Performance Analysis of a Prioritized, Limited Round-Robin System

~ Influence that a portion of quanta may have on the performance ~

Yoshiaki Shikata and Nobutane Hanayama

[Abstract]

In a prioritized, limited Round-Robin (RR) system where *N* priority classes are allowed in a single-server system, a higher-priority request is allocated a larger number of quanta compared to a lower-priority request. The sum of the number of quanta included in each RR cycle is restricted to a fixed value. An arriving request that cannot be allocated quanta because of this restriction is either entered in the service waiting queue (the waiting system) or rejected (the loss system). Practical performance measures such as the relationship between the mean sojourn time, mean queue waiting time, loss probability, and quantum size in the case that a portion of quanta must be subtracted from the remaining sojourn time are evaluated via simulation. In the evaluation, the sojourn time of an arriving request is evaluated using the requested service time, number of quanta included in each RR cycle, and quantum size. Then, the change in the number of quanta needed before service is complete is reevaluated at the arrival and departure of other requests. Tracking these events and calculations enables us to analyze the performance of the prioritized, limited RR rule.

Keywords:

prioritized RR, portion of quanta, sojourn time, loss probability, simulation

1. Introduction

Under the Round-Robin (RR) rule, a processor allocates a fixed amount of time, called a quantum, to each request in a fixed order. If the requested service time (the total time required from the processor) is completed in less than the quantum, the request leaves; otherwise, it is added to the end of the queue of requests waiting for a quantum allocation and waits its turn to receive another quantum of service. It continues in this fashion until its requested service time has been obtained from the processor. Moreover, under the prioritized RR rule, where *N* priority classes are permitted, the processor allocates m_i (≥ 1, i :1- N, called the priority ratio) quanta to each class-i request in each RR cycle. Here, the RR cycle is a series of quanta cyclically allocated to each request in a fixed order. In such a prioritized RR paradigm, the service ratio for individual requests decreases as the number of arriving requests increases. Therefore, theoretically, the sojourn time of each request increases to infinity as the number of arriving requests increases. Here, the sojourn time is the time from arrival of a request at the server to departure. In order to prevent such an increase and develop a realistic sharing model, the number of requests that are allocated quanta must be limited. In such

a prioritized, limited RR system, the number of quanta that can be included in each cycle is kept below a fixed value (the service-facility capacity). Arriving requests that cannot be allocated quanta because of such a restriction are entered into the service waiting queue or are rejected.

In this study, we evaluate practical performance measures such as the mean sojourn time of requests, mean waiting time in the service waiting queue, and loss probability of prioritized, limited RR systems in the case that a portion of quanta must be subtracted from the remaining sojourn time via simulation. In this simulation algorithm, the sojourn time of an arriving request is first evaluated using the requested service time, number of quanta included in each RR cycle, and quantum size. Then, the change in the number of quanta needed before service is complete is reevaluated at the arrival and departure of other requests. Tracking these events and calculations enables us to analyze the performance of the prioritized, limited RR rule. The proposed RR rule and performance evaluation results are realistic in server and client type communication systems where a time-shared computer is employed as the server system.

The Processor Sharing (PS) rule, an idealization of quantum-based RR scheduling at the limit where the quantum size becomes infinitesimal, has been the subject of many papers [1-4]. A limited PS system and a prioritized, limited PS system, in which the number of requests receiving service is kept below a fixed value, have also been proposed, and the performance of these systems has been analyzed [5, 6]. However, a few explicit results are available for the RR policy. The influence of the variable job or quantum size in the presence of switching overhead on the mean sojourn time of the non-limited RR rule is evaluated [7–9]. The performance of a prioritized, limited RR rule, where two priority classes are permitted, was analyzed via simulation [10]. However, in this analysis the influence that the subtraction of a portion of quanta from the remaining sojourn time may have on the mean sojourn time, mean waiting time in the service waiting queue, and the loss probability in the prioritized, limited RR system was not investigated.

2. Prioritized, limited round-robin system

2.1 System concept

Suppose there are N classes, and an arriving request encounters n_j class-j requests (including the arriving request). Furthermore, let m_j (≥ 1) denote the priority ratio of the class-j request and SFC ($\leq \infty$) denote the service-facility capacity. According to the proposed prioritized, limited RR rule, if $(\sum_{j=1}^{N} m_j * n_j) \leq$ SFC, an arriving class-k request is allocated m_k quanta. Otherwise, when $(\sum_{j=1}^{N} m_j * n_j) >$ SFC, the arriving request is queued in the corresponding class waiting room or rejected.

2.2 Extension or reduction of the remaining sojourn time

Under the prioritized RR rule, whenever the service for an arriving request begins or the requested service time of a request is over, the remaining sojourn time of each request currently receiving service is extended or reduced, respectively. This extension or reduction of the remaining sojourn time can be calculated using the number of RR cycles that are necessary before the service

is completed, the number of each class requests, and quantum size. By logging the change in remaining sojourn time in the simulation program, performance measures of practical interest such as the mean sojourn time for a request, mean waiting time in the service waiting queue, and loss probability may be evaluated. In the simulation program (Figure 1), the variable time increment method is used. In this method, the simulation time is skipped until the next event that causes a change in a system state occurs in order to shorten the simulation execution time. Events that can cause a system state change in the simulation of the prioritized, limited RR system are discussed in the following sections [10].

(1) Arrival of a request

At the arrival of a class-k request in the RR server, the number of quanta required to complete the service is obtained from the requested service time, S_r , divided by the quantum size, Qs. The time from the first quantum allocation to the end of service, S_e (the remaining sojourn time), is obtained from the time required to complete the RR cycles plus the portion of service time left. Each RR cycle includes m_k quanta allocated to each class-k request currently receiving service. The number of RR cycles needed to complete the service time is calculated by floor($Sr/(m_k * Qs)$); that is, let Cl denotes the time length of an RR cycle ($= Qs * \sum_{j=1}^{N} m_j * n_j$). Then,

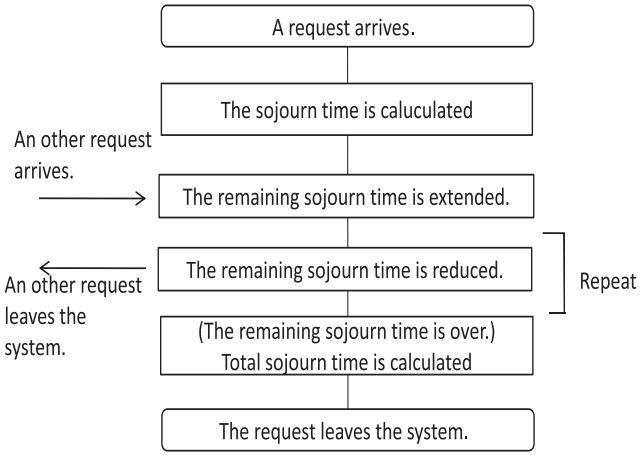

Figure 1 Simulation flow

$$Se = floor(Sr/(m_k * Qs)) * Cl + Sr - floor(Sr/(m_k * Qs)) * m_k * Qs. \quad (1)$$

Here, the operation floor(value) returns the next lowest integer value by rounding down, if necessary.

The sojourn time of other requests currently receiving service is extended by the addition of quanta inserted at each RR cycle for arriving requests. This extension of the remaining sojourn time of each request is obtained by evaluating the number of RR cycles included in the remaining sojourn time of other requests currently receiving service. Therefore, the remaining sojourn time of the requests receiving service is extended for the arrival of a class-k request according to

$$S_e = S_o + \text{ceil}(S_o / Cl) * m_k * Qs, \quad (2)$$

where S_o is the remaining sojourn time of each request currently receiving service just before the first quantum is allocated to an arriving request. Moreover, the function ceil(value) returns the next highest integer value by rounding up, if necessary.

(2) End of service

At the end of service for a request, the quantum allocated to this request is removed from the existing RR cycle. The reduction of remaining sojourn time of each request is also obtained by

evaluating the number of RR cycles included in the remaining sojourn time. Therefore, at the end of service for a class-k request, the remaining sojourn time of each request currently receiving service is shortened by

$$S_{e1} = S_o - \text{floor}(S_o/Cl) * m_k * Qs. \quad (3)$$

Moreover, when the portion of quanta allocated to the service end request is included in the remaining sojourn time of other requests, this portion must be subtracted from the remaining sojourn time of other requests (Figure 2) ; that is, if So - floor$(So / Cl) * Cl > Cl - Qs * m_k$,

$$S_e = S_{e1} - (So - \text{floor}(So / Cl) * Cl - (Cl - Qs * m_k)). \quad (4)$$

Otherwise, when So - floor$(So / Cl) * Cl <= Cl - Qs * m_k$,

$$S_e = S_{e1}. \quad (5)$$

Then, in the waiting system, a request in the service waiting queue is removed and begins receiving service.

Tracking these events and calculations enables us to evaluate practical performance measures such as the loss probability, waiting time in the service waiting queue, and mean sojourn time for requests.

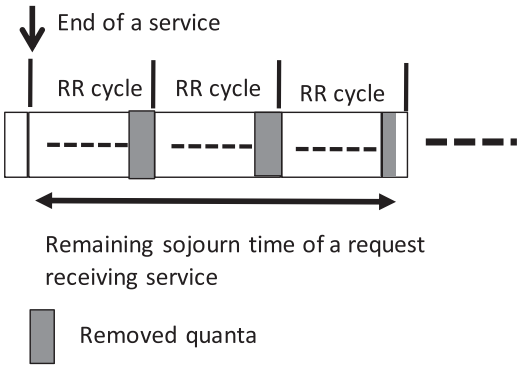

Figure 2 Service end in the middle of quanta

2.3 Simulation clock management

In the simulation program, the simulation clock is controlled by the arrival timer, or service timer of each request that is receiving service. At the arrival of each class request, the time duration until the next arrival of a request is inserted into the arrival timer according to the predetermined arrival time distribution. Moreover, the arrival time of each request is memorized in the corresponding variable. At the start of service of each class request, the service time (e.g., remaining sojourn time) of each arriving request is input into the service timer (Equation 1). When the request has been picked up from the service waiting queue, the waiting time of each request is evaluated using the arrival time and service start time. The service time of each request in the server is evaluated using these data and the service end time. In the while loop of this simulation program, the arrival processing, service start process, or service end processing mentioned in Section 2.2 is executed on the expiry of one of the arrival timers or service timers. Moreover, the service timer, or arrival timer with the next smallest value, is detected, and the time duration of this timer is subtracted from all the remaining timers. Therefore, in the next while loop, this timer expires. Simultaneously, the simulation clock is pushed forward by this time duration in order to skip the insignificant simulation clock. The while loop is repeated until the number of arriving requests attains a predetermined value.

3 Evaluation results

In the evaluation, class-1 (m_1=3) requests, class-2 (m_2=2) requests, and class-3 (m_3=1) requests were assumed to be served in a server. The arrival rate, or service rate of each priority class request, was assumed to be the same value. The two-stage Erlang inter-arrival time distribution and the exponential service time distribution were considered. Evaluation results were obtained from the average of ten simulation results. Approximately 140,000 requests were produced for each class in each run. In the evaluation results mentioned below, *Ar, S*, and SFC represent the arrival rate, mean requested service time, and service-facility capacity, respectively. The mean sojourn time, loss probability, and mean waiting time when the quantum size is 0 were obtained through the simulation of the processor sharing (PS) system [6].

3.1 Loss system

Figure 3 shows the relationship between the mean sojourn of class-1 requests (round markers), class-2 requests (cross markers), and class-3 requests (squarer markers) and the quantum size for the case when S=2, Ar=0.1 (straight lines) and S=1, Ar=0.2 (dashed lines). In each case, the value of $Ar * S$ is identical and equal to 0.2. In this figure, the range of the markers includes 95% of the reliability intervals obtained from the ten simulation runs. In these evaluation, the loss probability is negligible because SFC is sufficiently large (=30). The mean sojourn time increases as the quantum size increases. The mean sojourn time of each class request when S=1 is smaller than that when S=2. Moreover, the mean sojourn time when S=1 increases more rapidly than when S=2. In general, if the value of $Ar * S$ is constant, the mean sojourn time of the request with a smaller requested service time is more susceptible to the influence of the quantum size than when a larger service time is requested. Furthermore, the mean sojourn time of lower priority requests is more susceptible to the influence of the quantum size than that of higher priority requests.

Figure 4 shows the relationship between the increase ratio of the mean sojourn time,

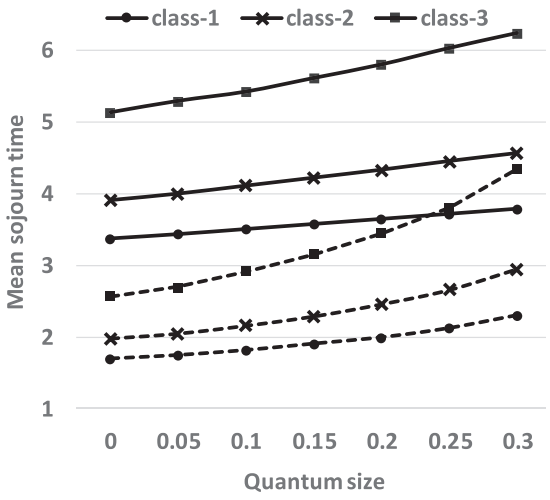

Figure 3 Mean sojourn time versus quantum size in a loss system (SFC=30)

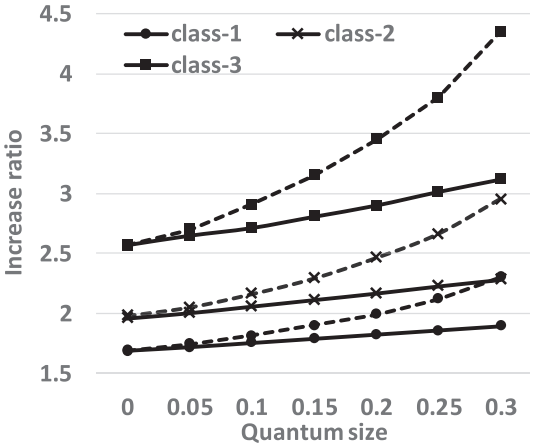

Figure 4 Increase ratio of the mean sojourn time versus quantum size in a loss system (SFC=30)

which is given by the obtained mean sojourn time divided by the requested service time, and the quantum size. As the quantum size becomes larger, the increase ratio of the mean sojourn time becomes larger. When the mean requested service time becomes smaller, the increase ratio of the mean sojourn time also becomes larger.

Figure 5 shows the relationship between the loss probability of each class request and the quantum size when SFC =12. The marker and line styles are the same as those used in Figure 3. The logarithm of the loss probability increases almost linearly as the quantum size increases. The loss probability when Ar=0.2 is larger than that when Ar=0.1. Moreover, the difference of the loss probabilities when Ar=0.2 and Ar=0.1. becomes larger as the quantum size increases.

Figure 6 compares the relationship between the mean sojourn time of each class request and the quantum size when SFC =12. The marker and line styles are the same as those used in Figure 3. The mean sojourn time also increases almost linearly as the quantum size increases. However, the influence of the quantum size on the mean sojourn time becomes smaller than when the loss probability is negligible (Figure 3). Note that the values extrapolated when the simulation results for quantum size 0 are consistent with the results obtained by the simulation of the PS system (Figures 3 - 6). Therefore, there is no contradiction between simulations of the RR system and PS system.

Figure 7 compares the relationship between the loss probability of each class request and the service-facility capacity when Q_S=0.2. The marker and line styles are the same as those used in Figure 3. The loss probability

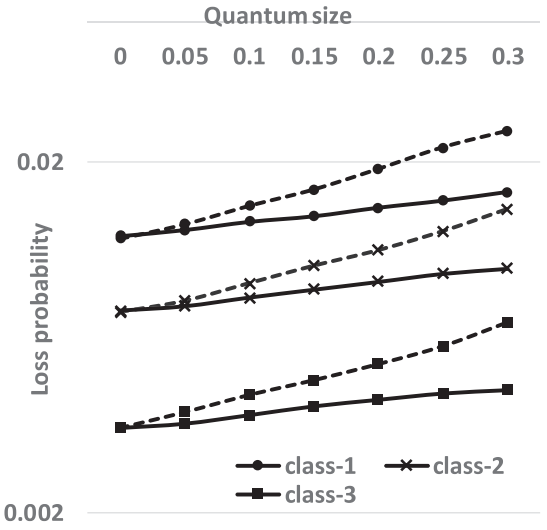

Figure 5 Loss probability versus quantum size in a loss system (SFC=12)

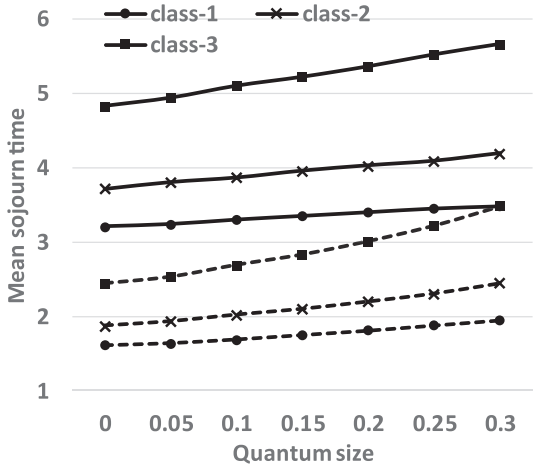

Figure 6 Mean sojourn time versus quantum size in a loss system (SFC=12)

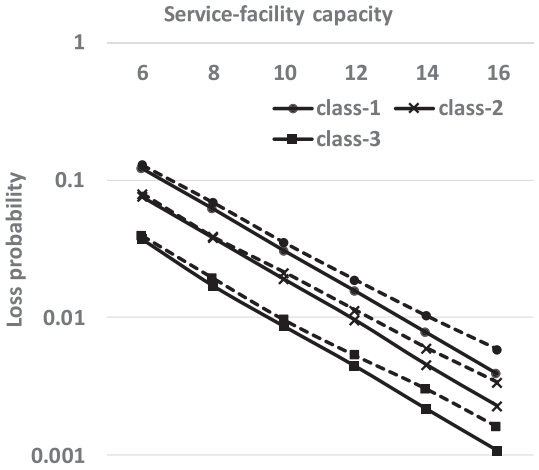

Figure 7 Loss probability versus service-facility capacity in a loss system (Q_S=0.2)

decreases almost linearly as the service-facility capacity increases. As the service-facility capacity increases, the difference in the loss probability between when S=1 and S=2 becomes larger.

Figure 8 compares the relationship between the mean sojourn time of each class request and the service-facility capacity when Q_S=0.2. Again, the marker and line styles are the same as those used in Figure 3. The mean sojourn time increases as the service-facility capacity increases. Moreover, as the service-facility capacity increases, the mean sojourn time of the request when S=2 increases more rapidly than when S=1.

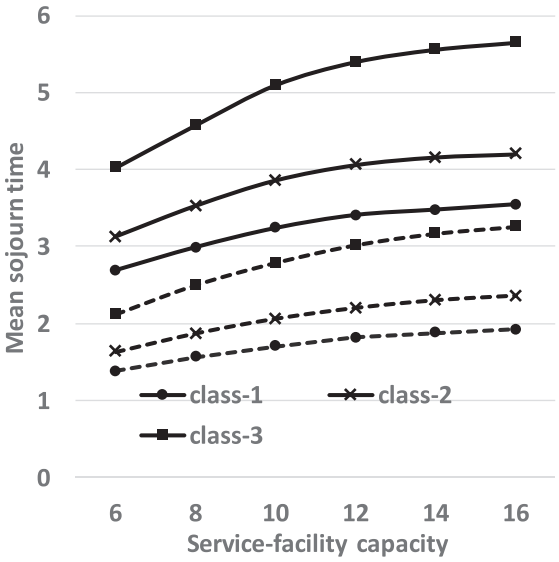

Figure 8 Mean sojourn time versus service-facility capacity in a loss system (Qs=0.2)

3.2 Waiting system

Figures 9 and 10 show the relationship between the mean sojourn times and mean waiting times in the service waiting queue and the quantum size in the waiting system. The service-facility capacity is assumed to be 12. The mean sojourn time increases almost linearly as the quantum size increases. The mean waiting time also increases as the quantum size increases. The mean waiting time of a higher class request is larger than that of a lower class request and increases more rapidly than that of lower class requests as the quantum size increases. Moreover, as the quantum size increases, the mean waiting time of a request when S=1 increases more rapidly than when S=2.

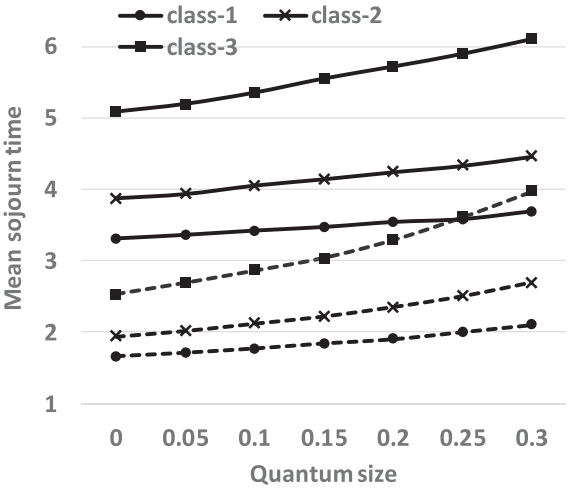

Figure 9 Mean sojourn time versus quantum size in a waiting system (SFC=12)

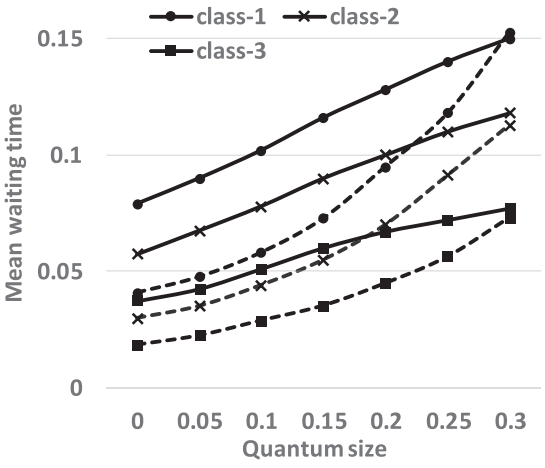

Figure 10 Mean waiting time versus quantum size in a waiting system (SFC=12)

Figures 11 and 12 show the relationship between the mean sojourn times and mean waiting times in the service waiting queue and the service-facility capacity in the waiting system. The quantum size is assumed to be 0.2. The mean sojourn time increases as the service-facility capacity increases. The mean waiting time increases as the service-facility

capacity decreases. As the service-facility capacity decreases, the mean waiting time of the request when S=2 increases more rapidly than when S=1. Moreover, as the service-facility capacity decreases, the mean waiting time of a higher class request increases more rapidly than that of a lower class request.

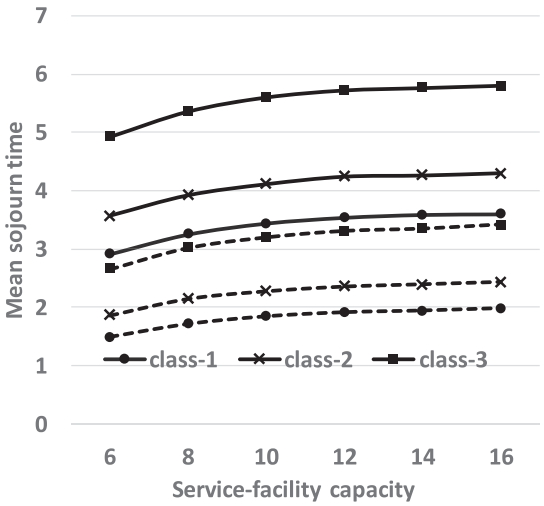

Figure 11 Mean sojourn time versus service -facility capacity in a loss system (Q_S=0.2)

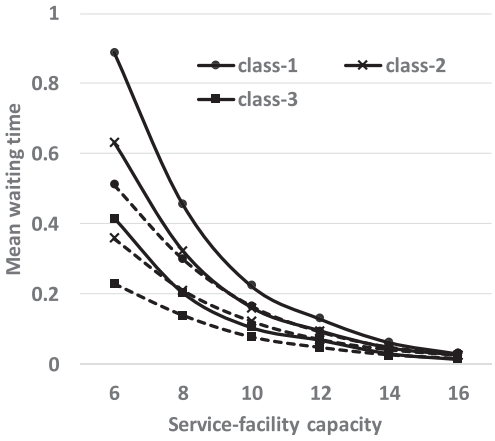

Figure 12 Mean waiting time versus service-facility capacity in a loss system (Q_S=0.2)

4 Conclusion

Practical performance measures of the prioritized, limited RR system, where N priority classes permitted, in the case that a portion of quanta must be subtracted from the remaining sojourn time are evaluated via simulation such as the mean sojourn time in the server, mean waiting time in the service waiting queue, and loss probability were evaluated using simulation programs. In these programs, the sojourn time of an arriving request was evaluated using the requested service time, number of quanta included in each RR cycle, and quantum size. Then, changes in the number of quanta needed before service is complete were reevaluated at the arrival or departure of other requests. It is clear that the mean sojourn time and loss probability increases in a loss system as the quantum size increases. Furthermore, if the value of $Ar * S$ is constant, the mean sojourn time of a request with a smaller requested service time is susceptible to the influence of the quantum size than that when a larger requested service time is considered. The mean sojourn time of each class request decreases as the service-facility capacity decreases. On the other hand, in the waiting system, the mean waiting time of each class request increases as the service-facility capacity decreases. In the future, we plan to study the performance of a prioritized PS or RR system with a maximum permissible sojourn time.

References

1. L. Kleinrock, “Time-Shared Systems: A Theoretical Treatment”, J.A.C.M Vol.1, No.14, 242-261 (1967).
2. G.Fayolle and I.Mitrani, “Sharing a Processor Among many Job Classes”, J.A.C.M Vol.27, No.3, July 1980. Pp519-532
3. E. Altman, K. Avrachenkov and U. Ayesta, “A survey on processor sharing”, Queueing Syst (2006) 53:53-63
4. M. Haviv and J. Val, “Mean sojourn times for phase-type discriminatory processor sharing system”, European

Journal of Operational Research, 189(2008), 375-386

5. G. Yamazaki and H. Sakasegawa, “An optimal design problem for limited sharing systems, Management Science”, vol.33(8), pp.1010--1019 (1987).
6. Y. Shikata, W. Katagiri, and Y. Takahashi, “Prioritized Limited Processor-Sharing System with its Performance Analysis”, International Conference on Operations Research, August 30 - 1, 2011 Zurich.
7. Varun Gupta,”Finding the optimal quantum size: Sensitivity analysis of the M/G/1 round-robin queue”, ACM SIGMETRICS Performance Evaluation Review archive Volume 36 Issue 2, September 2008 Pages 104-106
8. Varun Gupta, Jim Dai, MorHarchol-Balter, and BertZwart, “The effect of higher moments of job size distribution on the performance of an M/G/K queueing system”, Technical Report CMU-CS-08-106, School of Computer Science, Carnegie Mellon University, 2008.
9. Ward Whitt, “On approximations for queues, I: Extremal distributions”, AT&T Bell Laboratories Technical Journal, 63:115–138, 1984.
10. Y.Shikata and N.Hanayama, ”Performance Evaluation of a Prioritized, Limited RR system”, Journal of Informatics for Arts, No25 Page 75-85, Shobi University, 2016