Analysis of a Batch-Arrival Batch-Service Queueing System and Its Application to Optimization of the Order of Services

Guidance

Professor Masao FUKUSHIMA
Associate Professor Tetsuya TAKINE

Hideki TAKEGUCHI

2001 Graduate Course
in
Department of Applied Mathematics and Physics
Graduate School of Informatics
Kyoto University

February, 2003

Abstract

In batch-arrival batch-service queueing systems, customers arrive in batches, and customers in each batch are served simultaneously. Up to now, such a queueing system provides practical models for performance evaluation in computer and communication systems. In most of the past studies, finite buffer or no buffer is assumed. In particular, there is no study that considers the order of services in the system with a buffer of infinite capacity. Note, however, that the order of services is very influential in the performance of batch-arrival batch-service systems. For example, under the first-come, first-served (FCFS) discipline, some customers may have to wait even when there are some idle servers, if the customer at the head of the queue needs more servers than idle ones. Thus the throughput of the FCFS system goes down. On the other hand, if customers of a particular class are given priority, the average waiting time of this priority class is short, whereas that of non-priority classes may be very long. Therefore, by such a simple control of the order of services, customers are usually served inefficiently or unfairly.

In this thesis, we deal with a batch-arrival batch-service queueing system with a buffer of infinite capacity, and we aim to achieve fair and efficient service by controlling the order of services. To make things tractable, we assume to observe at most α customers from the head of the queue and determine the customer to be served among those according some policy. By doing so, this system is formulated as a Markov decision process with a countable state space. We develop a algorithmic method to compute the average waiting time for a given policy. Further, through numerical experiments, we find the tendency of optimal policies. Next, based on this observation, we develop a sequential improvement algorithm to compute a quasi-optimal policy for large α. Numerical results indicate that the proposed algorithm achieves fair and efficient service under an appropriate criterion.

Contents

1 Introduction

In batch-arrival batch-service queueing systems, customers arrive in batches, and customers in each batch are served simultaneously. In this thesis, we deal with the batch-arrival batch-service queueing system with c ($c > 1$) servers and a buffer. Because there is a buffer, arriving batches may wait in the queue when the system is busy. We assume that the capacity of buffer is infinite.

Up to now, such a queueing system provides practical models for performance evaluation in computer and communication systems, e.g., multiprogramming computer systems, for which each program requires the loading of a random number of memory units from a main memory store; a circuit-switched telecommunication system which supports a variety of traffic type (e.g., voice, video, etc), each of which having different bandwidth requirements and holding-time distributions.

Such a batch-arrival batch-service queueing system with a buffer of infinite capacity has been less well investigated. In most of the past studies, finite buffer or no buffer is assumed. Kino [1] analyzed the system with infinite buffer with the use of generating functions when the order of services is on a first-come, first-served (FCFS) basis. But there is no study that considers the order of services in the system with a buffer of infinite capacity.

In this system, the order of services is very influential in the performance of the system. For example, under the FCFS discipline, some customers may have to wait even when there are some idle servers, if the customer at the head of the queue needs more servers than idle ones. Thus the throughput of the FCFS system goes down. On the other hand, if customers of a particular class are given priority, the average waiting time of this priority class is short, whereas that of non-priority classes may be very long. Therefore, by such a simple control of the order of services, customers are usually served inefficiently or unfairly.

If we want more efficient and fair service, we must take the circumstances of the system into account and decide which customer to be served. If we could memorize all the class of customers in the queue and calculate the optimal order of services, we could get high-performance. But it is difficult to analyze the system when we use the information of all the customers in the queue. To make things tractable, therefore, we assume to observe at most α customers from the head of the queue and determine the customer to be served among those according some policy. By doing so, this system is formulated as a Markov decision process with a countable state space. We develop an algorithmic method to compute the average waiting time for a given policy. Further, through numerical experiments, we find the tendency of optimal policies. Next, based on this observation, we develop a sequential improvement algorithm to compute a quasi-optimal policy for large α.

The rest of this thesis is divided into six sections. In section 2, we describe the model considered in this thesis. In section 3, we develop an algorithmic method to compute the average waiting time for a given policy. In section 4, we show some numerical results to reveal the tendency of optimal policies. In section 5, we develop a sequential improvement algorithm to compute a quasi-optimal policy for large α. Finally, the conclusion is given in section 6.

2 Model Description

Let c denote the number of servers. We can see a batch consisting of n customers as a customer who uses n servers. Hereafter we introduce the latter view. We assume that a customer who belongs to class i ($i = 1, \cdots, k$) uses c_i ($1 \leq c_i \leq c$) servers. For the sake of brevity, we suppose that $\min_i c_i = c_1 = 1$.

Throughout this thesis, we assume that class i customers arrive at the system according to a Poisson process of rate λ_i. Service times of class i customers are distributed according to an exponential distribution with rate μ_i. We define $\rho_i = \lambda_i/\mu_i$ and $\lambda = \sum_{i=1}^{c} \lambda_i$.

It is very difficult to analyze the system when we use the information of classes of all the customers in the queue. So we assume to observe at most α customers from the head of the queue and determine the customer to be served among those according to some policy. We explain this point in a little more detail in the subsequent chapter.

3 The average waiting time for a given policy

In this section, we develop an algorithmic method to compute the average waiting time for a given policy. At first, we define the policy which describes the order of services. Next, we formulate the system as a Markov decision process for a given policy. Finally, we develop an algorithmic method to compute the steady state probability vector and the average waiting time of each class.

3.1 Policy and state space

In this subsection, we define the policy which we deal with in this thesis and describe the state space of the system.

As a precondition, we do not permit the interruption of services. So once a customer begins to be served, it has served until it departs from the system.

To consider the control of the order of services, we need information about not only customers in service but also customers in the queue. If we could use information of all customers in the queue and find an optimal policy within a reasonable time, it is clear that the optimal policy brings about the best performance. However if we memorize the class of all the customers in the queue, the number of states explode and it becomes difficult to analyze the system. Therefore we assume to memorize classes of α ($\alpha > 1$) customers from the head of the queue. It is considered that the more α increases, the more optimality of the policy increases. However the more α increases, the more difficulty of the analysis also increases. So it is an interesting problem to look into the relationship between the degree of the performance improvement and the degree of the difficulty of the problem when α is varied.

Now we refer to the state of the system. The state of the system is represented by a vector $Y = (N, G_1, \cdots, G_k, Q_1, \cdots, Q_k)$, where N is the number of customers in the system, G_i is the number of class i customers in service, and Q_j is the number of class j customers who are in α customers from the head of the queue. We call the variable N "level" and $(G_1, \cdots, G_k, Q_1, \cdots, Q_k)$ "phase". We denote the set of the possible state by $\mathcal{Y}$. Then, elements

of the set $\mathcal{Y}$ satisfy the following constraints.

1. $N \geq 0$,
   $0 \leq G_i \leq \lfloor \frac{c}{c_i} \rfloor \qquad (i = 1, \cdots, k)$,
2. $\sum_{i=1}^{k} G_i + \sum_{j=1}^{k} Q_j \leq N$,
3. $\sum_{i=1}^{k} c_i G_i \leq c$,
4. $\sum_{j=1}^{k} Q_j \leq \alpha$.

The constraint 1 is trivial. The constraint 2 represents the condition on the number of customers in the system. The constraint 3 represents the relation between the number of servers and the number of customers in service. The constraint 4 represents that we observe the class of at most α customers in the queue.

In this thesis, we define that a policy is applied just after the state of the system is changed. Strictly speaking, there are three patterns of the moment when the policy is applied; a customer arrives, a customer departs from the system, and a customer begins to be served according to the decision of the policy. At these moments, one customer out of α customers from the head of the queue or no customer begins to be served according to the decision of the policy. Note that if a customer begins to be served at a moment, at least one decision is made at the same time.

In this thesis, we deal with policies which depend on the phase. Namely, which class customer is served depends only on the phase of the system at that moment, not on the past history. To adopt these policies, the process which we consider is formulated as a Markov decision process with a countable state space. Let $S = \{1, 2, \cdots, F\}$ denote the set of phases of the system. For each phase $i \in S$, the set of actions is given by $A(i)$, where a $(\neq 0) \in A(i)$ denotes the class of the customer which begins to be served and a $(= 0) \in A(i)$ denotes that no customer begins to be served. Note that we restrict the customer who can begin to be served to one customer out of α customers from the head of the queue, so the action $a \in A(i)$ is restricted to the classes which satisfy the condition $Q_a \neq 0$.

We make a policy to choose action $a \in A(i)$ for all phase i $(i = 1, 2, \cdots, F)$. The number of phases is finite, and the number of actions for each phase is also finite. So the number of policies is also finite.

3.2 Transition rate matrix

Now we consider the transition rate between the states. The transition rate matrix differs from policy to policy. So we divide a transition into an arrival or departure and decision of the customers who are served.

At first, we consider the transition when a customer arrives. The arrival process is a Poisson process, so that at most one customer arrives to the system at a time. Let the state before the arrival be $y = (n, g_1, \cdots, g_k, q_1, \cdots, q_k)$, and the state after the arrival be $y' = (n + 1, g'_1, \cdots, g'_k, q'_1, \cdots, q'_k)$. Then the transition rate $p(y, y')$ is given by

$$p(y,y') = \begin{cases} \lambda, & g_i = g_i' \ (i=1,\cdots,k), \\ & \sum_{j=1}^{k} q_j = \alpha \text{ and } q_j = q_j' \ (j=1,\cdots,k), \\ \lambda_a, & g_i = g_i' \ (i=1,\cdots,k), \\ & \text{there exits a class } a \text{ such that } q_a + 1 = q_a' \\ & \text{ and } q_j = q_j' \ (j \neq a, \ j=1,\cdots,k), \\ 0, & \text{otherwise.} \end{cases}$$

Let A_i $(i = 0, 1, 2, \cdots, c+\alpha)$ denote transition matrix which corresponds to the transitions from states with i customers to states with $i+1$ customers. Note that the block matrices corresponding to the transition from $c+\alpha+1+i$ $(i \geq 0)$ customers are given by the same matrix A. Since the combination of phases is identical when many customers exist, the transition rate matrices are identical as well.

Next we consider the transition when a customer departs. Because we suppose exponential service times, at most one customer departs from system at a time. Let the state before the departure be $y = (n, g_1, \cdots, g_k, q_1, \cdots, q_k)$, the state after the departure be $y = (n-1, g_1', \cdots, g_k', q_1', \cdots, q_k')$. Then the transition rate $p(y, y')$ is given by

$$p(y,y') = \begin{cases} g_d\mu_d, & \text{there exits a class } d \text{ such that } g_d - 1 = g_d', \\ & g_i = g_i' \ (i \neq d, \ i=1,\cdots,k) \text{ and } q_j = q_j' \ (j=1,\cdots,k), \\ 0, & \text{otherwise.} \end{cases}$$

Let B_i $(i = 1, 2, \cdots, c+\alpha)$ denote the transition matrix which corresponds to the transition from states with i customers to states with $i-1$ customers. Note that the block matrices corresponding to the transition from $c+\alpha+1+i$ $(i \geq 0)$ customers are given by the same matrix B, because the combination of phases is identical.

Finally, we consider the transition when the policy is applied. We denote the state before the application of the policy by $y = (n, g_1, \cdots, g_k, q_1, \cdots, q_k)$, and the state after the application of the policy by $y' = (n, g_1', \cdots, g_k', q_1', \cdots, q_k')$. If the decision of the policy is to serve no customer, then $p(y, y')$ is 1 if $y = y'$ and 0 otherwise. Now we consider the case that the decision of the policy for the phase $(g_1, \cdots, g_k, q_1, \cdots, q_k)$ is to serve a customer of class s. Then the transition rate $p(y, y')$ is given by

$$p(y,y') = \begin{cases} 1, & \sum_{i=1}^{k} g_i + q_i = n, \ g_s + 1 = g_s', \ q_s - 1 = q_s', \\ & g_i = g_i' \ (i \neq s, \ i=1,\cdots,k) \text{ and } q_j = q_j' \ (j \neq s, \ j=1,\cdots,k), \\ \frac{\lambda_s}{\lambda}, & \sum_{i=1}^{k} g_i + q_i < n, \ g_s + 1 = g_s', \\ & g_i = g_i' \ (i \neq s, \ i=1,\cdots,k) \text{ and } q_j = q_j' \ (j=1,\cdots,k), \\ \frac{\lambda_a}{\lambda}, & \sum_{i=1}^{k} g_i + q_i < n, \ g_s + 1 = g_s', \\ & g_i = g_i' \ (i \neq s, \ i=1,\cdots,k), \\ & \text{there exits a class } a \text{ such that } q_a + 1 = q_a', \\ & q_s - 1 = q_s' \text{ and } q_j = q_j' \ (j \neq a, s, \ j=1,\cdots,k), \\ 0, & \text{otherwise.} \end{cases}$$

Let M_i $(i = 0, 1, 2, \cdots, c+\alpha)$ denote the transition matrix which corresponds to the transition according to one decision of the policy. Now we consider the power of M_i. The matrix of this product corresponds to the transition by successive decisions of the policy at a moment. Note that ${M_i}^j$ converges at most $j = c$, because up to c customers can be served at the same time. Let $\bar{M}_i$ be the converged matrix, i.e., $\bar{M}_i = M_i^c$. The block matrices corresponding to the transition from $c+\alpha+1+i$ $(i \geq 0)$ customers are given by the same matrix M, because the combination of phases is identical. Let $\bar{M}$ be the converged matrix, i.e., $\bar{M} = M^c$.

Now we denote a state probability vector by $z = (z_0, z_1, z_2, \cdots)$, where z_i is a state probability vector when i customers are in the system. We arrange the states in lexicographic order. Then the transition rate matrix of the Markov chain is as follows.

$$Q = \begin{pmatrix} \bar{S}_0 & \bar{U}_0 & & & & & & & \\ \bar{D}_1 & \bar{S}_1 & \bar{U}_1 & & & & & & \\ & \bar{D}_2 & \bar{S}_2 & \bar{U}_2 & & & & & \\ & & \ddots & \ddots & \ddots & & & & \\ & & & \bar{D}_{c+\alpha} & \bar{S}_{c+\alpha} & \bar{U}_{c+\alpha} & & & \\ & & & & \bar{D} & \bar{S} & \bar{U} & & \\ & & & & & \bar{D} & \bar{S} & \bar{U} & \\ & & & & & & \ddots & \ddots & \ddots \end{pmatrix}.$$

Here, $\bar{U}_i = A_i \bar{M}_{i+1}$ $(i = 0, 1, \cdots, c+\alpha)$ corresponds to the transition from i customers to $i+1$ customers. $\bar{D}_i = B_i \bar{M}_{i-1}$ $(i = 1, 2, \cdots, c+\alpha)$ corresponds to the transition from i customers to $i-1$ customers. Note that the block matrices corresponding to the transition from $c+\alpha+1+i$ $(i \geq 0)$ customers are given by the same matrices $\bar{U} = A\bar{M}, \bar{D} = B\bar{M}$. Since the combination of phases is identical when many customers exist, the transition rate matrices are identical as well. $\bar{S}_i$ and $\bar{S}$ are diagonal matrices whose elements are given by

$$-\left(\lambda + \sum_{i=1}^{k} g_i \mu_i\right),$$

when it corresponds to the state $y = (n, g_1, \cdots, g_k, q_1, \cdots, q_k)$.

3.3 Computation of the steady state probabilities for a policy

In this subsection, we compute the steady state probability vector with transition rate matrix Q.

Let e be the column vector where all elements are equal to one. The necessary and sufficient condition to the stability of the system is that the vector π which is computed from

$$\begin{aligned} &\pi(\bar{D} + \bar{S} + \bar{U}) = 0, \\ &\pi e = 1, \end{aligned}$$

satisfies

$$\pi\, D\, e > \pi\, U\, e\ .$$

This means that it is needed for large n that the drift which decreases customers is larger than that which increases customers.

For example, the stability condition of the system with $c = 2$, $c_1 = 1$ and $c_2 = 2$, where customers are served on a FCFS discipline, is given by

$$\frac{\lambda_1\lambda_2}{\mu_1\lambda} + \frac{\lambda_1}{\mu_1} + 2\frac{\lambda_2}{\mu_2} > 2\ .$$

Let x be the steady state probability vector of the Markov chain with transition rate matrix Q. Then x satisfies

$$\begin{aligned} xQ &= 0, \\ xe &= 1. \end{aligned}$$

Solving this equation, we get the steady state probability vector.

Now we consider the computation of the steady state probability vector x. At first, we make the transition probability matrix of the Markov chain. Let θ be the maximum of the absolute value of diagonal elements of Q, and let P be

$$P = I + \frac{1}{\theta}Q = \begin{pmatrix} S_0 & U_0 & & & & & & & \\ D_1 & S_1 & U_1 & & & & & & \\ & D_2 & S_2 & U_2 & & & & & \\ & & \ddots & \ddots & \ddots & & & & \\ & & & D_{c+\alpha} & S_{c+\alpha} & U_{c+\alpha} & & & \\ & & & & D & S & U & & \\ & & & & & D & S & U & \\ & & & & & & \ddots & \ddots & \ddots \end{pmatrix}.$$

Then P satisfies that

$$\begin{aligned} xP &= x(I + \tfrac{1}{\theta}Q) \\ &= x + \tfrac{1}{\theta}xQ \\ &= x. \end{aligned}$$

So we can compute the steady state probability x, solving that

$$\begin{aligned} xP &= x, \\ xe &= 1. \end{aligned}$$

However this is a system of simultaneous equations which have infinite dimension. So we need to use the structure that the block matrices are identical below a certain row.

Now we consider the fundamental period which starts from the state of $n = c+\alpha+1+k$ ($k \geq 0$) and ends at the state of $n = c + \alpha + k$ for the first time. Let G be the matrix whose (i, j) element is the probability that the system starts from state i and ends at state j in the fundamental period. Then G satisfies the following equation.

$$G \ = \ D + SG + UG^2.$$

Using this G, the transition probability matrix under the condition that the number of customers is less than $c + \alpha + 1$ is represented to be

$$T = \begin{pmatrix} S_0 & U_0 & & & & \\ D_1 & S_1 & U_1 & & & \\ & D_2 & S_2 & U_2 & & \\ & & \ddots & \ddots & \ddots & \\ & & & D_{c+\alpha-1} & S_{c+\alpha-1} & U_{c+\alpha-1} \\ & & & & D_{c+\alpha} & S_{c+\alpha} + U_{c+\alpha}G \end{pmatrix}.$$

Using this T, we can compute the steady state probability vector under the condition that the number of customers is less than $c+\alpha+1$. We denote the conditional steady state probability vector by $x' = (x'_0, x'_1, \cdots, x'_{c+\alpha})$. This vector satisfies the following equation.

$$\begin{aligned} x'T &= x', \\ x'e &= 1. \end{aligned}$$

The solution of this equation is obtained as follows. From the last column of T,

$$x'_{c+\alpha-1}U_{c+\alpha-1} + x'_{c+\alpha}(S_{c+\alpha} + U_{c+\alpha}G) = x'_{c+\alpha}.$$

So we can express that

$$x'_{c+\alpha} = x'_{c+\alpha-1}R_{c+\alpha-1}, \quad R_{c+\alpha-1} = U_{c+\alpha-1}(I - S_{c+\alpha} - U_{c+\alpha}G)^{-1}.$$

Next, from the last column but one of T,

$$\begin{aligned} x'_{c+\alpha-2}U_{c+\alpha-2} + x'_{c+\alpha-1}S_{c+\alpha-1} + x'_{c+\alpha}D_{c+\alpha} &= x'_{c+\alpha-1}, \\ x'_{c+\alpha-2}U_{c+\alpha-2} + x'_{c+\alpha-1}S_{c+\alpha-1} + x'_{c+\alpha-1}R_{c+\alpha-1}D_{c+\alpha} &= x'_{c+\alpha-1}. \end{aligned}$$

So we can express that

$$x'_{c+\alpha-1} = x'_{c+\alpha-2}R_{c+\alpha-2}, \quad R_{c+\alpha-2} = U_{c+\alpha-2}(I - S_{c+\alpha-1} - R_{c+\alpha-1}D_{c+\alpha})^{-1}.$$

In the same way, we can express that

$$x'_i = x'_{i-1}R_{i-1}, \quad R_{i-1} = U_{i-1}(I - S_i - R_iD_{i+1})^{-1} \quad (i = c+\alpha-1, c+\alpha-2, \cdots, 1).$$

Letting $x'_0 = 1$, using $x'_i = x'_{i-1}R_{i-1}$ $(i = 1, 2, \cdots, c+\alpha)$, and normalizing them, x'_i $(i = 0, 1, \cdots, c+\alpha)$ are evaluated.

Now we remove the condition on the number of customers. To do so, we consider the cycle in which the system starts from the state of $n = c+\alpha+k$ $(k \geq 0)$ and ends at the state of $n = c+\alpha+k$ for the first time. Let R be the matrix whose (i, j) element is the expectation number of times of state $(c+\alpha+k+1, j)$ visited on condition that the system starts from state i in the cycle. Then R satisfies the following equation.

$$R = U + RS + R^2D.$$

Note that the following equation holds between G and R [4].

$$R = U(I - S - UG)^{-1},$$
$$G = (I - S - RD)^{-1}D.$$

Using this R, we can get

$$x_{c+\alpha+k} = x_{c+\alpha}R^{\,k},$$

so $x'_{c+\alpha+1}$, $x'_{c+\alpha+2}$, $\cdots$ are computed in order.

Now we got the ratio of x'_i $(i = 0, 1, \cdots)$. Finally normalizing these ratios, we get the steady state probability vector x. We summarize the procedure as follows.

1. Compute $G, R_0, \cdots, R_{c+\alpha-1}, R$ respectively.
2. Let $x'_0 = 1$.
3. Compute $x'_i = x'_{i-1}R_{i-1}$ $(i = 1, 2, \cdots, c+\alpha)$.
4. Compute the normalization factor β by

$$\begin{aligned}
\beta &= \sum_{i=0}^{\infty} x'_i e \\
&= \sum_{i=0}^{c+\alpha-1} x'_i e \; + \sum_{i=c+\alpha}^{\infty} x'_i e \\
&= \sum_{i=0}^{c+\alpha-1} x'_i e \; + \sum_{i=0}^{\infty} x'_{c+\alpha} R^i e \\
&= \sum_{i=0}^{c+\alpha-1} x'_i e \; + \; x'_{c+\alpha}(I-R)^{-1} e.
\end{aligned}$$

5. Compute $x_i = x'_i \;/\; \beta \quad (i = 0, 1, \cdots, c+\alpha)$.
6. x_i $(i \geq c+\alpha+1)$ are computed by $x_i = x_{i-1}R$.

3.4 The average waiting time of each class

Using the steady state probability vector, we can compute the average waiting time. Let $L_n^{(i)}(y)$ be the expected number of class i customers, when the phase of the system is $y = (g_1, \cdots, g_k, q_1, \cdots, q_k)$, and the level of the system is n. The number of class i customers in service is given by g_i. The number of class i customers in α customers from the head of the queue is given by q_i. The customer in the rear of the queue is of class i with probability λ_i/λ, because the arrival process is a Poisson process. So $L_n^{(i)}(y)$ is given by

$$L_n^{(i)}(y) = \; g_i + q_i + \tfrac{\lambda_i}{\lambda}(n - \sum_{j=1}^{k}(g_j + q_j)).$$

Let $L_n^{(i)}$ be the vector when we arrange the $L_n^{(i)}(y)$ in lexicographic order of the state. Then the expected number of class i customers $L^{(i)}$ is as follows.

$$\begin{aligned}
L^{(i)} &= \sum_{k=0}^{\infty} x_k L_k^{(i)} \\
&= \sum_{k=0}^{c+\alpha-1} x_k L_k^{(i)} + \sum_{k=c+\alpha}^{\infty} x_k L_k^{(i)} \\
&= \sum_{k=0}^{c+\alpha-1} x_k L_k^{(i)} + \sum_{k=0}^{\infty} x_{c+\alpha} R^k L_{c+\alpha+k}^{(i)}.
\end{aligned}$$

When the number of customers is more than $c+\alpha-1$, the phases of the system are identical. So we obtain the following equations.

$$L^{(i)}_{c+\alpha+k}(y) = L^{(i)}_{c+\alpha}(y) + k\frac{\lambda_i}{\lambda} \qquad (k \geq 0),$$

$$L^{(i)}_{c+\alpha+k} = L^{(i)}_{c+\alpha} + k\frac{\lambda_i}{\lambda}\, e \qquad (k \geq 0).$$

Substituting this equation, we get

$$\begin{aligned}
\sum_{k=0}^{\infty} x_{c+\alpha} R^k L^{(i)}_{c+\alpha+k} &= \sum_{k=0}^{\infty} x_{c+\alpha} R^k (L^{(i)}_{c+\alpha} + k\frac{\lambda_i}{\lambda}\, e) \\
&= \sum_{k=0}^{\infty} x_{c+\alpha} R^k L^{(i)}_{c+\alpha} + \sum_{k=0}^{\infty} x_{c+\alpha} R^k k\frac{\lambda_i}{\lambda}\, e \\
&= x_{c+\alpha}(I-R)^{-1}\, L^{(i)}_{c+\alpha} + \tfrac{\lambda_i}{\lambda} x_{c+\alpha} \sum_{k=0}^{\infty} kR^k e \\
&= x_{c+\alpha}(I-R)^{-1}\, L^{(i)}_{c+\alpha} + \tfrac{\lambda_i}{\lambda} x_{c+\alpha}(I-R)^{-1}\{(I-R)^{-1} - I\}\, e.
\end{aligned}$$

So we obtain the expected number of class i customers.

$$L^{(i)} = \sum_{j=0}^{c+\alpha-1} x_j L^{(i)}_j + x_{c+\alpha}(I-R)^{-1}\, L^{(i)}_{c+\alpha} + \frac{\lambda_i}{\lambda} x_{c+\alpha}(I-R)^{-1}\{(I-R)^{-1} - I\}\, e.$$

Let W_i be the average sojourn time of class i customers, and w_i be the average waiting time of class i customers. Using Little's theorem, we compute these values as follows.

$$W_i = \frac{L^{(i)}}{\lambda_i},$$

$$w_i = W_i - \frac{1}{\mu_i}.$$

4 The characteristics of optimal policy

In this section, we apply the above-mentioned method to the case of $k = 2$. Using the method to all the possible policies, we search the optimal policy. Then we observe the optimal policies for different parameters and criteria to find the tendency of optimal policies.

When the criterion which evaluates the optimality of the policies with the average waiting time of each class is given, the optimal policy is found applying the above-mentioned method to all the policies and calculating the optimality of each policy.

We propose two criteria here; one is the weighed sum of the average waiting times; the other is the average waiting time of a certain class given that the average waiting time of the other class is not greater than a certain value (e.g. the average waiting time in FCFS). Here we call the former criterion 1, the latter criterion 2.

We experiment with two criteria above. After this, we set the parameters as follows; $c = 3, k = 2, c_1 = 1, c_2 = 3, \lambda_1 = \lambda_2 = 0.5$ and $\mu_1 = \mu_2 = 1$.

At first we show the figure of the optimal policy for criterion 1. Note that we can depict any policies as the figures in which the lattice points correspond to the phases. We set the objective function $w_1 + 3w_2$, where w_i $(i = 1, 2)$ is the average waiting time of class i.

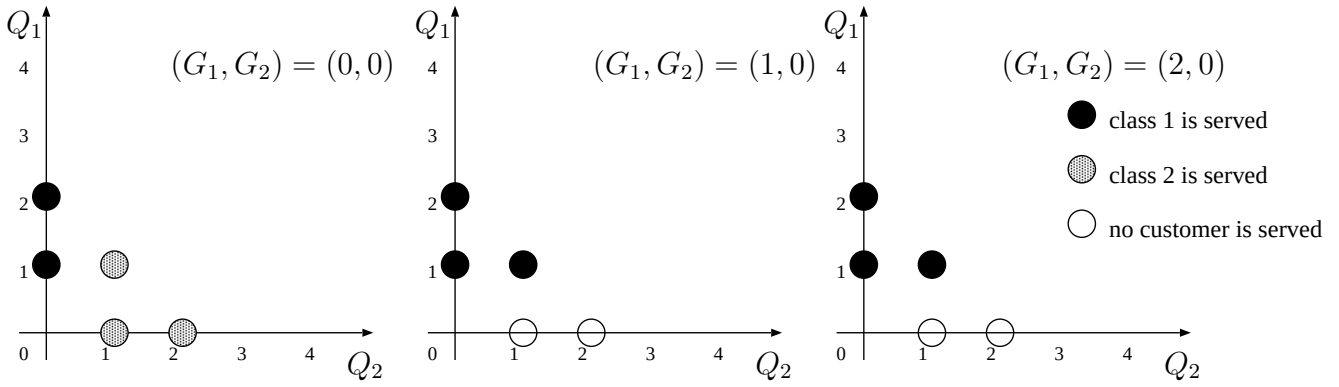

Figure 1: The optimal policy ($\alpha = 2$).

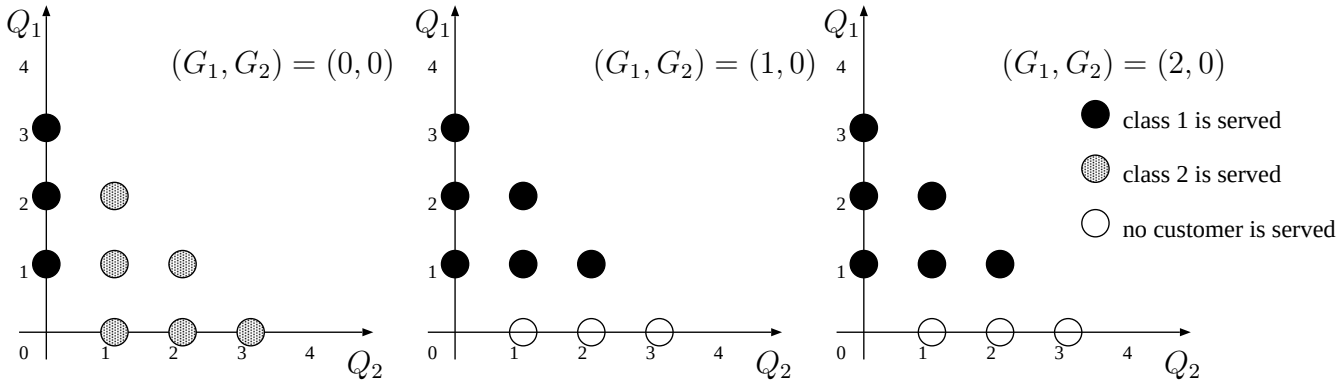

Figure 2: The optimal policy ($\alpha = 3$).

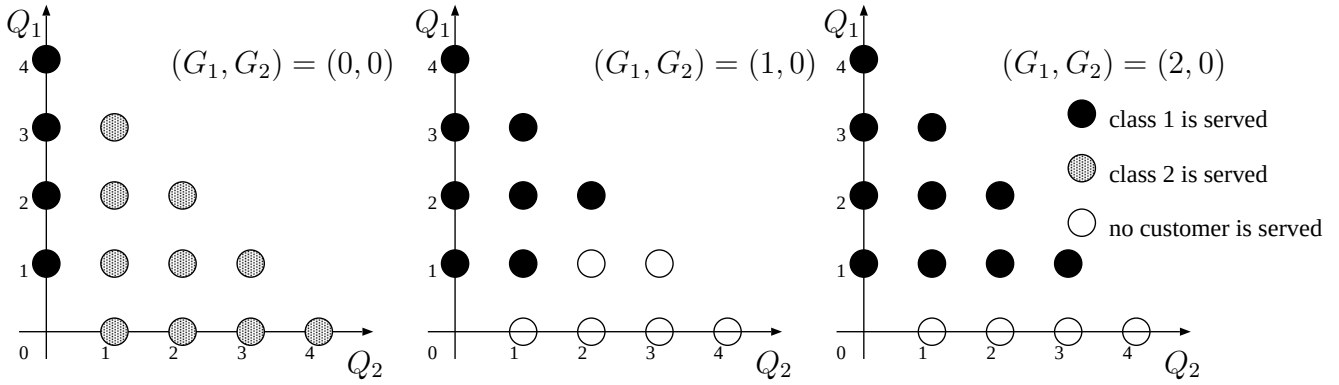

Figure 3: The optimal policy ($\alpha = 4$).

Next, we consider the criterion 2; the average waiting time of a certain class on condition that the average waiting time of the other class is not greater than a certain value. First, we set the objective function w_1 subject to $w_2 \leq w_{2,FCFS}$, where $w_{i,FCFS}$ is the average waiting time of class i when the customers are served on a FCFS basis. The optimal policy is as follows.

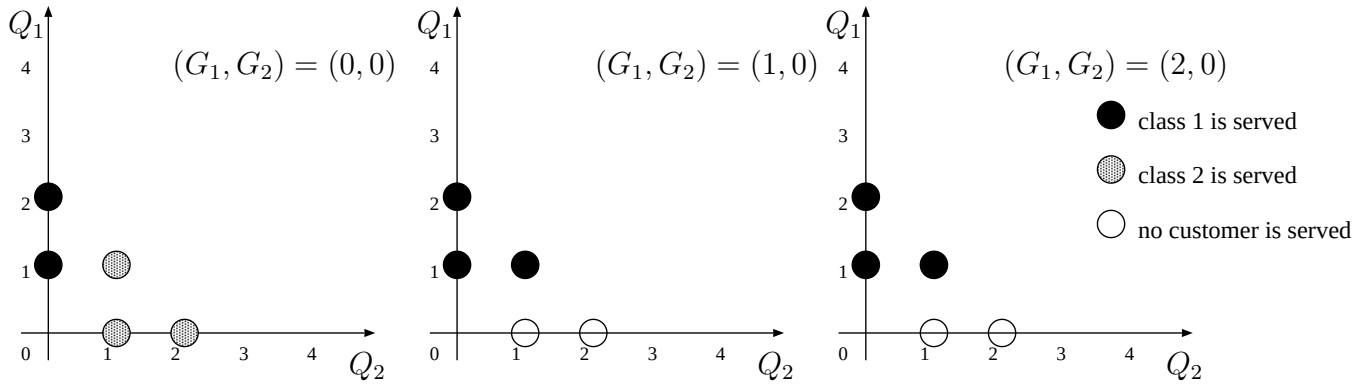

Figure 4: The optimal policy ($\alpha = 2$).

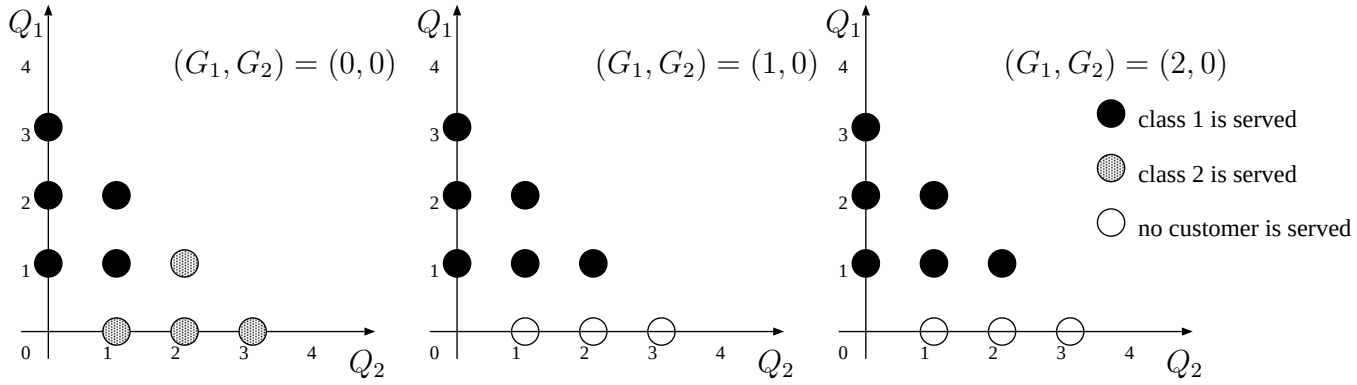

Figure 5: The optimal policy ($\alpha = 3$).

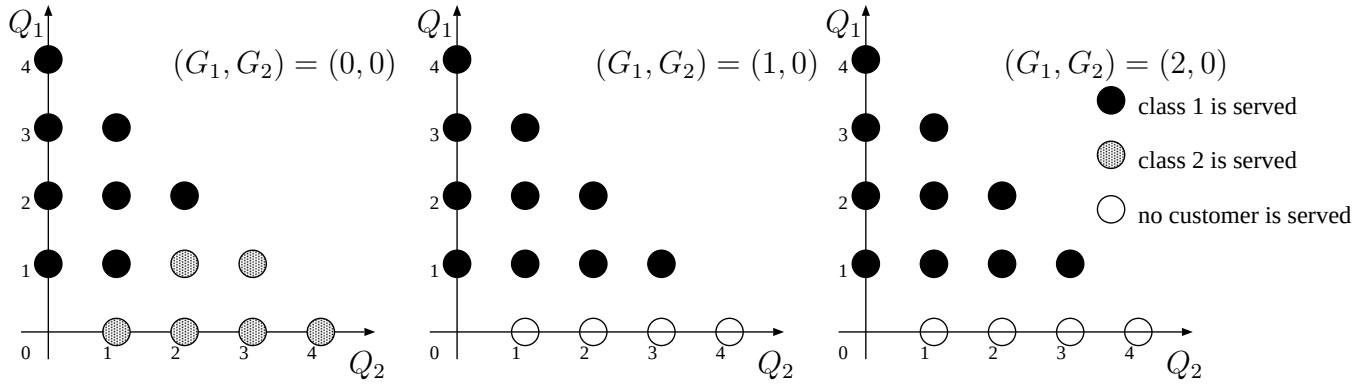

Figure 6: The optimal policy ($\alpha = 4$).

Next we show the optimal policy when we set the objective function w_2 subject to $w_1 \leq w_{1,FCFS}$.

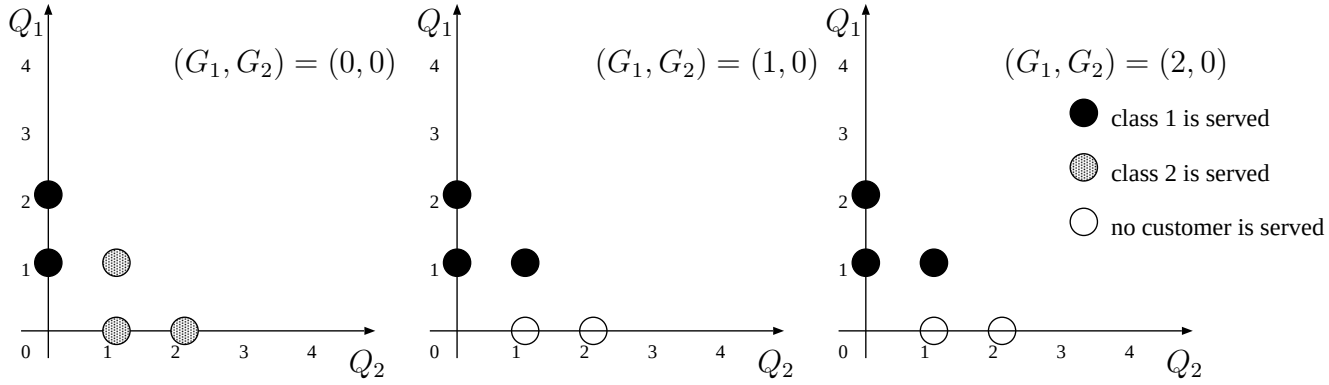

Figure 7: The optimal policy ($\alpha = 2$).

Observing these optimal policies, we can find two tendencies. One is that the optimal policies are formed of a threshold policy. A threshold policy is the policy in which the class of a customer to be served is divided by a switching curve which is nondecreasing and passes the point of the origin. Intuitively, it seems natural that the optimal policy is a threshold policy.

Another tendency of the optimal policies is the relation between the optimal policy for $\alpha = m$ and $\alpha = m + 1$. Let Ω_m^* be the optimal policy for $\alpha = m$ ($m \geq 2$). Ω_m^* and Ω_{m+1}^* are usually resemble each other, but sometimes slightly different. Strictly speaking, in the triangular region of $q_1 + q_2 \leq m - 1$ the class of a customer to be served in Ω_{m+1}^* is the same class as that in Ω_m^*. We can explain this property as follows. The phases which are included in the region $q_1 + q_2 \leq m - 1$ correspond to the same states for the system where $\alpha = m$ and $\alpha = m + 1$. However the phases over the line $q_1 + q_2 = m$ correspond to the different states. It seems natural that the same class customer begins to be served for the same state in optimal policy even if α is different.

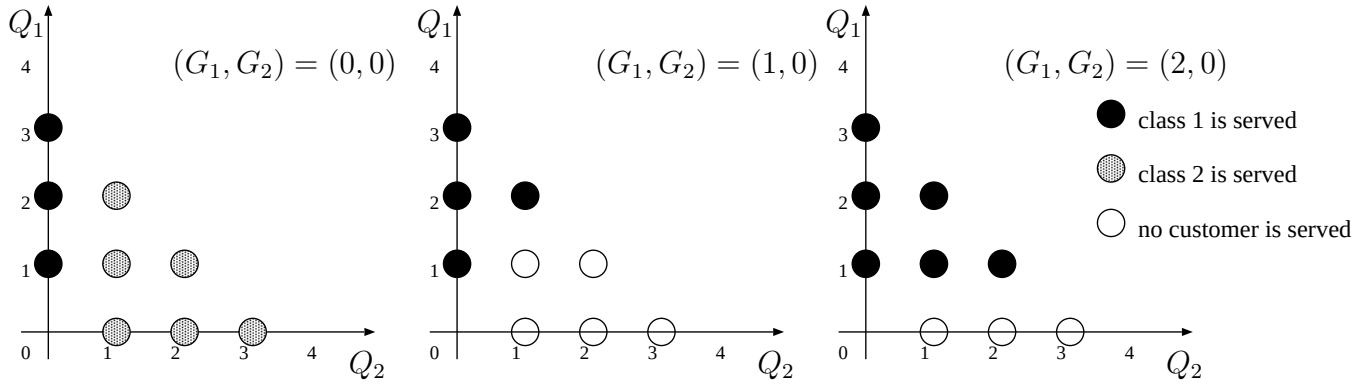

Figure 8: The optimal policy ($\alpha = 3$).

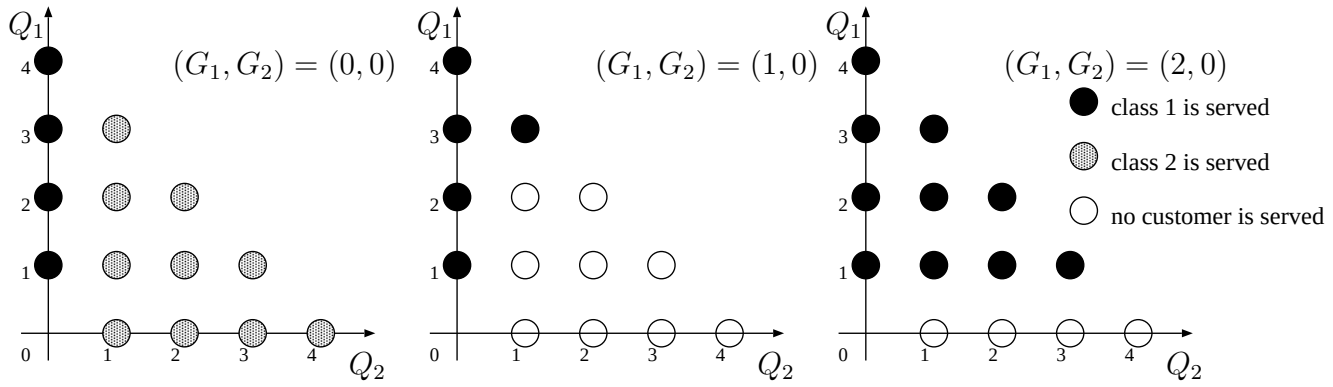

Figure 9: The optimal policy ($\alpha = 4$).

5 A sequential improvement algorithm

For the large α, the search by computing all possible policies needs a very long computational time, because the number of elements in the policy set increases exponentially when α increases. So it is necessary to cut the policies which have no chance of optimality if we want the optimal policy for large α. We consider the methods to cut the unlikely optimal policies.

In this section, we deal with the system of $k = 2$. And we narrow the search to the threshold policies. In addition, we propose a sequential improvement algorithm.

5.1 Threshold policies

At first, we introduce the notation of threshold policies. We denote the number of phase planes by β. Note that β is determined by c and c_i $(i = 1, \cdots, k)$, independent of α. For each phase plane, the switching curve of a threshold policy is expressed as a step function which passes lattice points. We set the lattice points over the switching curve contained in the region which includes q_2 axis. Then we can express a threshold policy Ω_α as β binary α-dimensional vectors as follows.

$$
\begin{aligned}
&\Omega_\alpha = (\boldsymbol{t}_1, \cdots, \boldsymbol{t}_\beta), \\
&\boldsymbol{t}_1 = (t_{1,1}, \cdots, t_{1,\alpha}), \\
&\qquad \vdots \\
&\boldsymbol{t}_\beta = (t_{\beta,1}, \cdots, t_{\beta,\alpha}),
\end{aligned}
$$

where $\boldsymbol{t}_i$ $(i = 1, \cdots, \beta)$ represents the switching curve for the phase plane i. In this thesis, the switching curve starts from the origin, and $t_{i,j}$'s $(i = 1, \cdots, \beta,\ j = 1, \cdots, \alpha)$ specify the switching curve as follows.

If $t_{i,1} = 0$ $(i = 1, \cdots, \beta)$, the switching curve for the phase plane i connects $(q_1, q_2) = (0, 0)$ and $(q_1, q_2) = (0, 1)$ with a horizontal line. If $t_{i,1} = 1$ $(i = 1, \cdots, \beta)$, the switching curve connects $(q_1, q_2) = (0, 0)$ and $(q_1, q_2) = (1, 0)$ with a vertical line. $t_{i,j}$ $(i = 1, \cdots, \beta,\ j \geq 2)$ are defined as follows. When the switching curve for the phase plane i gets to $(q_1, q_2) = (a, b)$, if $t_{i,a+b} = 0$ $(i = 1, \cdots, \beta)$, the switching curve connects $(q_1, q_2) = (a, b)$ and $(q_1, q_2) = (a, b+1)$, if $t_{i,a+b} = 1$ $(i = 1, \cdots, \beta)$, the switching curve connects $(q_1, q_2) = (a, b)$ and $(q_1, q_2) = (a + 1, b)$. By this means, the threshold policy is expressed by β binary α-dimensional vectors.

5.2 The sequential improvement algorithm

In section 4, we observed the characteristics of the optimal policies. In this subsection, we utilize these characteristics to search a quasi-optimal policy for large α. First, we confine the search policies to the threshold policies. Next, when the quasi-optimal policy for $\alpha = m$ is obtained, we restrict the search policies for $\alpha = m + 1$ to the policies which have a similar shape to the quasi-optimal policy for $\alpha = m$.

It was observed that Ω^*_{m+1} $(m \geq 2)$ is identical with Ω^*_m in the region of $q_1 + q_2 \leq m - 1$ in section 4. We note similarities between Ω^*_{m+1} and Ω^*_m. When the optimal policy for $\alpha = m$ is obtained, we propose that we only search the policies whose regions $q_1 + q_2 \leq m - 1$ of all phase planes are identical with Ω^*_m to search the optimal policy for $\alpha = m + 1$. The procedure is as follows.

1. Set $m = 2$, and search the optimal policy.
   Let Ω^*_2 be the optimal policy and $t^*_{i,j} (i = 1, \cdots, \beta,\ j = 1, 2)$ be the elements.
2. If $\alpha = m$, stop, and otherwise $m := m + 1$.
3. Set $t_{i,j} = t^*_{i,j}$ $(i = 1, \cdots, \beta,\ j = 1, \cdots, m - 2)$.
4. Search the optimal $t_{i,j}$ $(i = 1, \cdots, \beta,\ j = m - 1, m)$.
   Let Ω^*_m be optimal policy and $t^*_{i,j} (i = 1, \cdots, \beta,\ j = 1, \cdots, m)$ be the elements.
   Return to step 2.

We call this procedure "a sequential improvement algorithm." It is considered that this algorithm has the following advantages. At first this algorithm enables us to search the quasi-optimal policy for large α, because the number of search policies is independent of α in the step 4. Note that it depends only on β which is determined by c and c_i. In addition this algorithm is expected to keep the optimality because it is based on the common characteristics of optimal policies.

6 Numerical results of the sequential improvement algorithm

In this section, we provide some numerical results using the sequential improvement algorithm for several criteria and parameters. By these numerical results, we consider the properties of batch-arrival batch-service queueing systems. We observe the change of the objective function over α when the arrival rates and criterion are changed. Through this section, we set other parameters as follows; $c = 3, k = 2, c_1 = 1, c_2 = 3$ and $\mu_1 = \mu_2 = 1$.

6.1 The results of criterion 1

In this subsection, we adopt the criterion 1 to evaluate the optimality of the policy. Note that the criterion 1 is a weighed sum of average waiting times. Let the objective function be

$$min \qquad c_1 w_1 + c_2 w_2 = w_1 + 3w_2.$$

This function corresponds to the measure of the waiting time per the job which uses one server.

Figure 10 is the graph of the objective functions of optimal policies under the light load $\lambda_1 = \lambda_2 = 0.1$. Figure 11 and 12 are also the graphs under the medium load $\lambda_1 = \lambda_2 = 0.3$ and the heavy load $\lambda_1 = \lambda_2 = 0.5$, respectively. Note that $\alpha = 1$ corresponds to the case of FCFS. Observing these figures, it is clear that the objective functions of the optimal policies are monotone decreasing and converge as α gets large.

To observe the difference of the improvement factor among the different load, in Figure 13, we show the relative values of objective functions, where the objective function in FCFS is set to be 1 for each load. We discover the following properties. When the load is light, the objective function converges rapidly, and there is little room for the improvement factor. In contrast, when the load is heavy the objective function converges slowly, and there is plenty of room for the improvement factor. It is worthy of attention that the objective function is reduced to about 0.4, and it is achieved at most $\alpha \approx 10$ under the heavy load.

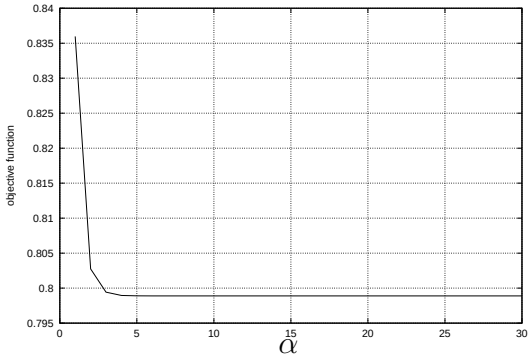

Figure 10: Optimal policy as a function of α ($\lambda_1 = \lambda_2 = 0.1$).

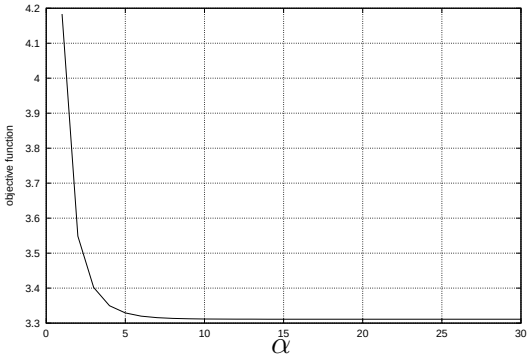

Figure 11: Optimal policy as a function of α ($\lambda_1 = \lambda_2 = 0.3$).

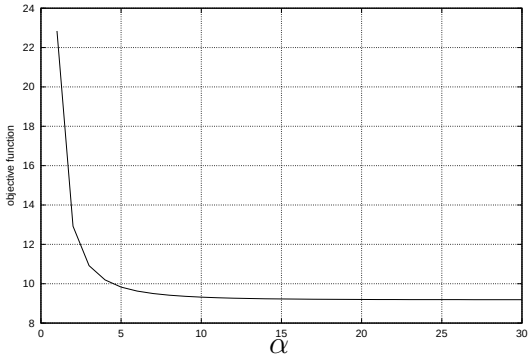

Figure 12: Optimal policy as a function of α ($\lambda_1 = \lambda_2 = 0.5$).

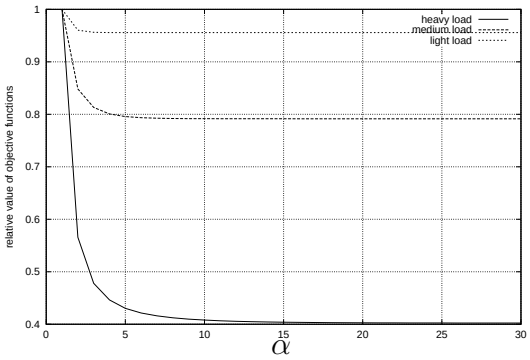

Figure 13: Improvement factor.

6.2 The results of criterion 2

In this subsection, we adopt the criterion 2 to evaluate the optimality of the policy. Note that the criterion 2 is the average waiting time of a certain class on condition that the average waiting time of the other class is not greater than a certain value. Let w_2 be the objective function. And we consider three constraints $w_1 < \infty$, $w_1 \leq w_{1,FCFS}$ and $w_1 \leq w_{1,\Omega_2^*}$, where $w_{1,FCFS}$ is the average waiting time of class 1 in FCFS, and w_{1,Ω_2^*} is the average waiting time of class 1 when the optimal policy for $\alpha = 2$ is applied.

At first, we compare three constraints when $\lambda_1 = \lambda_2 = 0.3$. Figure 14, 15 and 16 are the case of $w_1 < \infty$, $w_1 \leq w_{1,FCFS}$ and $w_1 \leq w_{1,\Omega_2^*}$, respectively. It is observed that the average waiting time of class 2 is most improved in Figure 14. However the average waiting time of class 1 gets worse compared with in FCFS. To keep the fairness of the system, we consider severer constraints $w_1 \leq w_{1,FCFS}$ and $w_1 \leq w_{1,\Omega_2^*}$. In Figure 15 and 16, w_1 converges to its constraint value, and w_2 also converges. Note that w_2 is not monotonic decreasing. This is caused by the discreteness of the policies. We deal with the policies which deterministically decide the class of a customer who begins to be served, so the policies for $\alpha = m+1$ do not include the policies for $\alpha = m$. Hence w_2 is not always monotonous.

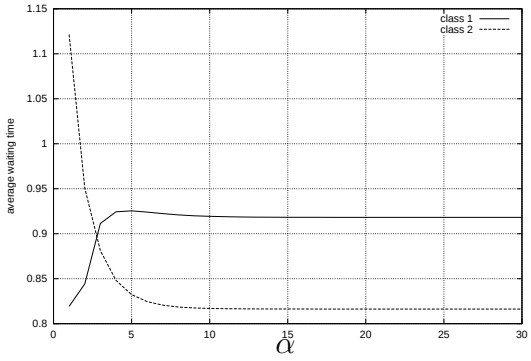

Figure 14: Prioritizing class 2 s.t. $w_1 < \infty$.

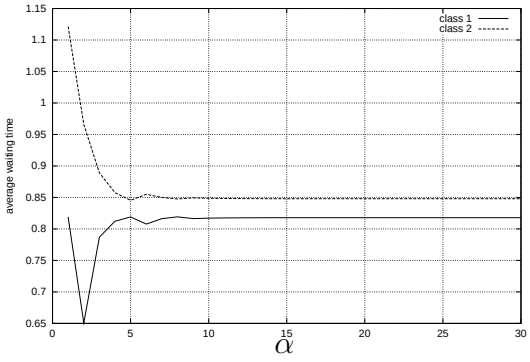

Figure 15: Prioritizing class 2 s.t. $w_1 \leq w_{1,FCFS}$.

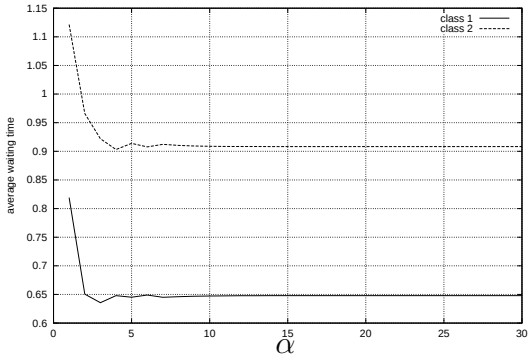

Figure 16: Prioritizing class 2 s.t. $w_1 \leq w_{1,\Omega_2^*}$.

Next, we compare three constraints when $\lambda_1 = \lambda_2 = 0.5$. Figure 17, 18 and 19 are the case of $w_1 < \infty$, $w_1 \leq w_{1,FCFS}$ and $w_1 \leq w_{1,\Omega_2^*}$, respectively. It is of interest that Figure 17 is the same as Figure 18. This result is interpreted as follows. In the optimization of the order of services in batch-arrival batch-service queueing systems, there are two components which reduce the average waiting time of each class. One is the improvement of the total efficiency by the optimization of the order of services. In other words, the improvement of throughput leads to the reduction of the average waiting time for all the classes. The other is the trade-off between classes 1 and 2. Under the heavy load, it is considered that the effect of the improvement of throughput has an extraordinary effect compared with the trade-off.

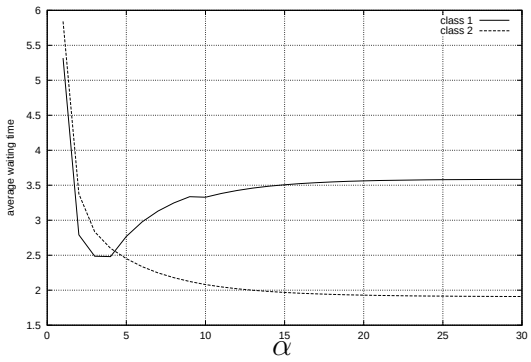

Figure 17: Prioritizing class 2 s.t. $w_1 < \infty$.

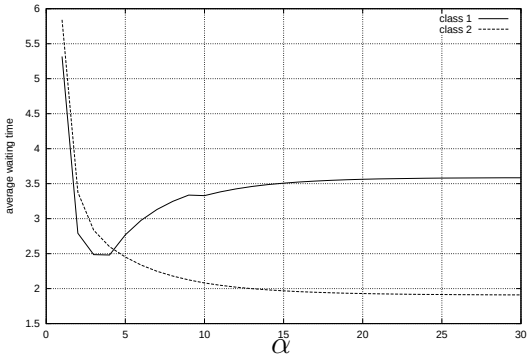

Figure 18: Prioritizing class 2 s.t. $w_1 \leq w_{1,FCFS}$.

7 Conclusion

In batch-arrival batch-service queueing systems, the order of services is very influential in the performance of the system. In particular, under the first-come, first-served discipline, customers are served very inefficiently and the average waiting time gets very long. In this thesis, we considered controlling the order of services to reduce the average waiting time fairly and effectively.

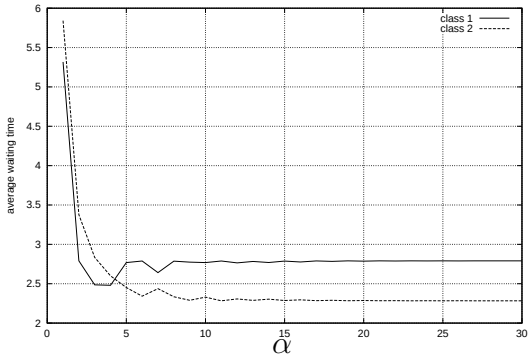

Figure 19: Prioritizing class 2 s.t. $w_1 \leq w_{1,\Omega_2^*}$.

At first, we defined the policy which decides the order of services. To make it possible to analyze the system, we assumed to memorize the classes of at most α ($\alpha > 1$) customers from the head of the queue, and deal with the policies which depend only on the phase. These definitions enable us to formulate the system as a Markov decision process with a countable state space.

Next, we developed an algorithmic method to compute the steady state probability vector and the average waiting time for a given policy. The transition rate matrix has such a structure that it is a tri-block-diagonal-matrix and block matrices are identical below a certain row. This structure enables us to compute the steady state probability vector and the average waiting time efficiently.

Using the above method to all the possible policies, we searched the optimal policy for the case of $k = 2$. Then we observed two tendencies of optimal policies. One is that the optimal policies are formed of the threshold policy. The other is that Ω_m^* and Ω_{m+1}^* are identical in the region $q_1 + q_2 \leq m - 1$ ($m > 1$). Based on these observations, we developed a sequential improvement algorithm to search a quasi-optimal policy for large α.

Using the sequential improvement algorithm for several parameters and criteria, we investigated the properties of the system. Consequently we found the following properties. For a fairly small α, the system performance was drastically improved. When the load is heavy, there is plenty of room for the improvement factor compared with the case that the load is light. Further, there are two components which reduce the average waiting time of each class in batch-arrival batch-service queueing systems. One is the improvement of the total efficiency by the optimization of the order of services. In other words, the improvement of throughput leads to the reduction of the average waiting time for all the classes. The other is the trade-off between class 1 and 2. Under the heavy load, it is considered that the effect of the improvement of throughput has an extraordinary effect compared with the trade-off.

In this thesis, we aimed to achieve fair and efficient service to control the order of services in a batch-arrival batch-service queueing system. Numerical results indicate that the proposed algorithm achieves fair and efficient service under an appropriate criterion.

Acknowledgments

I would like to express a great deal of appreciation to Associate Professor Tetsuya Takine for his significant suggestions and careful guidance to accomplish this thesis. I am very grateful to Professor Masao Fukushima for his valuable comments on this study. I also wish to thank students in Professor Fukushima's laboratory for their support. Lastly, I express my sincere gratitude to my family for their devotions.

References

[1] I. Kino, "A Batch-arrival Batch-service Queueing System," *Proceedings of ITC 13*, pp.671–676, Copenhagen, Denmark, June 19-26, 1991.

[2] K. W. Ross and D. H. K. Tsang, "The Stochastic Knapsack Problem," *IEEE Transactions on Communications*, vol.37, pp.740–747, 1989.

[3] G. Latouche, P. A. Jacobs and D. P. Gaver, "Finite Markov Chain Models Skip-Free in One Direction," *Naval Research Logistics Quarterly*, vol.31, pp.571–588, 1984.

[4] G. Latouche and V. Ramaswami, *Introduction to Matrix Methods in Stochastic Modeling*, ASA and SIAM, Philadelphia, 1999.

[5] Y. Aviv and A. Federgruen, "The Value Iteration Method for Countable State Markov Decision Processes," *Operations Research Letters*, vol.24, pp.223–234, 1999.

[6] R. G. Gallager, *Discrete Stochastic Processes*, Kluwer Academic Publishers, Boston, 1996.

[7] G. Avila-Godoy, A. Brau and E. Fernandez-Gaucherand, "Controlled Markov Chain with Discounted Risk-Sensitive Criteria: Appications to Machine Replacement," *Proceedings of the 36th IEEE CDC*, pp.1115–1120, San Diego, 1997.