Comparation of two single-server queueing systems with exponential service times and threshold-based renovation

Viana C. C. Hilquias[1], Ivan S. Zaryadov[1,2]

[1]*Department of Applied Probability and Informatics, Peoples' Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation*

[2]*Institute of Informatics Problems, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 44-2 Vavilov Str., Moscow 119333, Russian Federation*

Abstract

In this paper we compare the simulation results of two types of $GI/M/1$ infinite capacity queues with the implemented threshold-based renovation mechanism. As usual renovation implies probabilistic dropping of customers from the queue upon service completions. In the systems of the first type there is the threshold value (indication the queue length) which controls the activation of the renovation mechanism. In the systems of the second type the threshold value not only triggers the renovation, but also specifies the area in the queue wherefrom the customers cannot be dropped. For both types of systems the main stationary characteristics are obtained. Numerical results are also provided, which illustrate the performance of the queues for different sets of simulation parametres. The simulation results comparison are presented in the section 4.

Keywords

renovation mechanism, active queue management, threshold policy, congestion control, GPSS simulation

1. Introduction

According to [1], the development of modern mechanisms active queue management (AQM) keeps attracting attention from the operation research community. The classic such mechanism is the RED [2, 3] mechanism.

This work is devoted to the comparison of single-threshold queuing systems with renovation [4], which is a continuation of the research formulated in the work [5]. Here we elaborate further on the mechanism of renovation and describe two new settings. In the first setting we consider the single-server queue with the threshold, which determines the boundary in the queue, starting from which the dropping of customers begins. The second setting covers the case when the threshold value also specifies the area in the queue, wherefrom the customers cannot be dropped. The structure of the paper is as follows. Section 2 presents the queuing system under the first setting, section 3describes the system under the second setting. Simulation

Workshop on information technology and scientific computing in the framework of the XI International Conference Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems (ITTMM-2021), Moscow, Russian, April 19–23, 2021

hilvianamat1@gmail.com (V. C. C. Hilquias); zaryadov-is@rudn.ru (I. S. Zaryadov)
0000-0002-7909-6396 (I. S. Zaryadov)

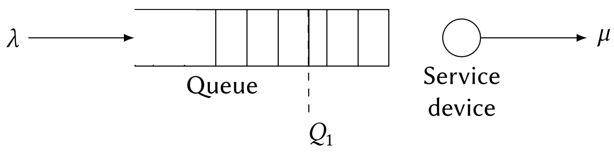

Figure 1: Queuing system type 1

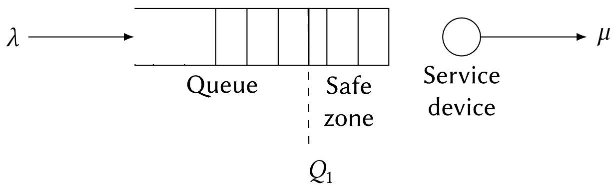

Figure 2: Queuing system type 2

results are presented in the section 4. The last section concludes the paper with the short discussion.

2. First setting

Consider the $GI/M/1/\infty$ queuing system, shown in the Fig. 1, with the implemented renovation [4, 5] mechanism and a threshold value Q_1, which determines the boundary in the queue, starting from which the dropping of customers begins.

Renovation mechanism: let i be the number of requests in queue. If $0 \leq i \leq Q_1$, then after the end of the service, the request simply leaves the system. If $i > Q_1$, then either with the probability $p(0 < p < 1)$ the request that has been served simply leaves the system, or with the probability $q = 1 - p$ it resets the queue.

3. Second setting

Consider now the $GI/M/1/\infty$ system, shown in the figure 2, where the threshold value Q_1 defines not only the boundary in queue, upon exceeding which by the current queue length the renovation mechanism is activated, but also specifies the area in the queue, wherefrom the customers cannot be dropped.

Renovation mechanism: let i be number of requests in the queue. If i, then after the end of the service, the request simply leaves the system. If $i > Q_1$, then either with the probability $p(0 < p < 1)$ the request that has been served simply leaves the system, or with the probability $q = 1 - p$ at the moment of leaving the system, reset all requests located after the threshold valueQ_1.

Table 1
Simulation results for different drop probabilities

Drop probability		**0.0025**	**0.005**	**0.01**	**0.025**	**0.05**	**0.1**
Generated tasks	**sys.1**	999703	999937	999592	999511	1000119	998767
	sys.2	999703	999714	999993	999997	1000311	1000045
Serviced tasks	**sys.1**	999702	999915	999513	999287	999679	998017
	sys.2	999702	999712	999982	999989	1000285	1000005
Serviced tasks without calling the renv. mech.	**sys.1**	998660	998898	998532	998386	998830	997327
	sys.2	998660	998682	998985	998995	999310	999057
Dropped tasks	**sys.1**	0	22	79	223	440	750
	sys.2	0	2	6	11	26	40
Probability of servicing tasks	**sys.1**	1.0000	1.0000	0.9999	0.9998	0.9996	0.9992
	sys.2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Probability of dropping tasks	**sys.1**	0.0000	0.0000	0.0001	0.0002	0.0004	0.0008
	sys.2	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Average queue length	**sys.1**	0.498	0.498	0.497	0.496	0.495	0.493
	sys.2	0.498	0.498	0.497	0.498	0.5	0.501
Maximum queue length	**sys.1**	17	17	17	17	17	15
	sys.2	17	17	17	17	17	17
Average waiting time of a task in the queue	**sys.1**	0.1	0.1	0.1	0.099	0.099	0.099
	sys.2	0.1	0.1	0.1	0.1	0.1	0.1
Calling the renovation mechanism (no reset)	**sys.1**	1041	1015	974	878	806	615
	sys.2	1041	1027	989	974	750	0.1
Calling the renovation mechanism (reset)	**sys.1**	0	2	7	22	43	75
	sys.2	0	2	7	19	49	90

4. Simulation results

The results of the simulation models of both systems are presented in the tables, which are constructed as follows: the first line - the number of orders generated during the simulation; the second line is the number of requests served; the third line is the number of requests serviced without calling the renovation mechanism; the fourth line is the number of discarded requests; the fifth line is the probability of servicing the request accepted into the system; the sixth line is the probability of dropping (due to the renovation mechanism) a request accepted into the system; the seventh line - the values of the average queue length; the eighth line is the maximum length of the queue; the ninth line is the average waiting time for service; the tenth line - how many times the renovatione mechanism was called without dropping orders; the last line - how many times the renovatione mechanism was called with dropping requests

The first three tables consider the behavior of various characteristics of two systems (sys.1 and sys.2) at low values of the drop probability for the following system load options: medium load ($\rho = 0.5$) — see table 1, high load ($\rho = 1$) — see table 2, and very high system load ($\rho = 2$) — see table 3.

At low values of the system load ($\rho < 0.2$) , the update mechanism is not activated in both systems, and therefore for these systems absolutely the same values of the main characteristics are obtained.

Table 2
Simulation results for different drop probabilities

Drop probability		**0.0025**	**0.005**	**0.01**	**0.025**	**0.05**	**0.1**
Generated tasks	**sys.1**	999437	1000627	999286	1000116	999289	999928
	sys.2	1000148	1001042	1000211	1000028	999623	1000510
Serviced tasks	**sys.1**	995695	995719	992516	988573	986160	984624
	sys.2	999407	999823	998559	997262	996341	996638
Serviced tasks without calling the renv. mech.	**sys.1**	960378	966875	971329	974864	978438	979997
	sys.2	951429	956594	963167	967636	971884	977464
Dropped tasks	**sys.1**	3713	4901	6769	11542	13128	15303
	sys.2	738	1212	1650	2757	3279	3861
Probability of servicing tasks	**sys.1**	0.9963	0.9951	0.9932	0.9885	0.9869	0.9847
	sys.2	0.9993	0.9988	0.9983	0.9972	0.9967	0.9961
Probability of dropping tasks	**sys.1**	0.0037	0.0049	0.0068	0.0115	0.0131	0.0153
	sys.2	0.0007	0.0012	0.0016	0.0028	0.0033	0.0039
Average queue length	**sys.1**	8.127	7.934	7.487	7.088	6.755	6.506
	sys.2	8.802	8.697	8.351	8.113	8.067	7.982
Maximum queue length	**sys.1**	79	75	75	70	56	48
	sys.2	80	80	75	71	58	57
Average waiting time of a task in the queue	**sys.1**	0.898	0.876	0.83	0.789	0.753	0.726
	sys.2	0.967	0.955	0.92	0.895	0.89	0.879
Calling the renovation mechanism (no reset)	**sys.1**	35215	28704	20994	13365	7318	4142
	sys.2	47858	43015	35036	28807	23227	17243
Calling the renovation mechanism (reset)	**sys.1**	101	139	192	343	403	484
	sys.2	119	213	355	818	1229	1930

The first table has a low threshold value $Q_1 = 10$, otherwise the update mechanism is not enabled, for the second and the third tables the threshold value Q_1 is set 30. The simulation time is 100000 unit of time.

As can be seen from the simulation results, presented in the table 1, the time characteristic (average waiting time of a task in the queue) and queue size characteristics (average and maximum queue lengths) in the case of an average load ($\rho = 0.5$) for the systems of both types are approximately the same, and in the systems of the second type the probability of dropping a task is zero.

As can be seen from the simulation results, presented in the table 2, the time characteristic (average waiting time of a task in the queue) in the case of high load ($\rho = 1$) for the second type system is 10-20% more than for the first type system (similarly for average and maximum queue lengths). But the probability of dropping an accepted claim for a system of the first type is four times greater than the value of the saim characteristic for a system of the second type

Finally, for the case of ultra-high system load ($\rho = 2$ and more) the values of the probability of tasks dropping are approximately the same. The values of average waiting time, average and maximum queue length for the second type system are only 10% higher then for the first type system.

The following three tables show the dependence of the characteristics of both systems on the threshold value; three options for the system load are also considered: medium ($\rho = 0.5$) — see

Table 3
Simulation results for different drop probabilities

Drop probability		**0.0025**	**0.005**	**0.01**	**0.025**	**0.05**	**0.1**
Generated tasks	**sys.1**	2001200	1999335	2001398	2001232	2000240	1999833
	sys.2	2000537	2000595	2000129	1999163	2001430	1999081
Serviced tasks	**sys.1**	1000981	1000105	998506	993705	990445	988356
	sys.2	1002552	1001822	1002170	1002148	1000680	1001185
Serviced tasks without calling the renv. mech.	**sys.1**	68748	130644	227915	422930	593801	739453
	sys.2	2336	4702	9663	23331	42012	73252
Dropped tasks	**sys.1**	1000026	999196	1002569	1007494	1009740	1011469
	sys.2	996859	998611	997931	996975	1000703	997859
Probability of servicing tasks	**sys.1**	0.5002	0.5002	0.4989	0.4965	0.4952	0.4942
	sys.2	0.5011	0.5008	0.5011	0.5013	0.5000	0.5008
Probability of dropping tasks	**sys.1**	0.4997	0.4998	0.5009	0.5034	0.5048	0.5058
	sys.2	0.4983	0.4992	0.4989	0.4987	0.5000	0.4992
Average queue length	**sys.1**	410.622	205.422	105.087	47.323	29.091	20.981
	sys.2	439.343	226.549	129.525	71.118	50.899	40.726
Maximum queue length	**sys.1**	3100	2206	1117	485	263	159
	sys.2	4168	1639	961	470	295	161
Average waiting time of a task in the queue	**sys.1**	20.541	10.296	5.271	2.382	1.469	1.062
	sys.2	21.961	11.324	6.476	3.557	2.543	2.037
Calling the renovation mechanism (no reset)	**sys.1**	929924	865109	762976	556640	376783	224057
	sys.2	997745	992130	982519	954360	910830	835335
Calling the renovation mechanism (reset)	**sys.1**	2308	4351	7614	14134	19860	24845
	sys.2	2470	4989	9987	24456	47837	92597

table 4 , large ($\rho = 1$) — see table 5, and super-large ($\rho = 2$) — see table 6.

It should be noted that in Table 4 the threshold values are taken less than in Tables 5 and 6, since at sufficiently large threshold values, both systems at low and medium system load, both systems behave completely the same

Below (see table 7) is presented a table with simulation results (GPSS simulations for both systems (sys.1 and sys.2) were performed with the following initial parameters: drop probability $q = 0.01$, arrival rate — 10 task per 1 unit of time, and the simulation time is 100000 unit of time) for different service rates.

5. Conclusion

In this paper we compared the simulation results for two types of a single-server queuing system $GI/M/1/\infty$ with an infinite capacity storage, with renovation mechanism and a threshold value.

Comparing the simulation results, we can draw the following conclusion: at low and super-large loads, systems of both types behave approximately the same; at high loads, a system of the second type is preferable, since it is significantly less likely to drop an incoming request.

Our future goal is to compare the analytical expressions for some general time-probability characteristics (such as the distribution of the number of applications in the system) with

Table 4
Simulation results for different threshold values

Threshold value		**3**	**5**	**7**	**10**	**15**
Generated tasks	**sys.1**	998461	999419	999184	999592	999010
	sys.2	998550	999042	999169	999993	998325
Serviced tasks	**sys.1**	995080	997899	998650	999513	998994
	sys.2	998009	998881	999119	999982	998324
Serviced tasks without calling the renv. mech.	**sys.1**	873689	967376	991321	998532	998974
	sys.2	873901	967608	991693	998985	998304
Dropped tasks	**sys.1**	3381	1519	534	79	16
	sys.2	541	161	50	6	1
Probability of servicing tasks	**sys.1**	0.9966	0.9985	0.9995	0.9999	1.0000
	sys.2	0.9995	0.9998	0.9999	1.0000	1.0000
Probability of dropping tasks	**sys.1**	0.0034	0.0015	0.0005	0.0001	0.0000
	sys.2	0.0005	0.0002	0.0001	0.0000	0.0000
Average queue length	**sys.1**	0.49	0.496	0.497	0.497	0.499
	sys.2	0.497	0.499	0.499	0.497	0.499
Maximum queue length	**sys.1**	16	18	17	17	17
	sys.2	16	18	17	17	17
Average waiting time of a task in the queue	**sys.1**	0.099	0.1	0.099	0.1	0.1
	sys.2	0.1	0.1	0.1	0.1	0.1
Calling the renovation mechanism (no reset)	**sys.1**	120231	30219	7254	974	19
	sys.2	122906	30970	7351	989	19
Calling the renovation mechanism (reset)	**sys.1**	1160	303	75	7	1
	sys.2	1202	302	75	7	1

the obtained simulation results in the GPSS system. Also we plan to analyze the $GI/M/1/\infty$ queueing system with renovation mechanism and two threshold values (the first value controls the activation of the renovation mechanism, the second value specifies the area in the queue wherefrom theincoming tasks cannot be dropped).

Acknowledgments

This paper has been supported by the RUDN University Strategic Academic Leadership Program (Viana C. C. Hílquias and I.S. Zaryadov, mathematical model formulation and simulation model development). Also the publication has been funded by Russian Foundation for Basic Research (RFBR) according to the research project No. 19-07-00739 (I.S. Zaryadov, mathematical model development and numerical analysis)

References

[1] F. J.Baker, G. Fairhurst, IETF Recommendations Regarding Active Queue Management, Request for Comments RFC 7567, Internet Engineering Task Force (IETF), 2015. URL: https://datatracker.ietf.org/doc/html/rfc7567.

Table 5
Simulation results for different threshold values

Threshold value		**10**	**30**	**50**	**100**
Generated tasks	**sys.1**	999875	999286	999775	999350
	sys.2	997944	1000211	999710	1000892
Serviced tasks	**sys.1**	965620	992516	998270	999229
	sys.2	983879	998559	999370	1000874
Serviced tasks without calling the renv. mech.	**sys.1**	738473	971329	994540	999185
	sys.2	708912	963167	994122	1000824
Dropped tasks	**sys.1**	34254	6769	1505	112
	sys.2	14058	1650	336	12
Probability of servicing tasks	**sys.1**	0.9657	0.9932	0.9985	0.9999
	sys.2	0.9859	0.9983	0.9997	1.0000
Probability of dropping tasks	**sys.1**	0.0343	0.0068	0.0015	0.0001
	sys.2	0.0141	0.0016	0.0003	0.0000
Average queue length	**sys.1**	5.623	7.487	8.668	9.166
	sys.2	6.418	8.351	9.11	9.467
Maximum queue length	**sys.1**	67	75	87	114
	sys.2	80	70	96	114
Average waiting time of a task in the queue	**sys.1**	0.64	0.83	0.954	1.008
	sys.2	0.72	0.92	1.002	1.036
Call of the renovation mechanism (no reset)	**sys.1**	224895	20994	3703	42
	sys.2	272352	35036	5184	48
Call of the renovation mechanism (reset)	**sys.1**	2251	192	27	1
	sys.2	2614	355	63	1

[2] S. Floyd, V. Jacobson, Random early detection gateways for congestion avoidance, IEEE/ACM Transactions on Networking 4 (1993) 397–413. doi:10.1109/90.251892.

[3] K. K. Ramakrishnan, S. Floyd, D. L. Black, The Addition of Explicit Congestion Notification (ECN) to IP, Request for Comments RFC 3168, Internet Engineering Task Force (IETF), 2001. URL: https://datatracker.ietf.org/doc/html/rfc3168.

[4] I. S. Zaryadov, A. V. Pechinkin, Stationary time characteristics of the $gi/m/n/\infty$ system with some variants of the generalized renovation discipline, Automation and Remote Control 70 (12009) 2085–2097. doi:10.1134/S0005117909120157.

[5] V. C. C. Hilquias, I. S. Zaryadov, V. V. Tsurlukov, T. A. Milovanova, E. V. Bogdanova, A. V. Korolkova, D. S. Kulyabov, The general renovation as the active queue management mechanism. some aspects and results, in: V. Vishnevskiy, K. Samouylov, D. Kozyrev (Eds.), Distributed Computer and Communication Networks. DCCN 2019, volume 1141 of *Communications in Computer and Information Science*, Springer, Cham, 2019, pp. 488–502. doi:10.1007/978-3-030-36625-4_39.

Table 6
Simulation results for different threshold values

Threshold value		**10**	**30**	**50**	**100**
Generated tasks	**sys.1**	2000277	2001398	1999814	2000181
	sys.2	2000055	2000129	2000507	1999895
Serviced tasks	**sys.1**	997658	998506	998085	999285
	sys.2	1002451	1002170	1002194	1002251
Serviced tasks without calling the renv. mech.	**sys.1**	91398	227915	332711	494479
	sys.2	9437	9663	9802	9618
Dropped tasks	**sys.1**	1002311	1002569	1001689	1000626
	sys.2	997594	997931	998220	997460
Probability of servicing tasks	**sys.1**	0.4988	0.4989	0.4991	0.4996
	sys.2	0.5012	0.5011	0.5010	0.5012
Probability of dropping tasks	**sys.1**	0.5011	0.5009	0.5009	0.5003
	sys.2	0.4988	0.4989	0.4990	0.4988
Average queue length	**sys.1**	101.95	105.087	108.049	127.486
	sys.2	112.636	129.525	150.897	200.546
Maximum queue length	**sys.1**	1121	1011	961	882
	sys.2	1121	1117	966	1104
Average waiting time of a task in the queue	**sys.1**	5.12	5.271	5.421	6.389
	sys.2	5.632	6.476	7.543	10.028
Calling the renovation mechanism (no reset)	**sys.1**	897129	762976	658714	499870
	sys.2	983167	982519	982488	992325
Calling the renovation mechanism (reset)	**sys.1**	9130	7614	6659	4935
	sys.2	9846	9987	9903	9618

6. Appendices

A. Code in GPSS for system 1

```
PROB FUNCTION RN1,D2
0.01,0/1,1                ;drop probability

Q_1 VARIABLE 30           ;threshold value

GENERATE(Exponential(1,0,2))
LINK LIST1 FIFO metka1
metka1 SEIZE Pribor

ADVANCE (Exponential(1,0,6))
TEST L CH$LIST1,V$Q_1,metka2
RELEASE Pribor
TRANSFER ,metka_end

metka2 TEST E FN$PROB,0,metka3
```

Table 7
Simulation results for different service intensities

Service intensity		**5 in 1s**	**8 in 1s**	**11 in 1s**	**15 in 1s**
Generated tasks	**sys.1**	999616	1001255	999286	999137
	sys.2	999540	997652	1000211	999137
Serviced tasks	**sys.1**	498806	787414	992516	999136
	sys.2	500950	801276	998559	999136
Serviced tasks without calling the renv. mech.	**sys.1**	113109	417466	971329	999129
	sys.2	4789	95898	963167	999129
Dropped tasks	**sys.1**	500645	213829	6769	0
	sys.2	498472	196344	1650	0
Probability of servicing tasks	**sys.1**	0.4990	0.7864	0.9932	1.0000
	sys.2	0.5012	0.8032	0.9983	1.0000
Probability of dropping tasks	**sys.1**	0.5008	0.2136	0.0068	0.0000
	sys.2	0.4987	0.1968	0.0016	0.0000
Average queue length	**sys.1**	106	35	7.487	1.34
	sys.2	131	54	8.351	1.34
Maximum queue length	**sys.1**	845	291	75	31
	sys.2	934	308	70	31
Average waiting time of a task in the queue	**sys.1**	10	3	0.83	0.201
	sys.2	13	5	0.92	0.201
Calling the renovation mechanism (no reset)	**sys.1**	381939	366330	20994	6
	sys.2	491273	698190	35036	6
Calling the renovation mechanism (reset)	**sys.1**	3757	3617	192	0
	sys.2	4887	7187	355	0

```
RELEASE Pribor
UNLINK LIST1,metka1,1
UNLINK LIST1,metka4,ALL
metka4 TERMINATE 0

metka3 RELEASE Pribor
metka_end UNLINK LIST1,metka1,1
TERMINATE 0

GENERATE        100000        ; Working time (seconds)
TERMINATE       1             ; Minus one minute
START    1                    ; Start from the first minute
```

B. Code in GPSS for system 2

```
PROB FUNCTION RN1,D2
0.01,0/1,1              ;drop probability
```

```
Q_1 VARIABLE 30            ;threshold value

GENERATE(Exponential(1,0,1/10))
LINK LIST1 FIFO metka1
metka1 SEIZE Pribor

ADVANCE (Exponential(1,0,1/11))
TEST L CH$LIST1,V$Q_1,metka2
RELEASE Pribor
TRANSFER ,metka_end

metka2 TEST E FN$PROB,0,metka3
RELEASE Pribor
UNLINK LIST1,metka1,1
UNLINK LIST1,metka4,(CH$LIST1-V$Q_1)
metka4 TERMINATE 0

metka3 RELEASE Pribor
metka_end UNLINK LIST1,metka1,1
TERMINATE 0

GENERATE       100000        ; Working time (seconds)
TERMINATE      1             ; Minus one minute
START   1                    ; Start from the first minute
```