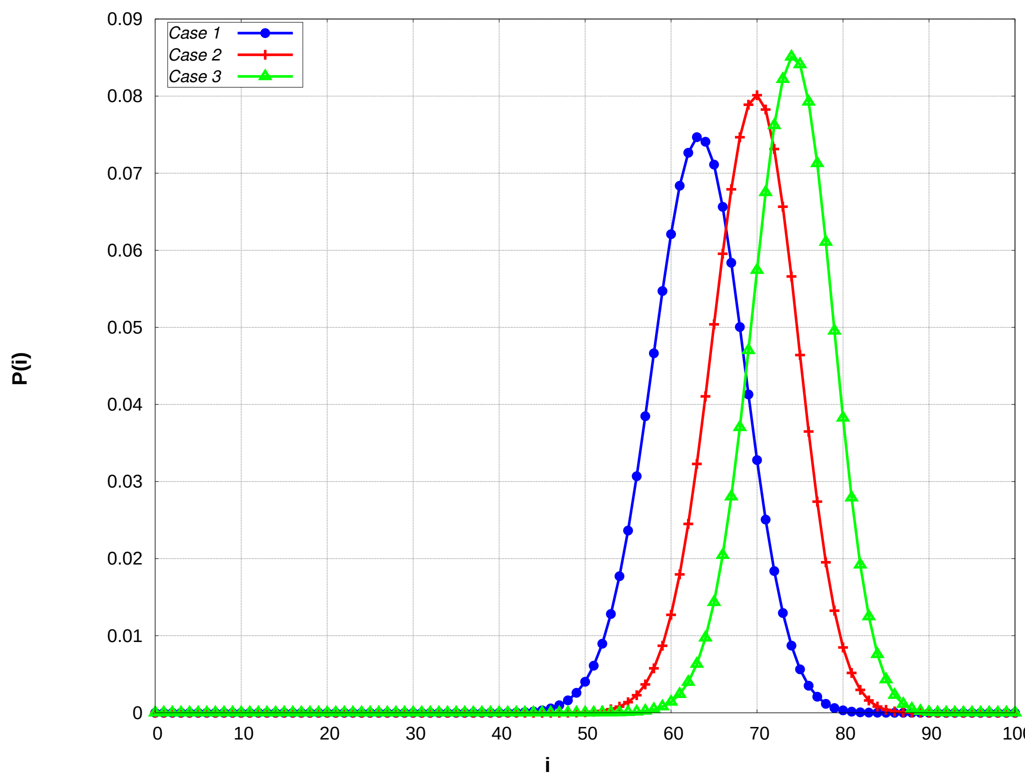

Fig. 2. Comparison of steady-state distributions.

Figure 2 shows the steady-state distribution of the cases under consideration. *Case 2* is a special case because it yields the exponential distribution, when the parameter α is equal to 1. The mean number of customers increases with the increment of parameters α and β and the curves of the steady-state distribution follow the normal law. The following table presents the considered performance measures in relation with the different cases (see Table 2).

In Table 2 the following notations is used: $E(NS)$ and $Var(NS)$ are the mean number and variance of the number of customers, $E(T)$ and $Var(T)$ are the mean and variance of the response time, $E(W)$ and $Var(W)$ are the mean and variance of the waiting time, and $E(S)$ and $Var(S)$ are the mean and variance of the successful service time.

Table 2. Numerical results of Scenario A

Case	$E(NS)$	$Var(NS)$	$E(T)$	$Var(T)$	$E(W)$	$Var(W)$	$E(S)$	$Var(S)$
1	63.0526	28.3198	170.5618	63092.8264	169.7553	62855.9266	0.3357	0.2255
2	69.6114	24.6949	228.9834	97974.6274	227.8834	97613.9701	0.5022	0.2523
3	73.9099	21.9244	283.1452	136396.9409	281.7728	135909.2373	0.6688	0.2238

The following three figures (Figs. 3,4, and 5) compare the mean waiting time of the two different operation modes in the *Cases* under consideration. Operation mode No. 1 corresponds to the mode when the interrupted requests get into the orbit instantaneously, and under operation mode No. 2 we consider the mode when the service of the interrupted request is suspended, and it continues after the server is repaired. In all the cases the results confirm the expectation that the operation mode No. 2 results in a lower mean waiting time. When the values of parameters α and β are higher, the difference between the modes is higher as well. With increase in the arrival intensity, we should expect longer waiting times; however, after λ/N reaches 0.15, it begins to decrease monotonically, which is an interesting phenomenon.

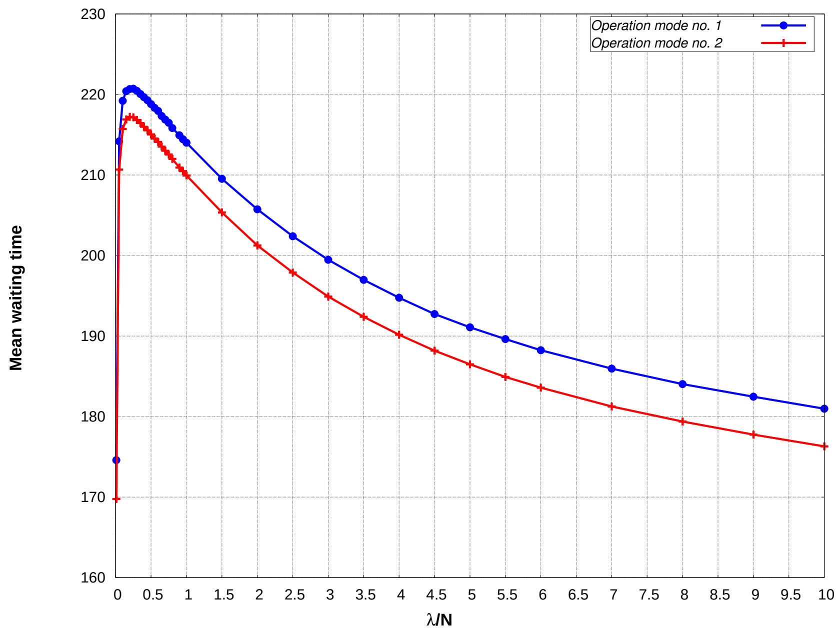

Fig. 3. Mean waiting time vs. intensity of incoming customers of *Case 1*.

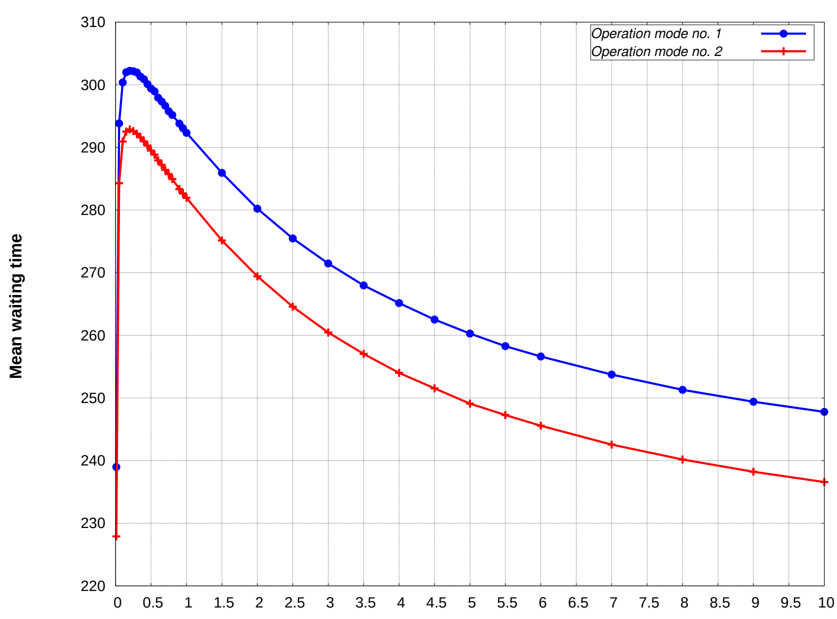

Fig. 4. Mean waiting time vs. intensity of incoming customers of *Case 2*.

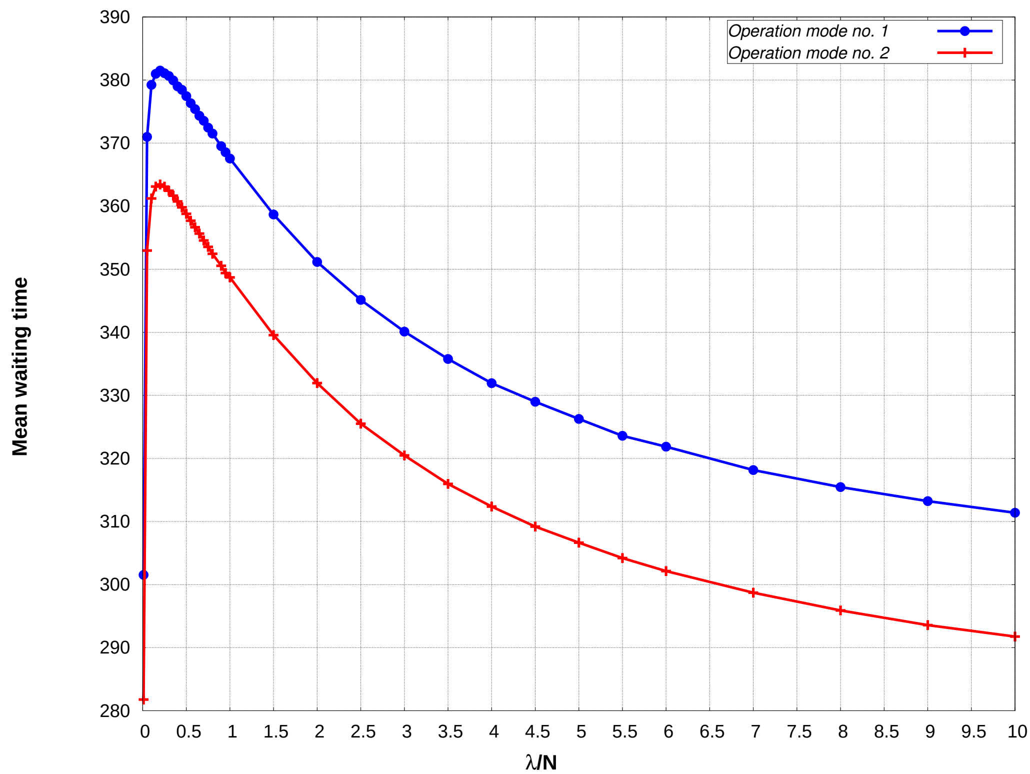

Fig. 5. Mean waiting time vs. intensity of incoming customers of *Case 3*.

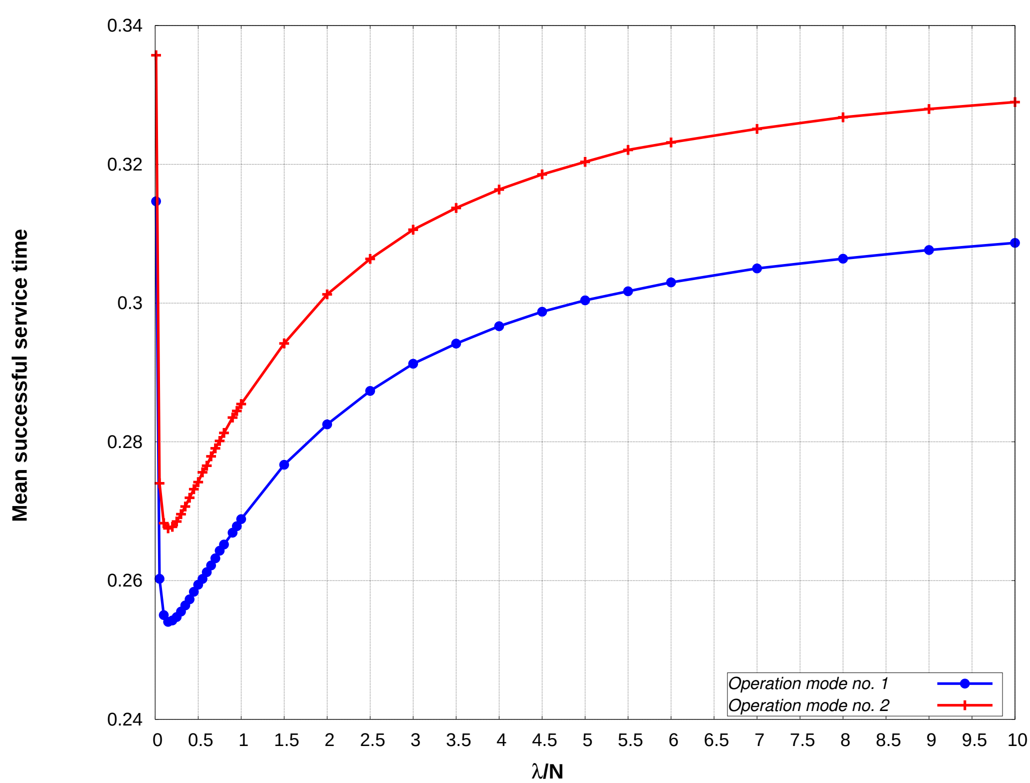

Fig. 6. Mean successful service time vs. intensity of incoming customers of *Case 1*.

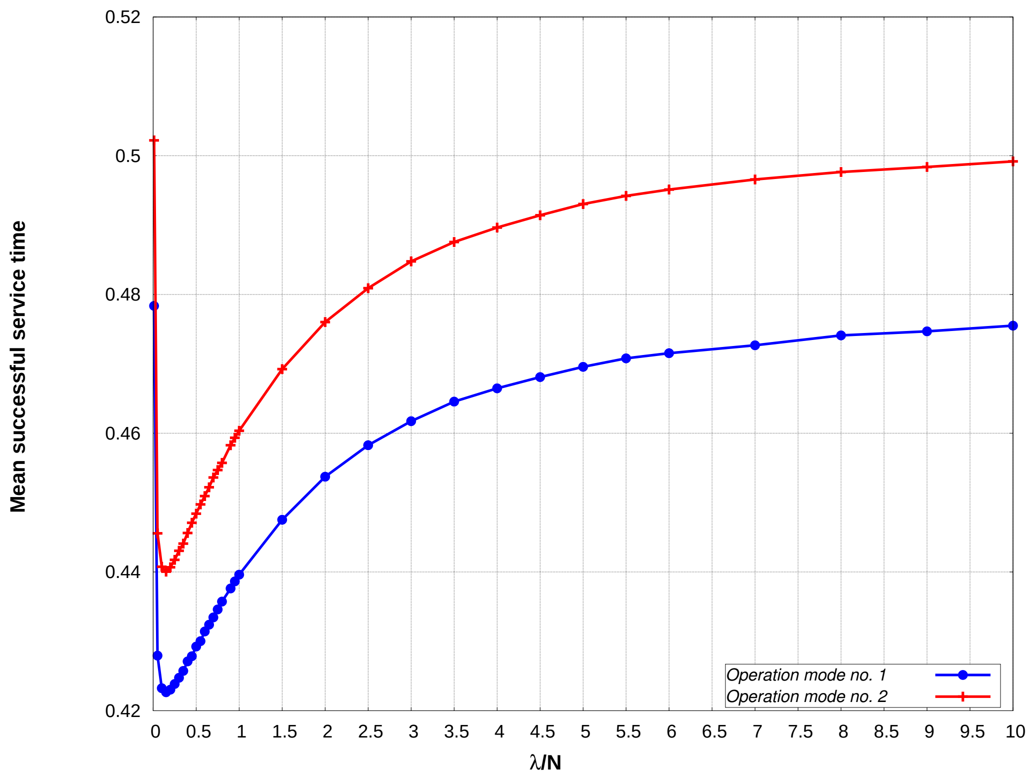

Fig. 7. Mean successful service time vs. intensity of incoming customers of *Case 2*.

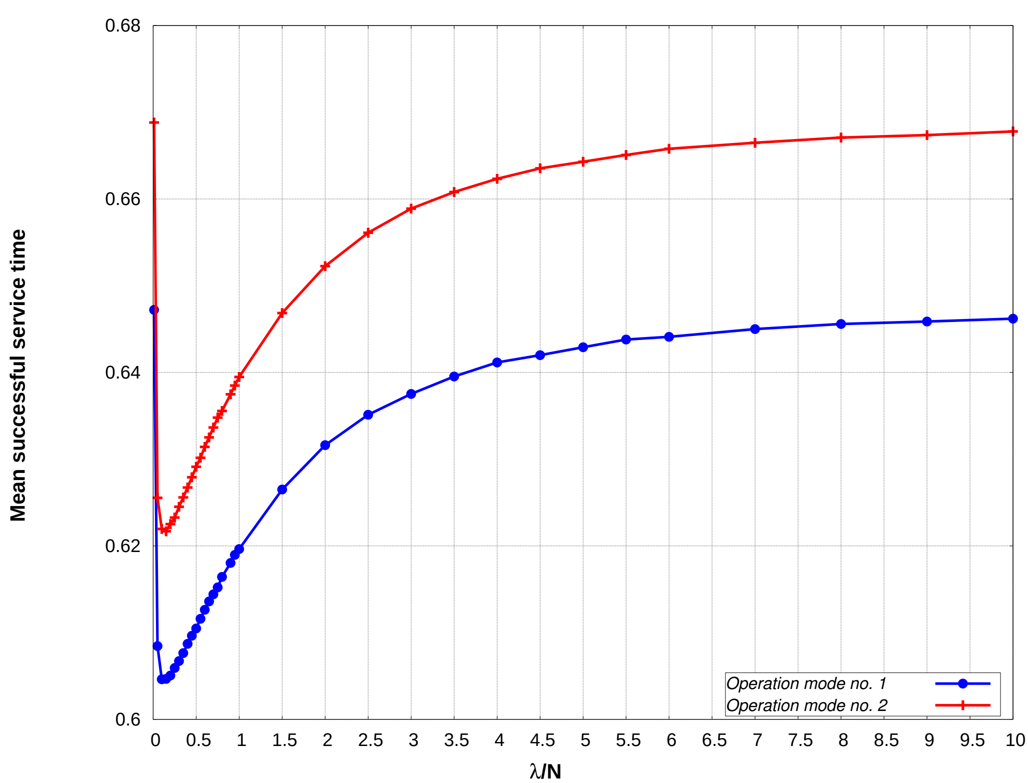

Fig. 8. Mean successful service time vs. intensity of incoming customers of *Case 3*.

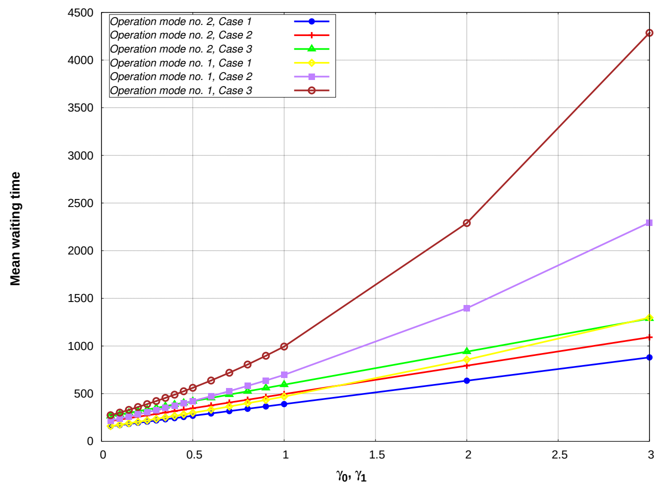

Fig. 9. Mean waiting time vs. intensity of failure rate.

Figure 9 shows the mean waiting time as a function of the failure rate. As is expected, the mean waiting time increases as the failure rate becomes with higher in all *Cases*. The difference between the operation modes is quite obvious. While using operation mode No. 2, the growth seems linear as in the case of operation mode No. 1. In *Case 3* the rise is quite significant.

3.2. Scenario B

In Scenario B the distribution of inter-arrival times of the customers is gamma distributed with parameter λ_1 and β_1. The following table (see Table 3) presents the numerical values of parameters of Scenario B.

Table 3. Numerical values of parameters of Scenario B

Case	N	α	β	γ_0	γ_1	γ_2	σ/N	α_1	β_1/N
1	100	1	1	0.1	0.1	1	0.01	0.5	0.01
2	100	1	1	0.1	0.1	1	0.01	1	0.01
3	100	1	1	0.1	0.1	1	0.01	2	0.01

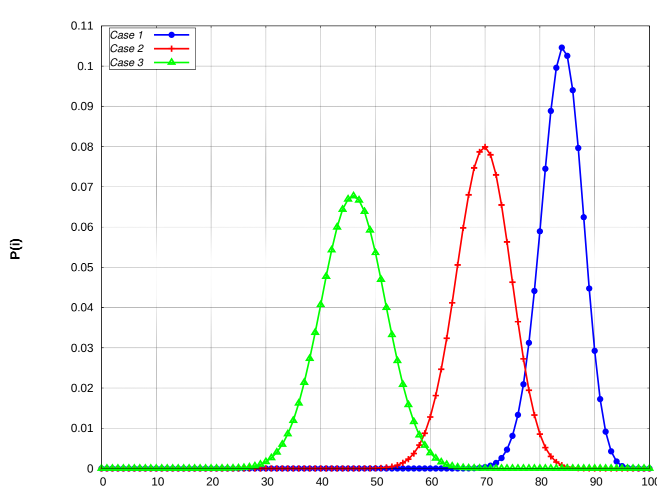

Fig. 10. Steady-state distributions of Scenario B.

Figure 6 displays the steady-state distributions under Scenario B. Now the service time is supposed to be exponentially distributed ($\alpha = 1$). With this modification compared to Scenario A the steady-state distributions still follow a normal distribution, and as the value of α_1 increases, the mean number of requests in the system decreases. In *Case 3* the mean number of customers in the system is significantly lower among the *Cases*. In Table 4 an estimate for basic performance measures can be found in connection with the cases.

Table 4. Numerical results of Scenario B

Case	E(NS)	$Var(NS)$	$E(T)$	$Var(T)$	$E(W)$	$Var(W)$	$E(S)$	$Var(S)$
1	83.856	14.4707	259.7716	121328.2493	258.6719	120904.8333	0.4707	0.2232
2	69.5984	24.7715	228.9076	97997.6219	227.8074	97637.2906	0.5025	0.2525
3	45.9508	34.3607	170.0134	62628.4067	168.9133	62379.0441	0.5806	0.337

The running parameter is α_1, which helps us to discover the impact of different distributions on the various examined performance measures. Figures 11,12, and 13 show a comparison of the mean waiting time between the two operation modes of the investigated *Cases*. As in Scenario A, using operation mode No. 2, when the interrupted requests stay at the server in case of server failure and their services continue after server is ready to process jobs again, ensures lower mean waiting times. In all the cases the mean waiting time starts to increase till α_1 reaches 0.3, after which it monotonically decreases. With higher values of α and β the differences between the operation modes are higher as well.

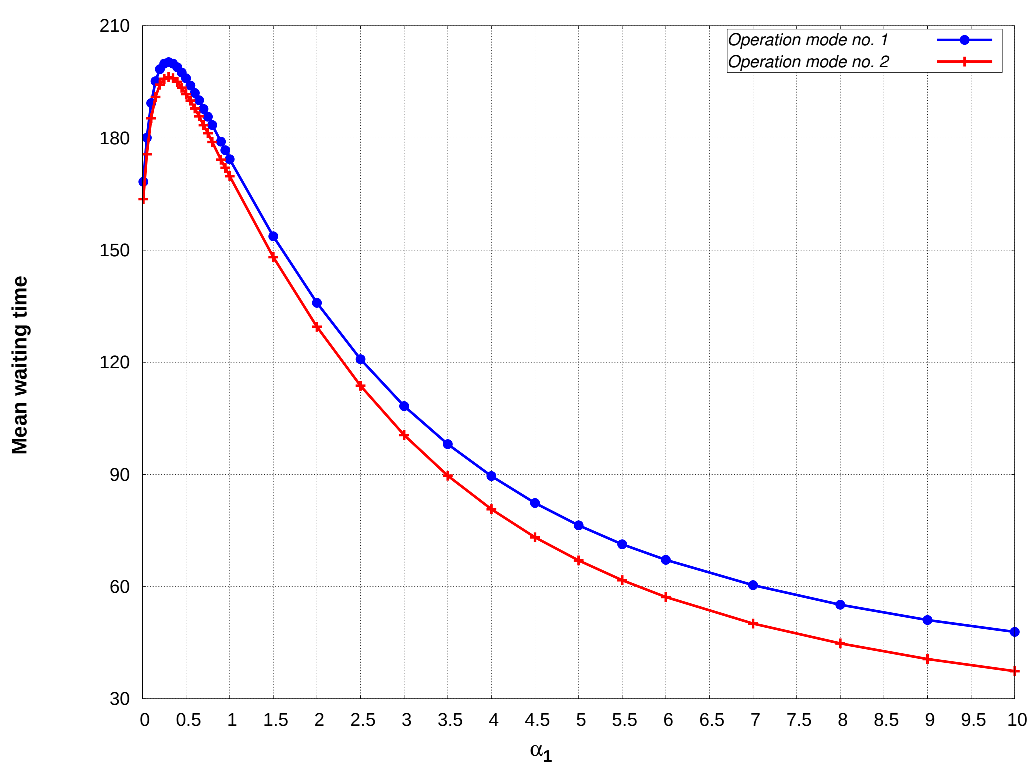

Fig. 11. Mean waiting time vs. shape parameter, $\alpha = \beta = 0.5$.

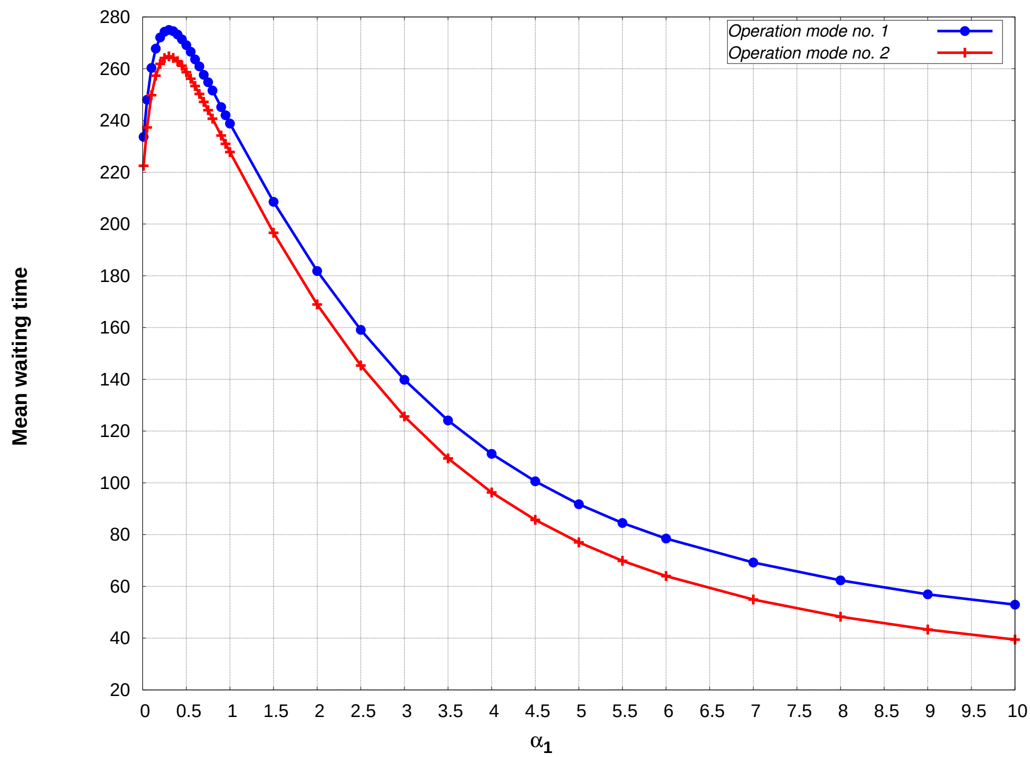

Fig. 12. Mean waiting time vs. shape parameter, $\alpha = \beta = 1$.

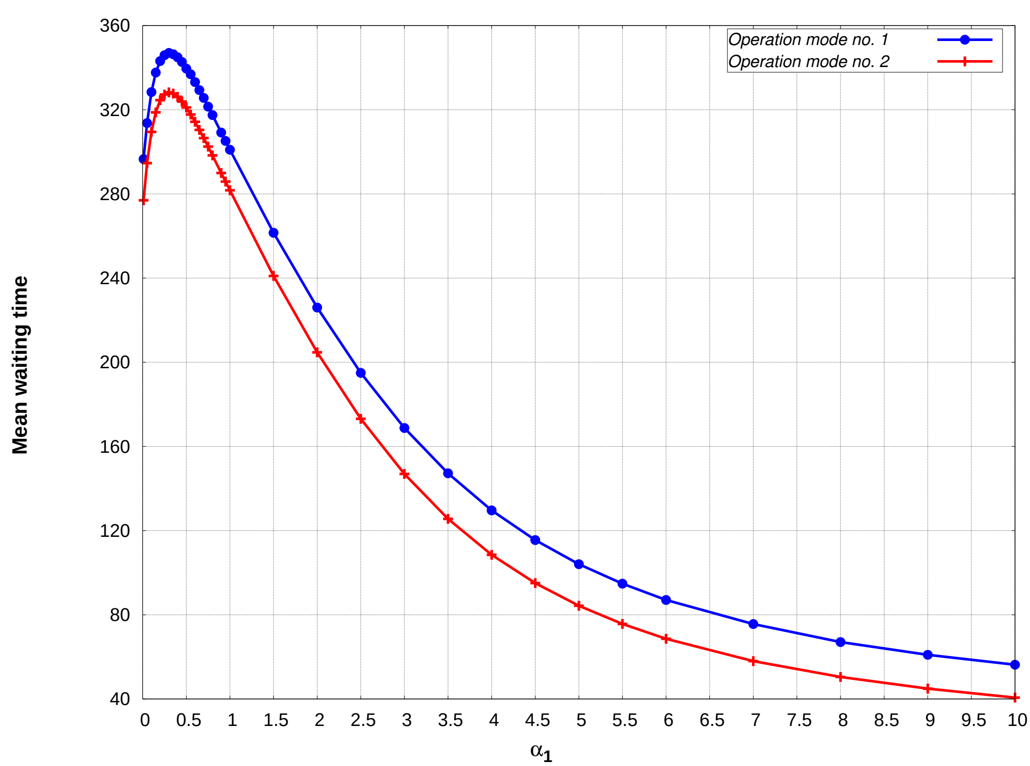

Fig. 13. Mean waiting time vs. shape parameter, $\alpha = \beta = 2$.

Figures 14, 15, and 16 show the mean successful service time vs. the shape parameter of the inter-arrival time using both operation mode. As we can see, in Scenario A we get what was expected: the use of operation mode No. 2 provides greater values of mean successful service time. The difference between the applied operation modes is quite large in all cases, especially if α and β are equal to 0.5. The mean successful service time behaves inversely as compared to the mean waiting time, because when the mean waiting time increases, the mean successful service time decreases, and vice versa.

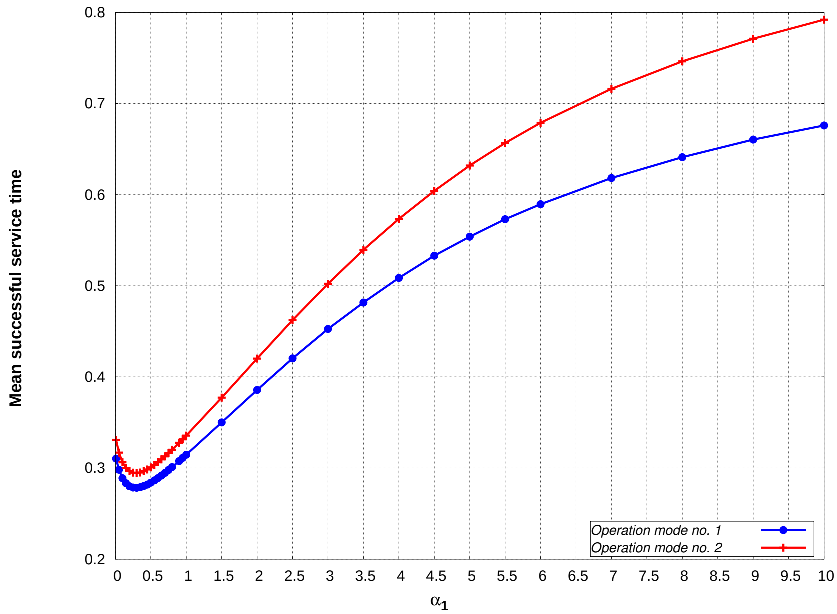

Fig. 14. Mean successful service time vs. shape parameter, $\alpha = \beta = 0.5$.

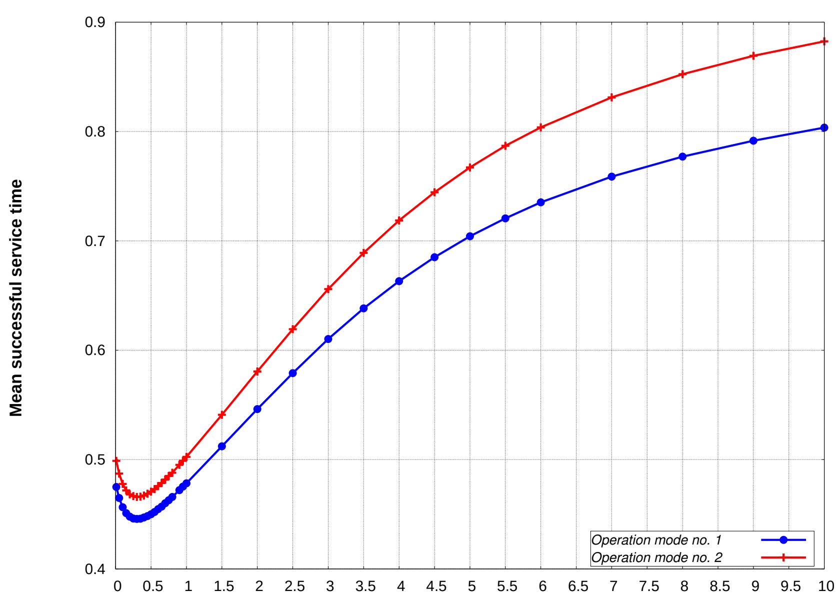

Fig. 15. Mean successful service time vs. shape parameter, $\alpha = \beta = 1$

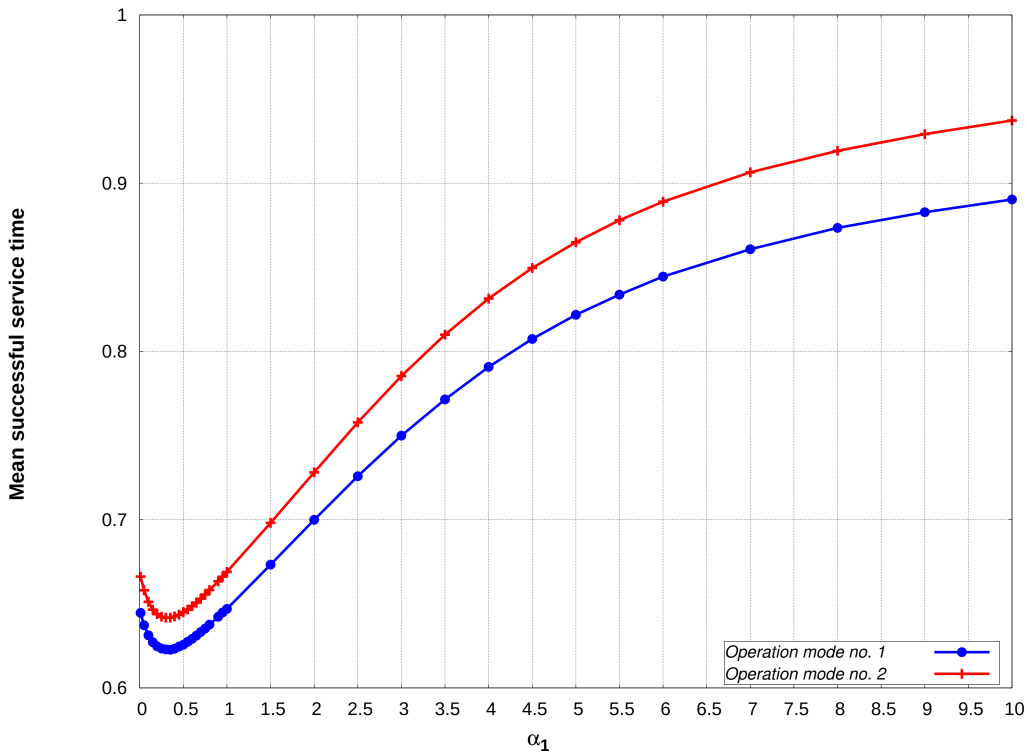

Fig. 16. Mean successful service time vs. shape parameter, $\alpha = \beta = 2$

4. Conclusions

In this paper a finite-source retrial queueing model with unreliable server and collisions was introduced. The effect of two different operation modes was analyzed with respect to basic performance measures. It was clearly recognizable that operation mode No. 2 gives more beneficial results as compared to operation mode No. 1. In all the cases in all scenarios it turns out that the steady-state distribution of the number of customers in the service facility is normal. We used SimPack to investigate the effect of various distributions of service and inter-arrival and retrial times on several main performance measures.

Acknowledgements

The work of János Sztrik is supported by the EFOP-3.6.1–16–2016–00022 project. The project is co-financed by the European Union and the European Social Fund. The work of Tamás Bérczes is supported by the EFOP–3.6.2–16-2017–00015 project. The project was supported by the European Union, co-financed by the European Social Fund.

REFERENCES

1. A.A. Ali and S. Wei, "Modeling of coupled collision and congestion in finite source wireless access systems," in: *Wireless Communications and Networking Conference (WCNC)* (2015), pp. 1113–1118.

2. B. Almási, J. Roszik, and J. Sztrik, "Homogeneous finite-source retrial queues with server subject to breakdowns and repairs," *Math. Comput. Model.*, **42**, No. 5–6, 673–682 (2005).

3. J.R. Artalejo and A. Gómez-Corral, *Retrial Queueing Systems: A computational Approach*, Springer, Berlin (2008).

4. S. Balsamo, G.L. Dei Rossi, and A. Marin, "Modelling retrial-upon-conflict systems with product-form stochastic petri nets," in: *International Conference on Analytical and Stochastic Modeling Techniques and Applications*, Springer (2013), pp. 52–66.
5. V.I. Dragieva, "Number of retrials in a finite source retrial queue with unreliable server," *Asia-Pac. J. Oper. Res.*, **31**, No. 2 (2014).
6. P. A. Fishwick, "Simpack: Getting started with simulation programming in C and C++," in: *WSC 92 Proceedings of the 24th Conference on Winter Simulation*, ACM, New York (1992), pp. 154–162.
7. N. Gharbi and C. Dutheillet, "An algorithmic approach for analysis of finite-source retrial systems with unreliable servers," *Comput. Math. Appl.*, **62**, No. 6, 2535–2546 (2011).
8. J. Kim and B. Kim, "A survey of retrial queueing systems," *Ann. Oper. Res.*, **247**, No. 1, 3–36 (2016).
9. J.S. Kim, "Retrial queueing system with collision and impatience," *Commun. Kor. Math. Soc.*, **25**, No. 4, 647–653 (2010).
10. A. Kvach and A. Nazarov, "Sojourn time analysis of finite source Markov retrial queuing system with collision," *Commun. Comput. Inform. Sci.*, **564**, 64–72 (2015).
11. A. Nazarov, A. Kvach, and V. Yampolsky, "Asymptotic analysis of closed Markov retrial queuing system with collision," *Commun. Comput. Inform. Sci.*, **487**, 334–341 (2014).
12. Y. Peng, Z. Liu, and J. Wu, "An M/G/1 retrial G-queue with preemptive resume priority and collisions subject to the server breakdowns and delayed repairs," *J. Appl. Math. Comput.*, **44**, No. 1–2, 187–213 (2014).
13. J. Wang, L. Zhao, and F.Zhang, "Analysis of the finite source retrial queues with server breakdowns and repairs," *J. Ind. Man. Opt.*, **7**, No. 3, 655–676 (2011).
14. F. Zhang and J. Wang, "Performance analysis of the retrial queues with finite number of sources and service interruptions," *J. Kor. Stat. Soc.*, **42**, No. 1, 117–131 (2013).