PAPER Special Section on Information Theory and Its Applications

The Effect of Message-Class Dependent Threshold-Type Scheduling on the Delay for the M/M/n Queue

Iwao SASASE†, Yoshifumi NISHIO†, *Members* *and* Hitomi NAKAMURA†*, *Associate Member*

SUMMARY The effect of mesage-class dependent threshold-type scheduling on queueing delay and resequencing delay for the $M/M/n$ queueing system is analyzed. We first derive the expressions for the state transition equations, mean queueing delay, resequencing delay and total delay for the $M/M/n$ queueing system shared by C different message classes under a threshold-type scheduling in which the threshold values depend on the message class at the head of queue and the number of messages in the buffer. Next, the numerical calculation and the computer simulation are carried out for the queueing system with two servers. It is found that the message-class dependent threshold-type scheduling is effective to reduce the resequencing delay of some specific message class, which can not be attained under the conventional threshold-type scheduling, and thus, the proposed scheduling can satisfy the different requirements of the different message classes, such as minimizing only queueing delay or total delay including resequencing delay.

key words: *M/M/n queue, threshold-type scheduling, queueing delay, resequencing delay*

1. Introduction

Multiple-server service centers are frequently encountered in modeling communication networks. A typical example is the transmission group (TG) in IBM's system network architecture (SNA)[1]. For reasons of higher reliability and efficiency, pairs of nodes in these networks are connected by one or more logical channels consisting of multiple physical links, and messages are permitted to use parallel links possibly of different speeds while going over a logical channel. Since the number of available parallel links is finite, messages that arrive while links are occupied have to be queued in the buffer. Thus, there lies the dynamic routing problem, where the appropriate communication links have to be assigned for the messages waiting in the buffer in order to optimize system performance such as minimizing the mean delay of a message. The $M/M/n$ queueing system sharing a common buffer, with Poisson arrival process and exponential service times is known as a useful queueing model for evaluating the system performance in various computer systems and communication networks[(2)-(6)]. In the queueing model, the restriction that service is provided by a single server without interruption is imposed. This is motivated by the fact that a message, once transmitted over one transmission line, cannot thereafter be switched to another line. Also, what is called a "server" in the model could be a communication link or a processor. A "service time" is then the time taken for the message to traverse the link or a processing time. Thus, our queueing model can be also applied to the dynamic routing problem in a multiprocessor system having different processors where a job entering the central dispatch would be assigned to one of the available processors, and in manufacturing networks where a part typically has to be processed at several work stations before it is a finished product. The goal is again to find efficient routing algorithms.

It is proven that for a queueing system with two servers of different service rates, the threshold-type scheduling is the optimum policy which minimizes the mean queueing delay which consists of the waiting time in the buffer and service time[2]. The faster server should accept a message from the buffer whenever it is available for service. On the other hand, the slower server should be utilized if and only if the number of waiting messages exceeds a certain threshold value, since the slower server needs large service time. The mean queueing delay for a queueing system with multiple servers under a threshold-type scheduling is approximated[3], and the general expressions for the state probabilities and the mean queueing delay are obtained[4]. The messages served by different servers may encounter different service time, and this might cause messages to go out of sequence when they completes their service. Since most store- and-forward computer networks require messages to put in the same order as they were generated, resequencing process is needed, in which out-of-sequence messges are stored in a resequence buffer to establish the original order. Hence, minimizing resequencing delay together with queueing delay is required. The probability distribution of the resequencing delay of the $M/M/n$ system is analyzed[5]. The mean resequencing delay for a queueing system with multiple servers of different service rates is obtained[6]. The effect of multiple

Manuscript received January 30, 1992.
Manuscript revised April 22, 1992.

† The authors are with the Faculty of Science and Technology, Keio University, Yokohama-shi, 223 Japan.
* Presently, with All Nippon Airways.

message classes on the resequencing delay in a system with two parallel $M/M/1$ queues without a threshold-type scheduling is analyzed[7]. However, in these systems the threshold-type policy is not employed. The mean resequencing dalay for a queueing system with two servers under a threshold-type scheduling is analyzed and the optimal conditions for minimizing the mean resequencing delay and the proportion of messages which experience the resequencing delay are obtained[8]. The effect of the threshold-type scheduling on resequencing delay for a queueing system with multiple servers under a threshold-type scheduling is also analyzed[9]. Recently, the effect of multiple classes of arrival messages on the resequencing delay for a queueing system under a threshold-type scheduling is analyzed[10]. However, in these analyses, the threshold-type scheduling does not depend on the message class, and thus, it is impossible to give priority to some specific message classes. The requirements of different message classes are generally different, for example, some message classes need to minimize only queueing delay, whereas the minimizing resequencing delay as well as queueing delay is required for some message classes. Therefore, it is of great interest to consider the threshold-type scheduling in which the control to give priority of some specific message classes can be attained for a queueing system with multiple servers.

In this paper, the effect of message-class dependent threshold-type scheduling on the queueing delay and resequencing delay for the $M/M/n$ queueing system is analyzed to consider the feasibility of giving priority of some specific message classes. We first derive the general expressions for the state transition equations, mean queueing delay, resequencing delay and total delay for the $M/M/n$ queueing system shared by C different message classes under a threshold-type scheduling in which the threshold values depend on the message class at the head of queue and the number of messages in the buffer. Next, the numerical calculation is carried out for the queueing system with two servers under a message-class dependent threshold-type scheduling to consider the effect of the message-class dependent threshold-type scheduling. It is found that the message-class dependent threshold-type scheduling is effective to reduce the resequencing delay of some specific message class, which can not be attained under the conventional threshold-type scheduling, and thus, the proposed scheduling can satisfy the different requirements of the different message classes, such as minimizing only queueing delay or total delay including resequencing delay.

2. System Model

Figure 1 shows the system model of the $M/M/n$ queueing system under a message-class dependent threshold-type scheduling. It is assumed that a total

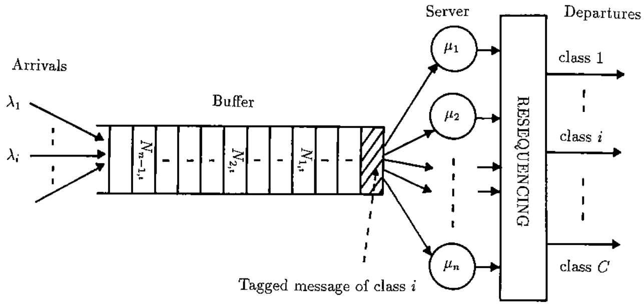


Fig. 1 System model of a $M/M/n$ queueing system under a message-class dependent threshold-type scheduling.

number of C different classes of the messages arrive independently at the source node, and that the messages belonging to class $i\,(1 \leqq i \leqq C)$, arrive according to a Poisson process with arrival rate λ_i. The total arrival rate λ is

$$\lambda = \sum_{i=1}^{C} \lambda_i \tag{1}$$

and $\beta_i = \lambda_i / \lambda$ is the probability that the incoming message belongs to class i. It is assumed that messages are queued in an infinite buffer with each message occupying one unit of the buffer. The messages are dispatched to servers according to the message-class dependent threshold-type scheduling which is described later. Each message has to complete all its service at only one server without interruption. The service time of server j is exponentially distributed with mean $\mu_j^{-1}\,(j=1, \cdots, n)$. Without loss of generality, we assume $\mu_1 \geqq \mu_2 \geqq \cdots \geqq \mu_n$. To ensure stability we shall also assume $\lambda < \mu_1 + \cdots + \mu_n$. The traffic intensity ρ is defined as

$$\rho = \frac{\lambda}{\mu_1 + \cdots + \mu_n}. \tag{2}$$

In the message-class dependent threshold-type scheduling, the thresholds depend on the class of the message at the head of the queue in the buffer, and the threshold values are changed according to the message class to give priority to the some specific message classes. A queue discipline of the message-class dependent threshold-type sheduling for a queueing system with multiple servers of possibly different service rates is described as follows: Whenever server 1 becomes available, it starts serving the first message in the buffer regardless of the total number of waiting messages, since server 1 gives fastest service on the average. When server f ($f=2, \cdots, n$) becomes available and servers 1, $\cdots, f-1$ are busy, server f starts serving a new message of class $i\,(i=1, \cdots, C)$ taken from the first position in the queue if and only if the number of waiting messages exceeds the corresponding threshold $N_{(f-1),i}$. Here, $N_{1,i} \leqq N_{2,i} \leqq \cdots \leqq N_{(n-1),i}$ is assumed. The threshold values are changed depending on the class of the message at the head of the queue in the buffer. All waiting messages which have arrived after this message, move up one position. Messages may complete their service in a different order from the one in which they enter the multiple-server service center. It the message bypasses the message of the same class, which arrived earlier in the system and are still being served by the other servers, the message will experience resequencing delay to establish the original order. As far as messages belonging to a given class are concerned, for example class i, messages of the rest $C-1$ classes play the role of interfering traffic. Thus, in the system with multiple message classes, resequencing delay is defined as the time a completed message has to wait for all bypassed messages of the same class until they complete their service. Queueing delay consists of waiting time in the buffer and service time. The total delay of a message is the sum of its queueing dalay and its resequencing delay in the queueing system.

Let the tagged message be a message which is about to begin transmission from the head of queue in the buffer just after a message arrives in the system. Here, it is assumed that the tagged message belongs to class i, and is sent to server r when a set of servers, denoted as $\boldsymbol{B}$, is already busy serving messages. Due to the memoryless property of the transmission times, the probability that a message is still being served by server j after an elapsed time t is $\exp(-\mu_j t)$, and the probability that a message on server f completes its service after an elapsed time t is $1-\exp(-\mu_f t)$. Since the message transmission times are all mutually independent random variables, and the transmission time density function of the tagged message on server r is expressed as $\mu_r \exp(-\mu_r t)$, the probability of the event that the tagged message bypasses the messages in $\boldsymbol{U}$, given that the servers in set $\boldsymbol{B}$ are busy, is expressed as[6]

$$\begin{aligned}
G(\boldsymbol{B}, \boldsymbol{U}, r) &= \int_0^\infty \Big(\prod_{j\in U} \exp(-\mu_j t)\Big) \cdot \Big(\prod_{f\in B-U} (1-\exp(-\mu_f t))\Big) \cdot \mu_r \exp(-\mu_r t)\, dt \\
&= \left(\frac{\mu_r}{\mu_k + \sum_{j\in U}\mu_j}\right) \cdot \sum_{f\in B-U}\left[\left(\frac{\mu_f}{\mu_r+\mu_f+\sum_{j\in U}\mu_j}\right) \cdot \sum_{m\in B-U-\{f\}}\left[\left(\frac{\mu_m}{\mu_r+\mu_f+\mu_m+\sum_{j\in U}}\right) \cdots \sum_{y\in B-U-\{f\}-\{m\}-\cdots}\left(\frac{\mu_y}{\mu_r+\sum_{j\in B}\mu_j}\right)\right]\cdots\right]
\end{aligned} \quad (3)$$

where $\boldsymbol{B}$, $\boldsymbol{U}$, and r in the notation of $G(\boldsymbol{B}, \boldsymbol{U}, r)$ denote the set of busy servers excluding the server which serves the tagged message, the set of servers whose message are bypassed by the tagged message, and the server which serves the tagged message, respectively. The expressions for $G(\boldsymbol{B}, \boldsymbol{U}, r)$ when one or two servers are busy are shown in detail in Ref.(9). The mean waiting time of the tagged message for the bypassed messages in $\boldsymbol{U}$ is obtained by calculating the complementary cumulative density function (ccdf) of the waiting time of the tagged message. Since the exponential time distribution has the memoryless property and the cumulative density functions of the waiting time of the messages are mutually independent, the mean waiting time for the bypassed messages in $\boldsymbol{U}$, is expressed as[6]

$$\begin{aligned}
V(\boldsymbol{U}) &= \int_0^\infty \Big[1-\prod_{j\in U}(1-\exp(-\mu_j t))\Big] dt \\
&= \Big(\sum_{j\in U}\frac{1}{\mu_j}\Big) - \Big(\sum_{\substack{j,g\in U\\ j\neq g}}\frac{1}{\mu_j+\mu_g}\Big) + \cdots + (-1)^{\|U\|+1}\Big(\frac{1}{\sum_{j\in U}\mu_j}\Big)
\end{aligned} \quad (4)$$

where $\|\boldsymbol{U}\|$ denotes the number of elements in set $\boldsymbol{U}$. Under a message-class dependent threshold-type scheduling, the on-off status of the servers, the number of messages in the buffer, and the class of the tagged message determine the server to which the tagged message is sent. Thus, the resequence depends on both of the on-off status of the servers at the transmission of the tagged message and the completion of its service. Under a message-class dependent threshold-type scheduling, the resequencing delay also depends on the class of the tagged message and classes of the bypassed messages, since the threshold values are changed according to the tagged message class and the tagged message experiences resequencing delay if it bypasses the messages of the same class.

3. State Transition Equations

We derive the expressions for the steady-state state transition equation for the state probability in the $M/M/n$ queueing system shared by C different classes of the arrival messages under a message-class dependent threshold-type scheduling. Suppose m servets are busy and m-m servers are idle when the tagged message of class i is about to begin transmission from the head of the queue. Let $p(i, k, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ be a steady-state probability of the state $S(i, k, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$, where k is the number of messages in the buffer, $\boldsymbol{B}_m$ is the set of m busy servers, and $\boldsymbol{B}_m^g$ denotes the set of g busy servers in $\boldsymbol{B}_m$ which serve the messages of the same class i as the tagged message. According to the number of messages in the buffer and the status of the servers, we can obtain the state transition equation for the state probability when m servers are busy at the transmission of the tagged message of class i by classifying into three cases as follows:

3.1 $S(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$

Let us consider the equation for the state transition of $S(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$, where there is no message in the buffer, the set of m servers excluding server 1, denoted as $\boldsymbol{B}_m=\{b_1, b_2, \cdots, b_m\}$ is busy, and the subset with g servers in $\boldsymbol{B}_m$, denoted as $\boldsymbol{B}_m^g$ is serving the messages of class i. The set of $n-m$ idle servers is written as $\boldsymbol{Z}_{n-m}=\boldsymbol{D}_n-\boldsymbol{B}_m$, where $\boldsymbol{D}_n=\{1, 2, \cdots, n\}$. When there are no messages waiting in the buffer, the class of an incoming tagged message is assumed to be proportional to $\beta_i=\lambda_i/\lambda$. Thus, the state probability $p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ can be expressed as

$$p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)=\beta_i p(0, \boldsymbol{B}_m, \boldsymbol{B}_m^g). \tag{5}$$

where $p(0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ is the state probability of the state $S(0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$, which is independent of the class of the incoming tagged message.

First, we consider the state transitions from $S(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ to other states. When a new message arrives in the system or any one of busy servers in $\boldsymbol{B}_m$ completes the service of the message, the state transition from the state $S(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ is occurred. The corresponding state transition probability is expressed as

$$(\lambda+\mu_{b1}+\cdots+\mu_{bm})p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g). \tag{6}$$

Next, we consider the state transitions to $S(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ from other states. Since no message is waiting in the buffer and server 1 is idle, the transition to this state is occurred only from the states $S(i, 0, \boldsymbol{B}_m+\{za\}, \boldsymbol{B}_m^g)$ and $S(i, 0, \boldsymbol{B}_m+\{\boldsymbol{za}\}, \boldsymbol{B}_m^g+\{za\})$, when the server za included in $\boldsymbol{Z}_{n-m}$ completes the service of message. The corresponding state transition probability is expressed as

$$\sum_{a=1}^{n-m}\mu_{za}[p(i, 0, \boldsymbol{B}_m+\{za\}, \boldsymbol{B}_m^g)+p(i, 0, \boldsymbol{B}_m+\{za\}, \boldsymbol{B}_m^g+\{za\})]. \tag{7}$$

Note that no transitions to $p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ due to the new message arrival or the simultaneous transmission of the consecutive messages in the buffer with the tagged message are occurred, since server 1 has to be idle. In the steady-state, the following equation for the state transition of $p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ is satisfied.

$$(\lambda+\mu_{b1}+\cdots+\mu_{bm})p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g) =\sum_{a=1}^{n-m}\mu_{za}[p(i, 0, \boldsymbol{B}_m+\{za\}, \boldsymbol{B}_m^g) +p(i, 0, \boldsymbol{B}_m+\{\boldsymbol{za}\}, \boldsymbol{B}_m^g+\{za\})]. \tag{8}$$

3.2 $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$

Let us consider the equation for the state transition of $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$, where there is no message in the buffer, m servers of the sets of $\boldsymbol{D}_d=\{1, 2, \cdots, d\}$ and $\boldsymbol{B}_{m-d}(d+2\leqq b1<\cdots<b(m-d)\leqq n)$ are busy, and the subset with g servers $\boldsymbol{B}_m^g$ included in $\boldsymbol{D}_d+\boldsymbol{B}_{m-d}$ is serving the messages of class i. The state transition probability from $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ to other states is expressed as

$$(\lambda+\mu_1+\cdots+\mu_d+\mu_{b1}+\cdots+\mu_{b(m-d)}) p(i, 0, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g). \tag{9}$$

Next, we consider the state transitions to $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ from other states. Four cases of transitions to this state are occurred. The first case corresponds to the state transition due to the arrival of a new message. The new arrival message is accepted by only server 1 if the threshold $N_{(f-1),S_f}>0(f=2, \cdots, d)$, where as server f may accept the new arrival message if the tagged message belongs to class S_f and the corresponding threshold $N_{(f-1),S_f}$ equals to 0. The state transition probability due to the arrival of a new message is expressed as

$$\lambda\beta_i(S_1, 0, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}-\{1\}, \boldsymbol{B}_m^g-\{\bar{1}\}) +\sum_{f=2}^{d}F(N_{(f-1),S_f})\lambda\beta_i p(S_f, 0, \boldsymbol{D}_d+\boldsymbol{B}_{m-d} -\{f\}, \boldsymbol{B}_m^g-\{\bar{f}\}) \tag{10}$$

where

$$F(n)=\begin{cases}1 & \text{if } n=0\\ 0 & \text{otherwise}\end{cases} \tag{11}$$

$$\{\bar{f}\}=\begin{cases}\{f\} & \text{if message on server } f \text{ belongs to class } i\\ \phi & \text{if a message on server } f \text{ belongs to class } j(\neq i).\end{cases} \tag{12}$$

The second case of the state transition to $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ corresponds to that due to the completion of the service of the message and simultaneous acceptance of a new message by server 1, or server f $(f=2, \cdots, d)$ if the class of the tagged message is S_f and the number of messages exceeds the threshold $N_{(f-1),S_f}$. The corresponding state transition probability due to the completion of the service of the message and simultaneous acceptance of a new message is expressed as

$$\mu_1\beta_i[p(S_1, 1, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g-\{\bar{1}\}) +p(S_1, 1, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g-\{\bar{1}\}+\{1\})]$$

$$+\sum_{f=2}^{d} L(-N_{(f-1),S_f})\mu_f\beta_i[p(S_f, 1, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g - \{\bar{f}\}) + p(S_f, 1, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g - \{\bar{f}\} + \{f\})] \quad (13)$$

where

$$L(n) = \begin{cases} 1 & \text{if } n \geqq 0 \\ 0 & \text{otherwise.} \end{cases} \quad (14)$$

The third case of the state transition to $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ corresponds to that due to the completion of service of the message by server $za(a=1, \cdots, n-m)$ included in $\boldsymbol{D}_n - \boldsymbol{D}_d - \boldsymbol{B}_{m-d}$, and the state transition probability due to the completion of service of the message is expressed as

$$\sum_{a=1}^{n-m} \mu_{za}(p(i, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} + \{za\}, \boldsymbol{B}_m^g) + p(i, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} + \{za\}, \boldsymbol{B}_m^g + \{za\})). \quad (15)$$

The fourth case of the state transition to $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ corresponds to that due to the simultaneous transmission of the consecutive message(s) in the buffer with the tagged message. Since the threshold values depend on the class of the tagged message, the simultaneous message transmission of the consecutive message(s) just after the tagged message may be occurred. Suppose the state for an $M/M/2$ queueing system is $S(i, k, j, 0)$, where $j=1, \cdots, C$, and $N_{1,m} \leqq k \leqq N_{1,i}$, for example. When the server 1 completes its service, then immediately the tagged message of class i is sent to server 1. If the successive message just after the tagged message belong to class m, then it is sent simultaneously to server 2, respectively, since server 2 is idle and the corresponding number of message in the buffer at the transmission of the consecutive message, k, exceeds the corresponding message-class dependent threshold $N_{1,m}$. The state transition to $S(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ due to the simultaneous transmission of the $e-1(2 \leqq e \leqq d)$ consecutive messages, denoted as $\{S_{e2}, \cdots, S_{ee}\}$, in the buffer with the tagged message S_{e1} is occurred, when server $e1$ completes its service, the tagged message and the following $e-1$ consecutive messages can be sent to the corresponding server $\{e1, e2, \cdots, ee\}$, respectively. The condition of the successive transmission of the $e-1$ consecutive messages is that the numbers of the messages at the transmission of the consecutive messages exceed the corresponding thresholds, respectively, that is, the product of $L(e-1-N_{(e1-1),S_{e1}})L(e-2-N_{(e2-1),S_{e2}})\cdots L(-N_{(ee-1),S_{ee}})$ must be 1. Thus, the state transition probability due to the simultaneous transmission of the consecutive message(s) is expressed as

$$\sum_{e=2}^{d}\sum_{E_e \in D_d} L(e-1-N_{(e1-1),S_{e1}}) L(e-2-N_{(e2-1),S_{e2}}) \cdots L(-N_{(ee-1),S_{ee}})\mu_{e1}\beta_{S_{e2}}\cdots\beta_{S_{ee}}\beta_i (p(S_{e1}, e, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} - \boldsymbol{E}_e + \{e1\}, \boldsymbol{B}_m^g - \bar{\boldsymbol{E}}_e) + p(S_{e1}, e, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} - \boldsymbol{E}_e + \{e1\}, \boldsymbol{B}_m^g - \bar{\boldsymbol{E}}_e + \{e1\})) \quad (16)$$

where $\boldsymbol{E}_e = \{e1, e2, \cdots, ee\}$ is the set of e servers included in $\boldsymbol{D}_d$. Thus, in the steady-state, the following equation for the state transition of $p(i, 0, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ is satisfied.

$$\begin{aligned}
&(\lambda + \mu_1 + \cdots + \mu_d + \mu_{b1} + \cdots + \mu_{b(m-d)}p(i, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g) \\
&= \lambda\beta_i p(S_1, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} - \{1\}, \boldsymbol{B}_m^g - \{\bar{1}\}) \\
&\quad + \sum_{f=2}^{d} F(N_{(f-1),S_f})\lambda\beta_i p(S_f, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} - \{f\}, \boldsymbol{B}_m^g - \{\bar{f}\}) \\
&\quad + \mu_1\beta_i(p(S_1, 1, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g - \{\bar{1}\}) + p(S_1, 1, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g - \{\bar{1}\} + \{1\})) \\
&\quad + \sum_{f=2}^{d} L(-N_{(f-1),S_f})\mu_f\beta_i(p(S_f, 1, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g - \{\bar{f}\}) + p(S_f, 1, \boldsymbol{D}_d + \boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g - \{\bar{f}\} + \{f\})) \\
&\quad + \sum_{a=1}^{n-m} \mu_{za}(p(i, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} + \{za\}, \boldsymbol{B}_m^g) + p(i, 0, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} + \{za\}, \boldsymbol{B}_m^g + \{za\})) \\
&\quad + \sum_{e=2}^{d}\sum_{E_e \in D_d} L(e-1-N_{(e1-1),S_{e1}}) L(e-2-N_{(e2-1),S_{e2}}) \cdots L(-N_{(ee-1),S_{ee}})\mu_{e1}\beta_{S_{e2}}\cdots\beta_{S_{ee}}\beta_i \\
&\quad (p(S_{e1}, e, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} - \boldsymbol{E}_e + \{e1\}, \boldsymbol{B}_m^g - \bar{\boldsymbol{E}}_e) + p(S_{e1}, e, \boldsymbol{D}_d + \boldsymbol{B}_{m-d} - \boldsymbol{E}_e + \{e1\}, \boldsymbol{B}_m^g - \bar{\boldsymbol{E}}_e + \{e1\})).
\end{aligned} \quad (17)$$

3.3 $S(i, k, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ $(N_{(h-1),i} < k \leqq N_{h,i})$

Let us consider the equation for the state transition of $S(i, k, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$, where $N_{(h-1),i} < k \leqq N_{h,i}$ $(h=1, \cdots, m-1$ and $N_{0,i}=0)$, m servers of the sets of $\boldsymbol{D}_d$ and $\boldsymbol{B}_{m-d}(d+2 \leqq b1 < \cdots < b(m-d) \leqq n)$ are busy, and the subset with g servers $\boldsymbol{B}_m^g$ included in $\boldsymbol{B}_{m-d}+\boldsymbol{D}_d$ is serving the messages of class i. Under the message-class dependent threshold-type scheduling, the set of h $(h \leqq d)$ servers, denoted as $\boldsymbol{D}_h$, has to be busy. By using the similar steps as those in Sect. 3.2, the steady-state transition equation for the state probability of $p(i, k, \boldsymbol{B}_{m-d}+\boldsymbol{D}_d, \boldsymbol{B}_m^g)$ is obtained as follows:

$$(\lambda + \mu_1 + \cdots + \mu_d + \mu_{b1} + \cdots + \mu_{b(m-d)})p(i, k, \boldsymbol{D}_d$$

$$+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g)$$
$$=\lambda p(i, k-1, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g)$$
$$+\sum_{f=2}^{d} F(N_{(f-1),S_f}-k)\lambda\beta_i p(S_f, k, \boldsymbol{D}_d$$
$$+\boldsymbol{B}_{m-d}-\{f\}, \boldsymbol{B}_m^g-\{\bar{f}\})$$
$$+\mu_1\beta_i(p(S_1, k+1, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g-\{\bar{1}\})$$
$$+p(S_1, k+1, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g-\{\bar{1}\}+\{1\}))$$
$$+\sum_{f=2}^{d} L(k-N_{(f-1),S_f})\mu_f\beta_i(p(S_f, k+1, \boldsymbol{D}_d$$
$$+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^g-\{\bar{f}\})$$
$$+p(S_f, k+1, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}, \boldsymbol{B}_m^\infty-\{\bar{f}\}+\{f\}))$$
$$+\sum_{a=1}^{n-m}\mu_{za}(p(i, k, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}+\{za\}, \boldsymbol{B}_m^g)$$
$$+p(i, k, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}+\{za\}, \boldsymbol{B}_m^g+\{za\}))$$
$$+\sum_{e=2}^{d}\sum_{E_e\in D_d} L(k+e-1-N_{(e1-1),S_{e1}})$$
$$L(k+e-2-N_{(e2-1),S_{e2}})\cdots$$
$$L(k-N_{(ee-1),S_{ee}})\mu_{e1}\beta_{S_{e2}}\cdots\beta_{S_{ee}}\beta_i$$
$$(p(S_{e1}, k+e, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}-\boldsymbol{E}_e+\{e1\},$$
$$\boldsymbol{B}_m^g-\bar{\boldsymbol{E}}_e)$$
$$+p(S_{e1}, k+e, \boldsymbol{D}_d+\boldsymbol{B}_{m-d}-\boldsymbol{E}_e+\{e1\}, \boldsymbol{B}_m^g$$
$$-\bar{\boldsymbol{E}}_e+\{e1\})). \tag{18}$$

4. Mean Queueing Delay

Now, we obtain the expression for the mean queueing delay for the $M/M/n$ queueing system under a message-class dependent threshold-type scheduling. Queueing delay consists of waiting time in the buffer and service time. The number of messages in the system is the sum of waiting messaes in the buffer and messages in the busy servers. By using the results in Sect. 3, and removing the conditions on m busy servers and class i of the tagged message, the expected number of messages in the system is obtained as follows:

$$EN=\sum_{i=1}^{C}\left\{\sum_{m=1}^{n-1}\left[\sum_{Bm\in Dn}\sum_{g=0}^{m}\sum_{B_m^g\in B_m} mp(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)\right.\right.$$
$$+\sum_{k=1}^{N_{1,i}}\sum_{B_{m-1}\in D_n-D_1}\sum_{g=0}^{m-1}\sum_{B_{m-1}^g\in B_{m-1}}(m+k)$$
$$\cdot p(i, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1, \boldsymbol{B}_{m-1}^g)$$
$$+\sum_{f=1}^{m-2}\sum_{k=N_{f,i}+1}^{n_{(f+1),i}}\sum_{B_{m-f-1}\in D_n-D_{f+1}}\sum_{d=0}^{f+1}$$
$$\sum_{g=0}^{m-f-1}\sum_{D_{f+1}^d\in D_{f+1}}\sum_{B_{m-f-1}^g\in B_{m-f-1}}(m+k)$$
$$\cdot p(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{m-f-1}^g+\boldsymbol{D}_{f+1}^d)$$
$$+\sum_{k=N_{(m-1),i}+1}^{N_{m,i}}\sum_{d=0}^{m}\sum_{D_m^d\in D_m}(m+k)$$
$$\left.\cdot p(i, k, \boldsymbol{D}_m, \boldsymbol{D}_m^d)\right]$$
$$\left.+\sum_{k=0}^{\infty}\sum_{d=0}^{n}\sum_{D_n^d\in D_n}(n+k)p(i, k, \boldsymbol{D}_n, \boldsymbol{D}_n^d)\right\} \tag{19}$$

where $\boldsymbol{B}_m^g$ is an arbitrary subset with g servers included in $\boldsymbol{B}_m$. $\sum \boldsymbol{B}_m^g \in \boldsymbol{B}_m$ denotes the summation for all combinations of $\boldsymbol{B}_m^g$ included in $\boldsymbol{B}_m$. Thus, from Little's formula, the mean queueing delay is

$$QD_{mean}=\frac{EN}{\lambda}. \tag{20}$$

In our analysis, the mean queueing delay is used as a performance criterion, instead of using 99th percentile delay which is useful in the practical system design. This is because the mean queueing delay is easily calculated and is often used as a useful measure in the analysis of queueing models where an infinite buffer and no loss probability due to buffer overflow are assumed[(3),(8)].

5. The Mean Resequencing Delay of the Tagged Message

Let us derive the general expressions for the mean resequencing delay for the $M/M/n$ queueing system shared by C different classes of the arrival messages under a message-class dependent threshold-type scheduling. Suppose m servers are busy and $n-m$ servers are idle when the tagged message of class i is about to begin transmission. Since m servers are busy, the number of messages in the buffer must be less than $M_{m,i}+1$, otherwise at least $m+1$ servers including servers $1, 2, \cdots, m+1$ should be busy. According to the class of the tagged message, the number of messages in the buffer, the status of the servers, and the bypassed messages, we can obtain the resequencing delay when m servers are busy at the transmission of the tagged message of class i by classifying into the six cases as follows:

5.1 $S(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$

Assume that there is no message in the buffer and that $\boldsymbol{B}_m$ is the set of m busy servers excluding server 1. Let $\boldsymbol{B}_m^g$ be the subset with g servers serving the messages of class i included in $\boldsymbol{B}_m$. When a new message belonging to class i arrives in the system, it is dispatched immediately to server 1 as the tagged message, since server 1 is idle. The resequencing occurs when the tagged message on server 1 completes its service earlier than any of the messages of class i being served by the set of servers $\boldsymbol{B}_m^g$. The mean resequencing delay in this case can be obtained by considering all possible situations of the on-off status of these g servers serving the messages of class i at the completion of the tagged message's service. Note that the tagged message has to wait for only the bypasses messages of the same class i for resequencing, even when it bypasses some messages of different classes. The mean resequencing delay $\boldsymbol{R}(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)$ can be obtained by using Eqs. (3) and (4) as

$$\boldsymbol{R}(i, 0, \boldsymbol{B}_m, \boldsymbol{B}_m^g)=\sum_{h=1}^{g}\sum_{\boldsymbol{B}_g^h \in \boldsymbol{B}_m^g} \boldsymbol{G}(\boldsymbol{B}_m^g, \boldsymbol{B}_g^h, 1)\, \boldsymbol{V}(\boldsymbol{B}_g^h). \tag{21}$$

5.2 $S(i, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1, \boldsymbol{B}_{m-1}^g)$

Suppose the number of waiting messages in the buffer is $k\,(0 \leqq k < N_{1,i})$ including the tagged message of class i at the head of the queue, and m servers including server 1 are busy. Under a message-class dependent threshold-type scheduling, only server 1 is allowed to accept a new message when $1 \leqq k < N_{1,i}$ and the tagged message belongs to class i. Here, let $\boldsymbol{B}_{m-1}$ be the set of $m-1$ busy servers excluding server 1 and $\boldsymbol{B}_{m-1}^g$ be the subset with g servers serving the messages of class i included in $\boldsymbol{B}_{m-1}$. The tagged message of class i is dispatched to server 1 as soon as the previous message on server 1 finishes its service. The tagged message will experience resequencing delay only if it completes its service before any of the mesages of class i being served by the set of servers $\boldsymbol{B}_{m-1}^g$. The mean resequencing delay $\boldsymbol{R}(i, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1, \boldsymbol{B}_{m-1}^g)$ is expressed as

$$\begin{aligned}\boldsymbol{R}&(i, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1, \boldsymbol{B}_{m-1}^g)\\ &=\sum_{h=1}^{g}\sum_{\boldsymbol{B}_g^h \in \boldsymbol{B}_{m-1}^g} \boldsymbol{G}(\boldsymbol{B}_{m-1}^g, \boldsymbol{B}_g^h, 1)\\ &\quad \sum_{j=1}^{h}\sum_{\boldsymbol{B}_h^j \in \boldsymbol{B}_g^h} \boldsymbol{G}(\boldsymbol{B}_g^h, \boldsymbol{B}_h^j, 1)\, \boldsymbol{V}(\boldsymbol{B}_h^j)\end{aligned} \tag{22}$$

5.3 $S(i, N_{f,i}, \boldsymbol{B}_{m-f}+\boldsymbol{D}_f, \boldsymbol{B}_m^g)$

Suppose m servers are busy and the number of waiting messages in the buffer is $N_{f,i}\,(1 \leqq f \leqq m)$, including the tagged message of class i at the head of the queue, and server $f+1$ is idle. Under a message-class dependent threshold-type scheduling, the set of servers $1, \cdots, f$, denoted as $\boldsymbol{D}_f$ have to be busy. Let $\boldsymbol{B}_{m-f}$ be the set of other $m-f$ busy servers excluding $\boldsymbol{D}_f$, and $\boldsymbol{B}_m^g$ be the subset with g servers serving the messages of class i included in $\boldsymbol{B}_{m-f}+\boldsymbol{D}_f$. When a new message arrives in the system, the number of messages waiting in the buffer exceeds $N_{f,i}$ and immediately the tagged message of class i is dispatched from the head of the queue to server $f+1$. The mean resequencing delay $\boldsymbol{R}(i, N_{f,i}, \boldsymbol{B}_{m-f}+\boldsymbol{D}_f, \boldsymbol{B}_m^g)$ is obtained as

$$\begin{aligned}\boldsymbol{R}&(i, N_{f,i}, \boldsymbol{B}_{m-f}+\boldsymbol{D}_f, \boldsymbol{B}_m^g)\\ &=\sum_{h=1}^{g}\sum_{\boldsymbol{B}_g^h \in \boldsymbol{B}_m^g} \boldsymbol{G}(\boldsymbol{B}_m^g, \boldsymbol{B}_g^h, f+1)\, \boldsymbol{V}(\boldsymbol{B}_g^h)\end{aligned} \tag{23}$$

5.4 $S(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{m-f-1}^g+\boldsymbol{D}_f^g)_a$

Suppose m servers are busy and the number of waiting messages in the buffer is $k\,(N_{f,i} \leqq k < N_{(f+1),i},\ 1 \leqq f < m-1)$, including the tagged message of class i at the head of the queue. Under a message-class dependent threshold-type scheduling, the set of servers $1, \cdots, f+1$, denoted as $\boldsymbol{D}_{f+1}$ have to be busy. Let $\boldsymbol{B}_{m-f-1}$ be the set of other $m-f-1$ busy servers excluding $\boldsymbol{D}_{f+1}$. Also, let $\boldsymbol{B}_{m-f-1}^g$ and $\boldsymbol{D}_f^d$ is the subsets with g and d servers serving the messages of class i included in $\boldsymbol{B}_{m-f-1}$ and $\boldsymbol{D}_{f+1}-\{a\}\,(\{a\} \in \boldsymbol{D}_{f+1})$, respectively. Since the number of messages waiting in the queue exceeds $N_{f,i}$, the tagged message of class i is dispatched from the head of the queue to any one of servers $1, \cdots, f+1$ depending on which server becomes idle first. Let the server a accepts the tagged message. Note that servers $b1, \cdots, b(m-f-1)$ can not accept a new message from the buffer even they become idle. Since the tagged message is accepted by server a just after the server a completes serving the message, this message does not depend on the resequencing delay, even though it belongs to class i. Thus, the corresponding state probability in this case is expressed as

$$\begin{aligned}p&(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{f-m-1}^g+\boldsymbol{D}_f^d)_a\\ &=p(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{f-m-1}^g+\boldsymbol{D}_f^d)\\ &\quad+p(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{f-m-1}^g+\boldsymbol{D}_f^d+\{a\}).\end{aligned} \tag{24}$$

In general,

$$\begin{aligned}p(i, k, \boldsymbol{B}, \boldsymbol{D})_a&=p(i, k, \boldsymbol{B}, \boldsymbol{D})\\ &\quad+p(i, k, \boldsymbol{B}, \boldsymbol{D}+\{a\}).\end{aligned} \tag{25}$$

The mean resequencing delay $\boldsymbol{R}(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{m-f-1}^g+\boldsymbol{D}_f^d)_a$ becomes

$$\begin{aligned}\boldsymbol{R}&(i, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}_{m-f-1}^g+\boldsymbol{D}_f^d)_a\\ &=\sum_{j=0}^{g}\sum_{\boldsymbol{B}_g^j \in \boldsymbol{B}_{m-f-1}^g} \boldsymbol{G}(\boldsymbol{B}_{m-f-1}^g+\boldsymbol{D}_f^d, \boldsymbol{B}_g^j+\boldsymbol{D}_f^d, a)\\ &\quad\cdot\sum_{h=1}^{j+d}\sum_{\boldsymbol{B}_{j+d}^h \in \boldsymbol{B}_g^j+\boldsymbol{D}_f^d} \boldsymbol{G}(\boldsymbol{B}_g^j+\boldsymbol{D}_f^d, \boldsymbol{B}_{j+d}^h, a)\, \boldsymbol{V}(\boldsymbol{B}_{j+d}^h)\end{aligned} \tag{26}$$

where $j+d \geqq 1$.

5.5 $S(i, k, \boldsymbol{D}_m, \boldsymbol{D}_{m-1}^d)_a$

Suppose the number of waiting messages in the buffer is $k\,(N_{(m-1),i} \leqq k < N_{m,i})$, including the tagged message of class i at the head of the queue. Under a message-class dependent threshold-type scheduling, the busy servers are the set of servers $1, \cdots, m$, denoted as $\boldsymbol{D}_m$. Let $\boldsymbol{D}^d_{m-1}$ is the subset with d servers serving the messages of class i included in $\boldsymbol{D}_m - \{a\}$ where $\{a\}$ is included in $\boldsymbol{D}_m$. Sisce the number of messages waiting in the queue exceeds $N_{(m-1),i}$, the tagged message of class i is dispatched from the head of the queue to any one of servers $1, \cdots, m$ depending on which server becomes idle first. Let the server a accepts the tagged message. Since the tagged messsage is accepted by server a just after the server a completes serving the message, this message does not depend on the resequencing delay, even though it belongs to class i. The mean resequencing delay $R(i, k, D_m, D^d_{m-1})_a$ is expressed as

$$\boldsymbol{R}(i, k, \boldsymbol{D}_m, \boldsymbol{D}^d_{m-1})_a = \boldsymbol{G}(\boldsymbol{D}^d_{m-1}, \boldsymbol{D}^d_{m-1}, a)$$

$$\sum_{h=1}^{d} \sum_{D^h_d \in D^d_{m-1}} \boldsymbol{G}(\boldsymbol{D}^d_{m-1}, \boldsymbol{D}^h_d, a) \, \boldsymbol{V}(\boldsymbol{D}^h_d). \qquad (27)$$

5.6 $S(i, N_{m,i}, \boldsymbol{D}_m, \boldsymbol{D}^d_m)$

When a new message arrives in the system, the number of messages waiting in the buffer exceeds $N_{m,i}$ and immediately the tagged message of class i is dispatched from the head of the queue to server $m+1$. The mean resequencing delay $\boldsymbol{R}(i, N_{m,i}, \boldsymbol{D}_m, \boldsymbol{D}^d_m)$ is

$$\boldsymbol{R}(i, N_{mi}, \boldsymbol{D}_m, \boldsymbol{D}^d_m)$$

$$= \sum_{h=1}^{d} \sum_{D^h_d \in D^d_m} \boldsymbol{G}(\boldsymbol{D}^d_m, \boldsymbol{D}^h_d, m+1) \, \boldsymbol{V}(\boldsymbol{D}^h_d). \qquad (28)$$

By using the results in 5.1-5.6, and removing the condition on m busy servers, the mean resequencing delay at the transmission of the tagged message of class i, RT_i, can be expressed as follows:

$$RT_i$$

$$= \sum_{m=1}^{n} \Bigg[\sum_{B_m \in D_m - D_1} \sum_{g=1}^{m} \sum_{B^g_m \in B_m} p(i, 0, \boldsymbol{B}_m, \boldsymbol{B}^g_m)$$

$$\boldsymbol{R}(i, 0, \boldsymbol{B}_m, \boldsymbol{B}^g_m)$$

$$+ \sum_{k=0}^{N_{1,i}-1} \sum_{B_{m-1} \in D_n - D_1} \sum_{g=1}^{m-1} \sum_{B^g_{m-1} \in B_{m-1}} p(i, k, \boldsymbol{B}_{m-1}$$

$$+ \boldsymbol{D}_1, \boldsymbol{B}^g_{m-1}) \, \boldsymbol{R}(i, k, \boldsymbol{B}_{m-1} + \boldsymbol{D}_1, \boldsymbol{B}^g_{m-1})$$

$$+ \sum_{f=1}^{m-1} \sum_{B_{m-f} \in D_n - D_{f+1}} \sum_{g=1}^{m} \sum_{B^g_m \in B_{m-f} + D_f} p(i, N_{f,i},$$

$$\boldsymbol{B}_{m-f} + \boldsymbol{D}_f, \boldsymbol{B}^g_m) \, \boldsymbol{R}(i, N_{f,i}, \boldsymbol{B}_{m-f} + \boldsymbol{D}_f, \boldsymbol{B}^g_m)$$

$$+ \sum_{f=1}^{m-2} \sum_{k=N_{f,i}}^{N_{(f+1),i}-1} \sum_{B_{m-f-1} \in D_n - D_{f+1}} \sum_{a=1}^{f+1}$$

$$\sum_{d=0}^{f} \sum_{g=0}^{m-f-1} \sum_{D^d_f \in D_{f+1} - \{a\}} \sum_{B^g_{m-f-1} \in B_{m-f-1}}$$

$$p(i, k, \boldsymbol{B}_{m-f-1} + \boldsymbol{D}_{f+1}, \boldsymbol{B}^g_{m-f-1} + \boldsymbol{D}^d_f)_a$$

$$\boldsymbol{R}(i, k, \boldsymbol{B}_{m-f-1} + \boldsymbol{D}_{f+1}, \boldsymbol{B}^g_{m-f-1} + \boldsymbol{D}^d_f)_a$$

$$+ \sum_{k=N_{(m-1),i}}^{N_{m,a}-1} \sum_{a=1}^{m} \sum_{d=1}^{m-1} \sum_{D^d_{m-1} \in D_m - \{a\}}$$

$$\cdot p(i, k, \boldsymbol{D}_m, \boldsymbol{D}^d_{m-1})_a \boldsymbol{R}(i, k, \boldsymbol{D}_m, \boldsymbol{D}^d_{m-1})_a$$

$$+ \sum_{d=1}^{m} \sum_{D^d_m \in D_m} p(i, N_{m,i}, \boldsymbol{D}_m, \boldsymbol{D}^d_m)$$

$$\cdot \boldsymbol{R}(i, N_{m,i}, \boldsymbol{D}_m, \boldsymbol{D}^d_m) \Bigg]. \qquad (29)$$

6. Mean Resequencing Delay due to the Simultaneous Transmission of the Consecutive Message

In this section, we consider the resequencing delay due to the simultaneous message transmission of consecutive message(s) with the tagged message. Suppose the state for an $M/M/3$ queueing system is $S(i, k, j, 0, 0)$, where $j = 1, \cdots, C$, and $N_{2,m} - 1 \leqq k \leqq N_{1,i}$, for example. When the server 1 completes its service, then immediately the tagged message of class i is sent to server 1. If the successive two messages just after the tagged message belong to class m, then these two consecutive messages are sent simultaneously to server 2 and server 3, respectively, since the numbers of messages in the buffer exceed the corresponding message-class dependent thresholds $N_{1,m}$ and $N_{2,m}$, respectively. If the message of class m on server 3 bypasses the message of class m on server 2, it will experience resequencing delay.

Now, we derive the general expressions for the resequencing delay due to the simultaneous transmission of the consecutive messages for the $M/M/n$ queueing system shared by C different message classes. First, we focus on the case where state probability is $p(j, k, \boldsymbol{B}_{m-1} + \boldsymbol{D}_1, \boldsymbol{B}^g_{m-1})$, and consider the resequencing delay due to the simultaneous transmission of the consecutive message of class i. Suppose the number of waiting messages in the buffer is k $(1 \leqq k < N_{1,j})$ including the tagged message of class j at the head of the queue, and m servers including server 1 are busy. Here, $\boldsymbol{B}_{m-1}$ and $\boldsymbol{B}^g_{m-1}$ are the set of $m-1$ busy servers excluding server 1, and the subset with g servers serving the messages of class j included in $\boldsymbol{B}_{m-1}$, respectively. The tagged message of class j is dispatched to server 1 as soon as the previous message on server 1 finishes its service, Here, $\boldsymbol{B}^h_{m-1}$ is assumed to be the set of h busy servers when the tagged message is dispatched to server 1. Suppose the message waiting at the second position from the head of queue belongs to class $i (\neq j)$ and the set of idle servers which can accept the message of class i is expressed as $\boldsymbol{A}(2, i, k, \boldsymbol{B}^h_{m-1} + \boldsymbol{D}_1)$, where $\boldsymbol{A}(w, y, q, \boldsymbol{B})$ denotes the set of idle servers which can accept the message of class y waiting at the w-th position from the head of queue, if it is sent as the consecutive message due to the simultaneous transmission with the tagged message, where q and $\boldsymbol{B}$ are the number of messages in the buffer and the set of busy servers, respectively, when the message of class y is about to be dispatched to the server. If $\boldsymbol{A}(2, i, k, \boldsymbol{B}^h_{m-1} + \boldsymbol{D}_1)$ is not empty, the message of class i is sent due to the simultaneous fransmission to fastest server included in $\boldsymbol{A}(2, i, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1)$, denoted as $\min[\boldsymbol{A}(2, i, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1)]$. Similarly, the consecutive message of class i waiting at the e-th position from the head of queue $(2<e\leqq n-h)$ can be dispatched due to the simultaneous transmission on condition that

$$\prod_{w=2}^{e-1}\{\beta_y Q[\boldsymbol{A}(w, y, k+2-w, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1+\boldsymbol{J}(w-1, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1))]\}\beta_i Q[\boldsymbol{A}(e, i, k+2-e, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1+\boldsymbol{J}(e-1, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1))]. \quad (30)$$

When at least one server is included in $\boldsymbol{A}(w, y, q, \boldsymbol{B})$, $Q[\boldsymbol{A}(w, y, q, \boldsymbol{B})]=1$. Otherwise, $Q[\boldsymbol{A}(w, y, q, \boldsymbol{B})]=0$. The consecutive message due to the simultaneous transmission is accepted by the fastest server in $\boldsymbol{A}(w, y, q, \boldsymbol{B})$ denoted as $\min[\boldsymbol{A}(w, y, q, \boldsymbol{B})]$. $\boldsymbol{J}(w, q, \boldsymbol{B})$ is the set of servers which serve the consecutive messages due to the simultaneous transmission with the tagged message, expressed as

$$\boldsymbol{J}(w, q, \boldsymbol{B})=\boldsymbol{J}(w-1, q, \boldsymbol{B})+\min[\boldsymbol{A}(w, i, q+2-w, \boldsymbol{B}+\boldsymbol{J}(w-1, q, \boldsymbol{B}))] \quad w=2, \cdots, e-1 \quad (31)$$

and $\boldsymbol{J}(1, q, \boldsymbol{B})=\phi$. The simultaneously transmitted message of class i (e-th position in the queue) will experience the resequencing delay only if it completes its service before any of the messages of class i being served by the set of z servers, $\boldsymbol{E}^z(i)$, included in $\boldsymbol{B}_{m-1}+\boldsymbol{D}_1+\boldsymbol{J}(e-1, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1)$. The corresponding resequencing delay is expressed as

$$\sum_{t=1}^{z}\sum_{\boldsymbol{E}^t_z\in\boldsymbol{E}^z(i)} G(\boldsymbol{E}^z(i), \boldsymbol{E}^t_z, \min[\boldsymbol{A}(e, i, k+2-e, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1+\boldsymbol{J}(e-1, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1))])\, V(\boldsymbol{E}^t_z). \quad (32)$$

Thus, the mean resequencing delay due to the consecutive message of class i when the state probability is $p(j, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1, \boldsymbol{B}^g_{m-1})$, is obtained as

$$\sum_{h=0}^{m-1}\sum_{\boldsymbol{B}^h_{m-1}\in\boldsymbol{B}_{m-1}} G(\boldsymbol{B}_{m-1}, \boldsymbol{B}^h_{m-1}, 1)\, \boldsymbol{T}(n-h, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1, 1) \quad (33)$$

where

$$\boldsymbol{T}(f, k, \boldsymbol{B}, a)=\sum_{e=2}^{f}\sum_{m_2=1(\neq j)}^{C}\cdots\sum_{m_{e-1}=1(\neq j)}^{C}\prod_{w=2}^{e-1}\{\beta_m Q[\boldsymbol{A}(w, m, k+2-w, \boldsymbol{B}+\boldsymbol{J}(w-1, k, \boldsymbol{B}))]\}\beta_i Q[\boldsymbol{A}(e, i, k+2-e, \boldsymbol{B}+\boldsymbol{J}(e-1, k, \boldsymbol{B}))]\sum_{t=1}^{z}\sum_{\boldsymbol{E}^t_z\in\boldsymbol{E}^z(i)} G(\boldsymbol{E}^z(i), \boldsymbol{E}^t_z, \min[\boldsymbol{A}(e, i, k+2-e, \boldsymbol{B}+\boldsymbol{J}(e-1, k, \boldsymbol{B}))])\, V(\boldsymbol{E}^z_t). \quad (34)$$

and a denotes the server which serves the tagged message of class j. By considering all cases of state probabilities and removing the condition of m, the resequencing delay due to the simultaneous transmission of the consecutive message of class i, RC_i, under the message-class dependent threshold-type scheduling is obtained as follows:

$$\begin{aligned}RC_i=&\sum_{j=1(\neq i)}^{C}\sum_{m=1}^{n}\Bigg[\sum_{k=0}^{N_{1,j}-1}\sum_{\boldsymbol{B}_{m-1}\in\boldsymbol{D}_n-\boldsymbol{D}_1}\sum_{g=0}^{m-1}\sum_{\boldsymbol{B}^g_{m-1}\in\boldsymbol{B}_{m-1}} p(j, k, \boldsymbol{B}_{m-1}+\boldsymbol{D}_1, \boldsymbol{B}^g_{m-1})\\&\sum_{h=0}^{m-1}\sum_{\boldsymbol{B}^h_{m-1}\in\boldsymbol{B}_{m-1}} G(\boldsymbol{B}_{m-1}, \boldsymbol{B}^h_{m-1}, 1)\, \boldsymbol{T}(n-h, k, \boldsymbol{B}^h_{m-1}+\boldsymbol{D}_1, 1)\\&+\sum_{f=1}^{m-1}\sum_{\boldsymbol{B}_{m-f}\in\boldsymbol{D}_n-\boldsymbol{D}_{f+1}}\sum_{g=0}^{m}\sum_{\boldsymbol{B}^g_m\in\boldsymbol{B}_{m-f}+\boldsymbol{D}_f} p(j, N_{f,j}, \boldsymbol{B}_{m-f}+\boldsymbol{D}_f, \boldsymbol{B}^g_m)\, \boldsymbol{T}(n-m, N_{f,j}, \boldsymbol{B}_{m-f}+\boldsymbol{D}_{f+1}, f+1)\\&+\sum_{f=1}^{m-2}\sum_{k=N_{f,j}}^{N_{(f+1),j}-1}\sum_{\boldsymbol{B}_{m-f-1}\in\boldsymbol{D}_n-\boldsymbol{D}_{f+1}}\sum_{a=1}^{f+1}\sum_{d=0}^{f}\sum_{g=0}^{m-f-1}\sum_{\boldsymbol{D}^d_f\in\boldsymbol{D}_{f+1}-\{a\}}\sum_{\boldsymbol{B}^g_{m-f-1}\in\boldsymbol{B}_{m-f-1}} p(j, k, \boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1}, \boldsymbol{B}^g_{m-f-1}+\boldsymbol{D}^d_f)_a\\&\sum_{h=0}^{m-f-1}\sum_{\boldsymbol{B}^h_{m-f-1}\in\boldsymbol{B}_{m-f-1}} G(\boldsymbol{B}_{m-f-1}, \boldsymbol{B}^h_{m-f-1}, a)\, \boldsymbol{T}(n-h-f, k, \boldsymbol{B}^h_{m-f-1}+\boldsymbol{D}_{f+1}, a)\\&+\sum_{k=N_{(m-1),j}}^{N_{m,j}-1}\sum_{a=1}^{m}\sum_{d=0}^{m-1}\sum_{\boldsymbol{D}^d_{m-1}\in\boldsymbol{D}_m-\{a\}} p(j, k, \boldsymbol{D}^d_{m-1})_a\, \boldsymbol{T}(n-m-1, k, \boldsymbol{D}_m, a)\\&+\sum_{d=0}^{m}\sum_{\boldsymbol{D}^d_m\in\boldsymbol{D}_m} p(j, N_{m,i}, \boldsymbol{D}_m, \boldsymbol{D}^d_m)\, \boldsymbol{T}(n-m, N_{m,i}, \boldsymbol{D}_{m+1}, m+1)\Bigg]. \quad (35)\end{aligned}$$

7. The Mean Resequencing Delay and Total Delay

The mean resequencing delay is obtained as

$$RD_{mean}=\sum_{i=1}^{C}(RT_i+RC_i)=\sum_{i=1}^{C}(p_T(i)+p_C(i))RD(i) \quad (36)$$

where $RD(i)$ is the mean resequencing delay of the message of class i, $p_T(i)$ is the probability that the message of class i becomes the tagged message, and $p_C(i)$ is the probability that the message of class i becomes the consecutive message just after the tagged message which experience the simultaneous transmission, respectively.

$$RD(i)=\frac{(RT_i+RC_i)}{(p_T(i)+p_C(i))} \tag{37}$$

$$\begin{aligned}
p_T(i)=\sum_{m=1}^{n-1}\Bigg[&\sum_{B_m\in D_n}\sum_{g=0}^{m}\sum_{B_m^g\in B_m} p(i,0,\boldsymbol{B}_m,\boldsymbol{B}_m^g)\\
&+\sum_{k=1}^{N_{1,i}}\sum_{B_{m-1}\in D_n-D_1}\sum_{g=0}^{m-1}\sum_{B_{m-1}^g\in B_{m-1}} p(i,k,\\
&\boldsymbol{B}_{m-1}+\boldsymbol{D}_1,\boldsymbol{B}_{m-1}^g)\\
&+\sum_{f=1}^{m-2}\sum_{k=N_{f,i}+1}^{N_{(f+1),i}}\sum_{B_{m-f-1}\in D_n-D_{f+1}}\\
&\sum_{d=0}^{f+1}\sum_{g=0}^{m-f-1}\sum_{D_{f+1}^d\in D_{f+1}}\sum_{B_{m-f-1}^g\in B_{m-f-1}}\\
&p(i,k,\boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1},\boldsymbol{B}_{m-f-1}^g\\
&+\boldsymbol{D}_{f+1}^d)\\
&+\sum_{k=N_{(m-1),i}+1}^{N_{m,i}}\sum_{d=0}^{m}\sum_{D_m^d\in D_m} p(i,k,\boldsymbol{D}_m,\boldsymbol{D}_m^d)\Bigg]\\
&+\sum_{k=0}^{\infty}\sum_{d=0}^{n}\sum_{D_n^d\in D_n} p(i,k,\boldsymbol{D}_n,\boldsymbol{D}_n^d).
\end{aligned} \tag{38}$$

$$\begin{aligned}
p_C(i)=\sum_{j=1(\neq i)}^{C}\sum_{m=1}^{n}\Bigg[&\sum_{k=0}^{N_{1,i}-1}\sum_{B_{m-1}\in D_n-D_1}\sum_{g=0}^{m-1}\\
&\sum_{B_{m-1}^g\in B_{m-1}} p(i,k,\boldsymbol{B}_{m-1}+\boldsymbol{D}_1,\boldsymbol{B}_{m-1}^g)\\
&\sum_{h=0}^{m-1}\sum_{B_{m-1}^h\in B_{m-1}} \boldsymbol{G}(\boldsymbol{B}_{m-1},\boldsymbol{B}_{m-1}^h,1)\,\boldsymbol{H}(n-h,\\
&k,\boldsymbol{B}_{m-1}^h+\boldsymbol{D}_1)\\
&+\sum_{f=1}^{m-1}\sum_{B_{m-f}\in D_n-D_{f+1}}\sum_{g=0}^{m}\sum_{B_m^g\in B_{m-f}+D_f}\\
&p(i,N_{f,i},\boldsymbol{B}_{m-f}+\boldsymbol{D}_f,\boldsymbol{B}_m^g)\\
&\boldsymbol{H}(n-m,N_{f,i},\boldsymbol{B}_{m-f}+\boldsymbol{D}_{f+1})\\
&+\sum_{f=1}^{m-2}\sum_{k=N_{f,i}}^{N_{(f+1),i}-1}\sum_{B_{m-f-1}\in D_n-D_{f+1}}\\
&\sum_{a=1}^{f+1}\sum_{d=0}^{f}\sum_{g=0}^{m-f-1}\sum_{D_f^d\in D_{f+1}-\{a\}}\sum_{B_{m-f-1}^g\in B_{m-f-1}}\\
&p(i,k,\boldsymbol{B}_{m-f-1}+\boldsymbol{D}_{f+1},\boldsymbol{B}_{m-f-1}^g+\boldsymbol{D}_f^d)_a\\
&\sum_{h=0}^{m-f-1}\sum_{B_{m-f-1}^h\in B_{m-f-1}} \boldsymbol{G}(\boldsymbol{B}_{m-f-1},\boldsymbol{B}_{m-f-1}^h,a)\\
&\boldsymbol{H}(n-h-f,k,\boldsymbol{B}_{m-f-1}^h+\boldsymbol{D}_{f+1})\\
&+\sum_{k=N_{(m-1),i}}\sum_{a=1}^{m}\sum_{d=0}^{m-1}\sum_{D_{m-1}^d\in D_m-\{a\}}\\
&p(i,k,\boldsymbol{D}_m,\boldsymbol{D}_{m-1}^d)_a\boldsymbol{H}(n-m-1,k,\boldsymbol{D}_m)\\
&+\sum_{d=0}^{m}\sum_{D_m^d\in D_m} p(i,N_{m,i},\boldsymbol{D}_m,\boldsymbol{D}_m^d)\\
&\boldsymbol{H}(n-m,N_{m,i},\boldsymbol{D}_{m+1})\Bigg]
\end{aligned} \tag{39}$$

where

$$\begin{aligned}
\boldsymbol{H}(n,k,\boldsymbol{B})=\sum_{e=2}^{n}\prod_{w=2}^{e-1}\sum_{m=1(\neq j)}^{C}&\beta_m\\
&Q[\boldsymbol{A}(w,m,k+2-w,\boldsymbol{B}\\
&+\boldsymbol{J}(w-1,k,\boldsymbol{B}))]\\
&\beta_i Q[\boldsymbol{A}(e,i,k+2-e,\boldsymbol{B}\\
&+\boldsymbol{J}(e-1,k,\boldsymbol{B}))].
\end{aligned} \tag{40}$$

The total delay of the message of class i and the mean total delay is represented, respectively as

$$TD(i)=QD_{mean}+RD(i) \tag{41}$$

$$TD_{mean}=QD_{mean}+RD_{mean}. \tag{42}$$

When $N_{f,1}=N_{f,2}=\cdots=N_{f,C}$ $(f=1,\cdots,m)$, the expressions for the mean queueing delay and resequencing delay are identical with those in Ref.(10) obtained under a message-class independent threshold-type scheduling. Note that the queueing delay as well as the resequencing delay depend on the distribution of arrival message classes under a message-class dependent threshold-type scheduling, since the threshold values are changed according to the class of the tagged message. On the other hand, in the conventional threshold-type scheduling, the queueing delay does not depends on the distribution of the message classes.

8. Numerical Calculation of the Delay for the $M/M/2$ Queue

In this section, we consider the effect of the message-class dependent threshold-type scheduling on the delay of the $M/M/2$ queueing system with two classes of arriving messages by numerical calculation, and then compare the results with those for the conventional queueing system in which the threshold values are independent of the message class of the tagged message. Suppose the messages of class 1 need the resequencing, and thus, require the minimization of mean total delay $TD(1)$. On the other hand, the messages of class 2, which do not need the resequencing, and require the minimization of mean queueing delay QD_{mean}. Here, the weighted sum $\beta_1 TD(1)+\beta_2 QD_{mean}$ is used as the evaluation function. The eptimum values of the message-class dependent thresholds and the conventional threshold are summarized in Table 1, where $\lambda_1:\lambda_2=6:4$, $\mu_1:\mu_2=9:1$ and $\mu_1+\mu_2=1$. It is shown from Table 1 that under the message-class dependent threshold-type scheduling, the optimum values of the thresholds $N_{1,1}$ and $N_{1,2}$ are different. The value of $N_{1,1}$ is larger than $N_{1,2}$, and as the traffic intensity is increased, $N_{1,2}$ tends to be small while $N_{1,1}$ is kept large compared to $N_{1,2}$ The reason is the following. When the traffic intensity is very small, server 1 can manage to serve almost all messages with reasonably small queueing delay, and the resequencing delay becomes zero if only server 1 is used. Thus, if the traffic intensity is very small, the evaluation function can be minimized by increasing the threshold value appropriately, which corresponds to increase the priority of server 1 regardless of the class of messages. When the traffic intensity is relatively large, some of the messages have to be served by server 2 in order to avoid excessive queueing delay. Since the messages of class 2 do not require resequencing and the portion of messages causing resequencing delay is more increased as more messages of class 1 are sent to server 2, the message-class thresholds are adjusted in such a way where server 2 receives more messages of class 2 than those of class 1. This means that the message-class dependent threshold scheduling mainly works to control the routing probability. The routing probability of sending the messages of class 2 to server 2 is controlled to be larger than that of sending the messages of class 1 to server 2, and hence, the priority of the message of class 2 is emphasized for server 2. This corresponds to the decrease of the value of threshold $N_{1,2}$ and the increase of the corresponding value of $N_{1,1}$ appropriately to minimize the evaluation function. As the traffic intensity is larger, the value of the threshold $N_{1,2}$ should become smaller in order to maintain small queueing delay, while the value of threshold $N_{1,1}$ is kept relatively large compared to the corresponding $N_{1,2}$ to achieve the small resequencing delay of the message of class 1. The numerical results and the computer simulated results for mean queueing delay, resequencing delay and total delay of the systems under a message-class dependent threshold-type scheduling are shown in Figs. 2-5, where the optimum thresholds in Table 1 are used, where the parameters are $\lambda_1 : \lambda_2 = 6:4$, $\mu_1 : \mu_2 = 9:1$, $\mu_1 + \mu_2 = 1$. For the comparison, the results of the system without threshold-type scheduling (Zero-threshold, $N_{1,1} = N_{1,2} = 0$) and those of the system under the conventional threshold-type scheduling ($N_{1,1} = N_{1,2} = N_1$) are also shown in Figs. 2-5. Figure 2 shows the mean queueing delay QD_{mean} versus traffic intensity ρ. It is found that the threshold-type scheduling is more effective to reduce the mean queueing delay in the region modest traffic intensity when the service times of the servers are different. When the traffic intensity is small, the number of waiting message is relatively small and the faster server can manage to serve most of messages. Thus, the queueing delay due to large service time of the slower server becomes very small, although the queueing delay due to the waiting time in the buffer is somewhat increased. It is also shown in Fig. 2 that the mean queueing delay under the message-class dependent threshold-type scheduling is more or less the same as that attained under the conventional threshold-type scheduling. Note that $N_{1,2}$

Table 1 Optimum thresholds for the queueing system under a message-class dependent threshold-type scheduling or a conventional threshold-type scheduling. $\lambda_1 : \lambda_2 = 6:4$, $\mu_1 : \mu_2 = 9:1$, $\mu_1 + \mu_2 = 1$

ρ	conventional threshold-type	class dependent threshold-type	
	N_1	$N_{1,1}$	$N_{1,2}$
0.1	6	6	6
0.2	9	9	6
0.3	11	10	5
0.4	10	13	4
0.5	10	16	3
0.6	8	22	3
0.7	7	31	2
0.8	6	34	2
0.9	4	17	1

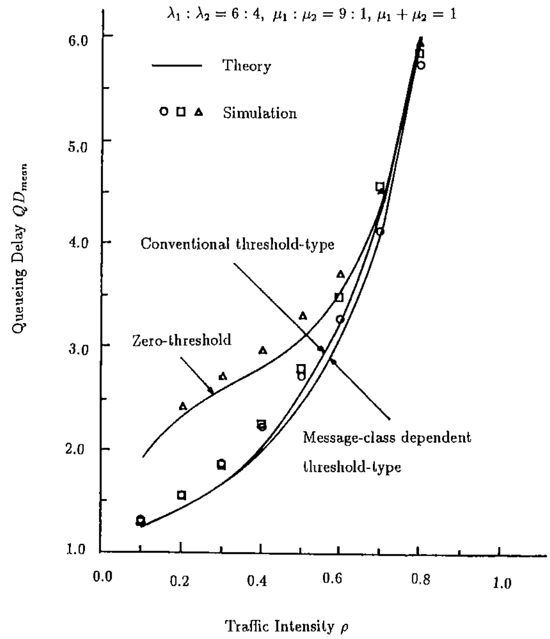


Fig. 2 Mean queueing delay QD_{mean} versus traffic intensity ρ. $\lambda_1 : \lambda_2 = 6:4$, $\mu_1 : \mu_2 = 9:1$ and $\mu_1 + \mu_2 = 1$.

and $N_{1,2}$ are not always the same and mean value of them may be a real number, and thus, the more flexible routing control might be achieved compared to the conventional threshold-type scheduling where the threshold value has to be an integer. It is also shown that the computer simulated results agree well with the numerical results. Figure 3 shows the mean resequencing delay $RD(1)$ of the message of class 1 versus traffic intensity ρ. It is found that the message-class dependent threshold-type scheduling is effective to reduce the mean resequencing delay of the message of class 1 even when the traffic intensity is relatively high. It is also shown that the computer simulated results agree well with the numerical results. Since the messages of class 2 do not need resequencing, and thus, minimizing

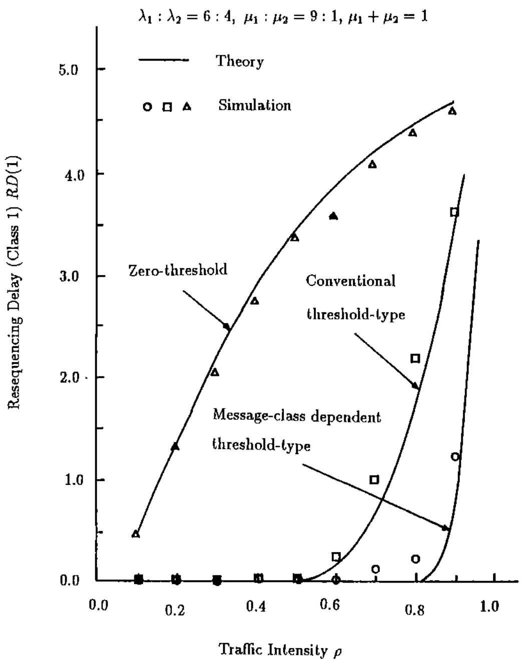


Fig. 3 Mean resequencing delay $RD(1)$ of the message of class 1 versus traffic intensity ρ. $\lambda_1 : \lambda_2 = 6 : 4$, $\mu_1 : \mu_2 = 9 : 1$ and $\mu_1 + \mu_2 = 1$.

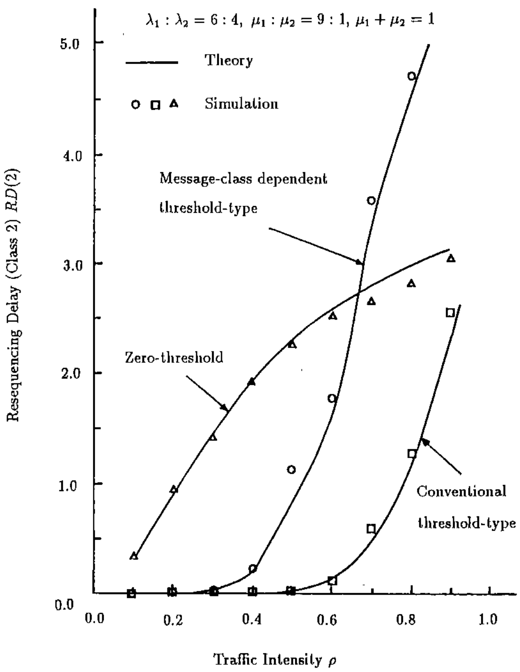


Fig. 4 Mean resequencing delay $RD(2)$ of the message of class 2 versus traffic intensity ρ. $\lambda_1 : \lambda_2 = 6 : 4$, $\mu_1 : \mu_2 = 9 : 1$ and $\mu_1 + \mu_2 = 1$.

mean queueing delay is requimed. However, the resequencing delay of the message of class 2 is also calculated in order to consider the behavior of the message-class dependent threshold-type scheduling. The results

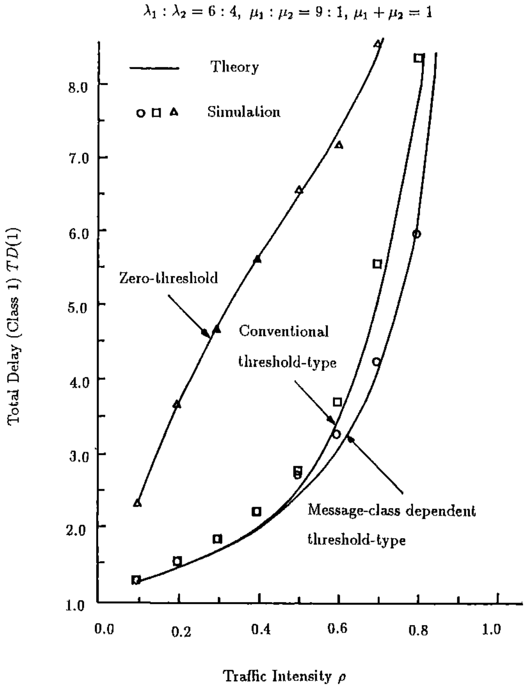


Fig. 5 Mean total delay $TD(1)$ of the message of class 1 versus traffic intensity ρ. $\lambda_1 : \lambda_2 = 6 : 4$, $\mu_1 : \mu_2 = 9 : 1$ and $\mu_1 + \mu_2 = 1$.

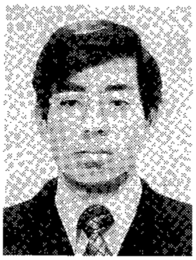
are shown in Fig. 4. It is shown that the resequencing delay of the message of class 2 will be large if the resequencing is done, though it is assumed that the messages of class 2 do not need resequencing. Figure 5 shows the mean total delay $TD(1)$ of the message of class 1 versus traffic intensity ρ. It is shown that the message-class dependent threshold-type scheduling is effective to reduce the total delay of the messages of class 1 even when the traffic intensity is relatively high. It is also shown that the computer simulated results agree well with the numerical results. The message-class dependent threshold-type scheduling is shown to be effective to reduce the resequencing delay and the total delay of the message of class 1, while the mean queueing delay is kept as small as that attained under the conventional threshold-type scheduling. Thus, the proposed scheduling has more flexibility to satisfy the different requirements of the different message classes, such as the minimization of only queueing delay or total delay including resequencing delay, by selecting the values of the message-class dependent thresholds appropriately.

9. Conclusions

The effect of message-class dependent threshold-type scheduling on the queueing delay and resequencing delay for the $M/M/n$ queueing system has been analzed. We first derive the expressions for the state equations, mean queueing delay, resequencing delay and total delay for the $M/M/n$ queueing system shared by C different message classes under a threshold-type scheduling in which the threshold values depend on the message class at the head of queue and the number of messages in the buffer. Next, the numerical calculation is carried out for the queueing system with two servers under a message-class dependent threshold-type scheduling. It is found that the message-class dependent threshold-type scheduling is effective to reduce the resequencing delay of some message class, which can not be attained under the conventional threshold-type scheduling, and thus, the proposed sheduling can satisfy the different requirements of the different message classes, such as the minimization of only queueing delay or total delay including resequencing delay.

References

(1) Gray J. P. and McNeil T. B.: "SNA multiple system networking", IBM Syst. J., **18**, pp. 263-279 (1979).
(2) Lin W. and Kumar P. R.: "Optimal Control of a queueing system with two heterogeneous servers", IEEE Trans. Automat. Contr., **AC-29**, 8, pp. 696-703 (August 1984).
(3) Nelson R. and Towsley D.: "Approximating the mean time in system in a multiple-server queue that uses threshold scheduling", Operation Research, **35**, 3, pp. 419-427 (May 1987).
(4) Sasase I. and Mori S.: "Analysis of queueing systems with multiple servers under a threshold-type scheduling", Trans. IEICE, **E 73**, 4, pp. 545-553 (April 1990).
(5) Agrawal S. and Ramaswamy R.: "Analysis of the resequencing delay for $M/M/m$ systems", Proc. of the 1987 ACM SIGMETRIBS Conf. on Measurement and Modeling of Computer Systems, pp. 27-35 (May 1987).
(6) Yum T.-S. P. and Ngai T.-Y.: "Resequencing of messages in communication networks", IEEE Trans. Commun., **COM-34**, 2, pp. 143-149 (February 1986).
(7) Iliadis I. and Lien L. Y.-C.: "Resequencing in distributed systems with multiple classes", Proc. of INFORCOM '88, **3**, pp. 881-888 (April 1988).
(8) Iliadis I. and Lien L. Y.-C.: "Resequencing delay for a queueing system with two heterogeneous servers under a threshold-type scheduling", IEEE Trans. Commun., **COM-36**, 6, pp. 692-702 (June 1988).
(9) Sasase I. and Mori S.: "Resequencing delay for a queueing system with multiple servers under a threshold-type scheduling", Trans. IEICE, **E 73**, 6, pp. 928-935 (June 1990).
(10) Sasase I. and Mori S.: "The effect of multiple arrival classes on resequencing delay for a queueing system with multiple servers under a threshold-type scheduling", IEICE Trans., **E 74**, 4, pp. 706-714 (April 1991).



Iwao Sasase was born in Osaka, Japan on November 17, 1956. He received the B.E., M.E., and Ph. D. degrees in Electrical Engineering from Keio University, Yokohama, Japan in 1979, 1981, and 1984, respectively. From 1984 to 1986 he was a Post Docoral Fellow and a Lecturer of Electrical Engineering at University of Ottawa, Canada. He is now an Associate Professor of Electrical Engineering at Keio University, Japan His research interests include modulation and coding, satellite and microwave communications, optical communications, communication networks, and information theory. He received 1984 IEEE Communication Society Student Paper Award (Region 10), 1988 Hiroshi Ando Memorial Young Engineering Award, and 1988 Shinohara Memorial Young Engineering Award. Dr. Sasase is a member of the Institute of Electrical and Electronics Engineers (IEEE), and the Society of Informatio Theory and Its Applications.

Yoshifumi Nishio was born in Mie, Japan, on January 24, 1966. He received the B.E. and M.E. degrees in Electrical Engineering from Keio University, Yokorama, Japan, in 1988 and 1990, respectively. He is currently working towards the Ph. D. degree in the Department of Electrical Elgineering, Keio University. His research interests are in nonlinear circuits, network theory and cryptosystem. He is a student member of IEEE.

Hitomi Nakamura was born in Aomori, Japan, on October 21, 1968. She received the B.E. degree in Electrical Elgineering from Keio University, Yokohama, Japan in 1991. Her research interests include queueing theory and communication networks. She has been working in Flight Operation Engineering for All Nippon Airways since April 1991.