STOCHASTIC ORDERING PROPERTIES AND OPTIMAL ROUTING CONTROL FOR A CLASS OF FINITE CAPACITY QUEUEING SYSTEMS

D. Towsley, P. Sparaggis and C. Cassandras

COINS Technical Report 90-72
June, 1990

STOCHASTIC ORDERING PROPERTIES AND OPTIMAL ROUTING CONTROL FOR A CLASS OF FINITE CAPACITY QUEUEING SYSTEMS [1]

Don Towsley
Dept. of Comp. & Inf. Science

Panayotis D. Sparaggis **Christos G. Cassandras**
Dept. of Electr. & Comp. Engineering

University of Massachusetts
Amherst, MA 01003

ABSTRACT

We consider the problem of routing jobs to parallel queues with identical exponential servers and *unequal finite* buffer capacities. Stochastic ordering and weak majorization properties on critical performance measures, such as the joint queue lengths at any time t and the number of customers that are rejected by t, are established by means of event-driven inductions. In particular, we show that the intuitive *Join the Shortest Non-Full Queue* (SNQ) policy is optimal with respect to an overall function that accounts for holding and blocking costs. Moreover, we solve the buffer allocation problem, by proving the intuitive result that, for a fixed total buffer capacity, the optimal allocation scheme is the one in which the difference between the maximum and minimum queue capacities is minimized, i.e., becomes either 0 or 1. Finally, when buffering space is available at the controller, we show that a modified version of the SNQ policy, called the *Shortest Non-Full Queue Delayed* (SNQD) policy, is optimal.

June 22, 1990

submitted to *IEEE Transactions on Automatic Control*

[1]The work of the first author was performed while he was on sabbatical at Laboratoire MASI, Université Pierre et Marie Curie, Paris, France. The work of the other two authors was supported in part by the Office of Naval Research under Contract N00014-87-K-0304, by the National Science Foundation under Grant ECS-8801912 and by NASA under contract NAG 2-595

1 Introduction

A fundamental routing problem arises when jobs arrive at a controller in front of a system consisting of a number of parallel queues with identical servers. The controller is responsible for determining which queue to route an arriving job to so as to minimize (maximize) a cost (respectively reward)-type performance measure. When queues have infinite buffer capacities, the service times are independent and exponentially distributed with the same rate μ, and the controller can observe the queue lengths, the simple *Join the Shortest Queue* (SQ) rule has been shown to be optimal.

The optimality of this intuitive routing policy was first established by Winston in [23]. He proved that the SQ policy maximizes the discounted number of jobs which complete service by a certain time. Having based his analysis on *Markov Decision Processes*, Winston assumed that the arrival process is Poisson. Weber [21] generalized Winston's result by imposing no assumption on the arrival process. Furthermore, he proved that the SQ policy is optimal when the service times are independently and identically distributed (i.i.d.) random variables characterized by an *increasing failure rate* (IFR) (see [15] for a definition).

The above assumptions on the service times are crucial as shown by Whitt in [22]. By means of counterexamples, he demonstrated that there exist service time distributions for which it is not optimal to join the shortest queue, when all service times are independent and identically distributed. He further constructed counterexamples to show that, if in addition the residual service times of the customers in service are known, it is still not optimal to use the *shortest expected delay* (SD) policy (a natural generalization of the SQ policy), under which an incoming customer joins the queue that minimizes its individual expected delay.

Ephremides et al. [7] established the SQ optimality with respect to delay in a system with two queues. They also showed that when it is not possible to observe the queue lengths the optimal routing strategy is to alternate between queues, provided that the initial distributions of the queue sizes are the same. Using the standard uniformization method to convert a continuous-time model into a discrete-time one Johri [11] extended the optimality of the SQ rule to systems with state-dependent exponential service times. Specifically, he showed that with Poisson arrivals the SQ policy stochastically minimizes the number of customers in the system at any time $t > 0$ and also minimizes the long run expected sojourn time if the service rates are non-decreasing concave and bounded with respect to the queue lengths. Finally, Walrand ([20], pp. 260-264), generalized the results of [7] by establishing stochastic ordering relations regarding the queue lengths at all time instants, by means of a forward induction on event (arrivals or service completions) times. All the above authors considered systems consisting of queues with *infinite* buffer capacity.

It is noteworthy that the SQ policy is both *socially* and *individually* optimal. Roughly speaking, individual policies try to be equally fair to all incoming jobs, while social ones attempt to minimize (maximize) a *total* cost (resp. reward) function, to which not all jobs contribute the same (see for example [2]). The SQ policy, for example, is clearly optimal for any incoming job, in the sense that it minimizes its individual expected waiting time. Rarely, however, do

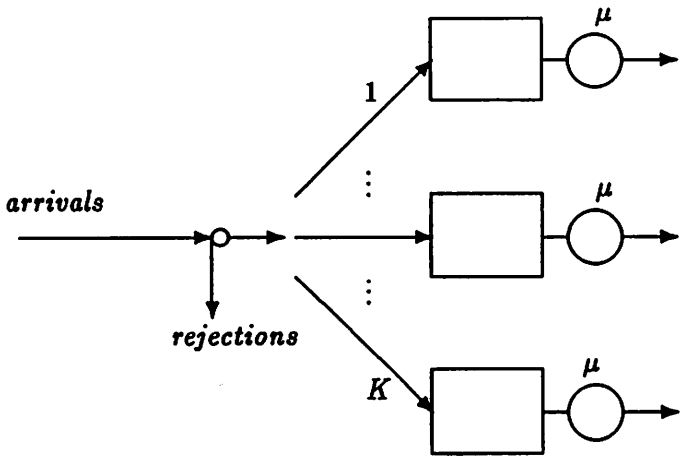

Figure 1: A system with K parallel queues

individual and social policies coincide in optimization problems. In the context of the dynamic routing problem which we discuss, it was shown in [5] that, when the servers have different exponential rates, it is not socially optimal to assign incoming jobs to the queue with the smallest expected time until all present jobs are served. Similar observations were made in [1].

In general, there exist no simple rules for the routing of jobs to stations with different service rates. For a two-station Markovian system with heterogeneous servers and linear cost, the optimal routing policy was shown by Hajek [8] to be characterized by a switching curve in the two-dimensional state space. For the same Markovian system, with K stations, a suboptimal routing policy was recently proposed and evaluated by Krishnan in [12].

In this paper, we address a more difficult question: what is the effect of blocking on the structure of the optimal policy? Assume that all queues have finite capacities so that jobs are rejected when they arrive in front of a *full* system (see also Figure 1). Then, is the SQ policy optimal with respect to *both* throughput and delay, for example? We answer these questions by proving that the *Shortest Non-full Queue*(SNQ) policy is optimal for a wide class of performance measures (including throughput and delay), even when queues have unequal capacities. In particular, the SNQ policy is shown to minimize the total number of jobs present in the system at any time instant t while also minimizing the number of jobs that are rejected by t. This result consitutes a challenge to the classical trade-off between throughput and delay in finite capacity queueing systems. Our results naturally reduce to those obtained in earlier works for the case of systems with infinite capacities.

The optimality of the SNQ rule is established using arguments based on the *weak majorization* of joint queue length vectors under different policies for a single sample path. This majorization leads to a *weak Schur-convex ordering* between the joint queue lengths under the SNQ policy and any other feasible policy at an arbitrary time $t > 0$. A similar ordering was established in [20]. In addition, we prove a strong stochastic ordering between the number of jobs that are rejected by any time $t > 0$, under SNQ and any other feasible policy π. Similar relations are

shown to hold for systems with different buffer allocations (i.e., different ways of distributing a fixed number of buffers to a fixed number of queues), when all systems operate under the SNQ policy. We exploit these orderings to show that the optimal allocation scheme (over the same class of performance measures for which the SNQ policy is optimal) is the one in which the difference between the maximum and minimum capacity queue is minimized, i.e., becomes either 0 or 1. A major technical difficulty for establishing these orderings arises from the fact that queues have *unequal* capacities. This complicates the treatment of arrival events as well as the development of comparison arguments between different policies, especially when different allocation schemes are considered. We overcome this problem by using a variety of technical arguments.

Further, we consider an extended version of the problem, in which buffering space is available at the controller. This implies that whenever a customer arrives at the system, he may be queued at the controller before he is routed to one of the parallel service stations. In this case it is shown that the optimal policy always delays making routing decisions, i.e., holds customers at the controller as long as none of the parallel stations is empty, otherwise it routes customers to empty stations. Moreover, when the controller is full an incoming customer is routed to the station with the shortest queue length. We call this policy the *Shortest Non-Full Queue Delayed* (SNQD) policy.

This paper is organized as follows. In section 2 we define weak majorization and Shur-convex ordering, and present some preliminary results. In section 3 we establish the optimality of the SNQ policy. We first consider systems in which all queues have identical capacities and then extend our results to queues with unequal capacities. Extensions to systems with bulk arrivals and systems with state-dependent arrival processes are discussed in section 4. In section 5 the buffer allocation problem, related to the discussed system, is analyzed, and the optimal allocation scheme is determined. We then establish the optimality of the SNQD policy, in systems with buffers at the controller, in section 6. Finally, our conclusions are given in the last section of this paper.

2 Weak Majorization and Weak Schur-Convex Ordering

In this section we present the mathematical framework on which our analysis will be based and obtain some preliminary results. A more complete treatment of the material used in this section can be found in [14]. Let $\mathbf{N}, \mathbf{M} \in I\!N^K$, $I\!N = \{0, 1, 2, ...\}$, be two arbitrary K-dimensional vectors. We introduce the notation $\hat{N}_k$ to denote the *k-th largest element* in vector $\mathbf{N}$ and define the following dominance relation.

Definition 1 *For any two vectors* $\mathbf{N}, \mathbf{M}$ *we say that* $\mathbf{N}$ *weakly majorizes* $\mathbf{M}$ *(written* $\mathbf{M} \prec_w \mathbf{N}$*) if*

$$\sum_{i=1}^{k} \hat{N}_i \geq \sum_{i=1}^{k} \hat{M}_i, \quad k = 1, \cdots, K.$$

For instance, if $\mathbf{N} = (5,0,3,3,1), \mathbf{M} = (4,1,2,2,3)$ we write $\mathbf{M} \prec_w \mathbf{N}$.

Remark. Whenever $\mathbf{N}$ and $\mathbf{M}$ further satisfy the relation

$$\sum_{i=1}^{K} \hat{N}_i = \sum_{i=1}^{K} \hat{M}_i$$

then $\mathbf{N}$ is said to *majorize* $\mathbf{M}$ (written $\mathbf{M} \prec \mathbf{N}$). Again, the reader is referred to [14] for further details on this relation.

Remark. The definitions of majorization and weak majorization are not restricted to vectors whose components are non-negative integers. However, we have presented them as such as our usage of these comparisons is restricted solely to such vectors.

This dominance relation was used by Walrand in [20] for showing the optimality of the shortest queue (SQ) policy in an infinite capacity system. It has also been used in [19] to show the optimality of certain classes of longest queue policies in the context of a network flow control problem.

Define the following two operators.

Operator D: Let $D_k\mathbf{N}$ denote the vector that results by subtracting a unit quantity from the k-th largest element of $\mathbf{N}$; if $\hat{N}_k = 0$ then we set $D_k\mathbf{N} = \mathbf{N}$.

Operator A: Let $A_k\mathbf{N}$ denote the vector that results by adding a unit quantity to the k-th largest element of $\mathbf{N}$.

We denote the l-th largest element in $D_k\mathbf{N}$ and $A_k\mathbf{N}$ by $(\widehat{D_k\mathbf{N}})_l$ and $(\widehat{A_k\mathbf{N}})_l$ respectively. Note that it is not necessarily true that $(\widehat{D_k\mathbf{N}})_k = (\hat{N}_k - 1, 0)^+$, or, $(\widehat{A_k\mathbf{N}})_k = \hat{N}_k + 1$. For instance, if $\mathbf{N} = (3,3,3)$ then $A_2\mathbf{N} = (4,3,3)$, where $(\widehat{A_2\mathbf{N}})_2 = \hat{N}_2$ rather than $(\widehat{A_2\mathbf{N}})_2 = \hat{N}_2 + 1$. When the vector $\mathbf{N}$ represents queue lengths then the operators D and A correspond to service completions and arrivals respectively.

We now state some conditions under which the relation '$\prec_w$' is preserved with respect to the operators D and A, in the following lemmas. These results will be used later within various proofs throughout the paper.

Lemma 1 *For any two vectors* $\mathbf{N}, \mathbf{M} \in I\!N^K$ *such that* $\mathbf{M} \prec_w \mathbf{N}$ *it follows,*

1. $D_f\mathbf{M} \prec_w D_g\mathbf{N}, \quad g \geq f, \quad g, f \in \{1, \ldots, K\}$

2. $A_f\mathbf{M} \prec_w A_g\mathbf{N}, \quad g \leq f, \quad g, f \in \{1, \ldots, K\}$

Proof: We consider property 1 first. The proof is trivial when $\hat{N}_g = 0$ and $\hat{M}_f = 0$, or, $\hat{N}_g = 0$ and $\hat{M}_f > 0$. The case, $\hat{N}_g, \hat{M}_f > 0$ corresponds to Lemma 5.D.2 in [14, p. 135]. Last, consider the case $\hat{N}_g > 0$ and $\hat{M}_f = 0$. It follows from this inequality that $\hat{M}_i = 0$, $f \leq i \leq K$ and that

$\hat{N}_g > \hat{M}_g = 0$. In addition $\mathbf{M} \prec_w \mathbf{N}$ implies that $\sum_{i=1}^{g-1} \hat{N}_i \geq \sum_{i=1}^{g-1} \hat{M}_i$ which along with the previous observation further implies that $\sum_{i=1}^{l} \hat{N}_i > \sum_{i=1}^{l} \hat{M}_i$, $g \leq l \leq K$. Let $g^* = \max(j \geq g : \hat{N}_j = \hat{N}_g)$. It then follows that $\sum_{i=1}^{l} (\widehat{D_g \mathbf{N}})_i = \sum_{i=1}^{l} \hat{N}_i \geq \sum_{i=1}^{l} \hat{M}_i = \sum_{i=1}^{l} (\widehat{D_f \mathbf{M}})_i$, $1 \leq l < g^*$ and $\sum_{i=1}^{l} (\widehat{D_g \mathbf{N}})_i = (\sum_{i=1}^{l} \hat{N}_i) - 1 \geq \sum_{i=1}^{l} \hat{M}_i = \sum_{i=1}^{l} (\widehat{D_f \mathbf{M}})_i$, $g^* \leq l \leq K$. This completes the proof of property 1.

The proof of property 2 follows in a similar way. ∎

We present another result which is repeatedly used within various proofs in the following sections. This result however is more technical than intuitive, compared to the properties given in the previous lemma.

Lemma 2 *Consider any two vectors* $\mathbf{N}, \mathbf{M} \in I\!N^K$ *with* $\mathbf{M} \prec_w \mathbf{N}$. *Furthermore, assume that* $\sum_{i=1}^{f} \hat{N}_i > \sum_{i=1}^{f} \hat{M}_i$ *for some* $f \in \{1, .., K\}$ *and that* $\hat{M}_g = \hat{M}_{g+1} = ... = \hat{M}_f$ *for some* $g \leq f$. *Then it follows,*

$$\sum_{i=1}^{k} \hat{N}_i > \sum_{i=1}^{k} \hat{M}_i \ \ \forall k = g, \ldots, f. \tag{1}$$

Proof. From the hypothesis $\mathbf{M} \prec_w \mathbf{N}$ it follows

$$\sum_{i=1}^{k} \hat{N}_i \geq \sum_{i=1}^{k} \hat{M}_i \ \ \forall k = g, .., f \tag{2}$$

Suppose that the above equation holds as an equality for some $k \in \{g, .., f\}$. Moreover, let l be the minimum such integer in $\{g, .., f\}$. If $l > 1$ then we have $\sum_{i=1}^{l-1} \hat{N}_i > \sum_{i=1}^{l-1} \hat{M}_i$; by the definition of l we also have

$$\sum_{i=1}^{l} \hat{N}_i = \sum_{i=1}^{l} \hat{M}_i \tag{3}$$

This implies that $\hat{N}_l < \hat{M}_l$. If $l = g = 1$ then by the definition of l it follows that $\hat{N}_l = \hat{M}_l$. Thus, in general,

$$\hat{N}_l \leq \hat{M}_l \tag{4}$$

Since $\hat{N}_l \geq \hat{N}_i$ and $\hat{M}_l = \hat{M}_i$ for $l \leq i \leq f$, it follows from (4) that $\hat{N}_i \leq \hat{M}_i$ for $l \leq i \leq f$. This, combined with (3) implies,

$$\sum_{i=1}^{f} \hat{N}_i \leq \sum_{i=1}^{f} \hat{M}_i$$

This contradicts the lemma's hypothesis. Therefore (2) holds a strict inequality for all $k \in \{g, .., f\}$. ∎

In concluding this section, we define a stochastic ordering among random vectors that is related to weak majorization.

Definition 2 *A function $\phi : I\!N^K \rightarrow I\!R$ is said to be a weak Schur-convex function iff*

$$\mathbf{M} \prec_w \mathbf{N} \Rightarrow \phi(\mathbf{M}) \leq \phi(\mathbf{N}), \forall \mathbf{M}, \mathbf{N} \in I\!N^K.$$

Examples of weak Schur-convex functions include $\sum_{k=1}^K f(N_k)$, for all convex f (e.g., $\sum_{k=1}^K N_k$) and $\max_k N_k$. These functions are related to the class of *Schur-convex functions* defined and studied in [14]. Many of the results for such functions are easily extended to the class of weak Schur-convex functions.

Definition 3 *If* $\mathbf{N}$ *and* $\mathbf{M}$ *are random vectors of dimension* K*, we say that* $\mathbf{N}$ *is larger than* $\mathbf{M}$ *in the sense of weak Schur-convex order (written* $\mathbf{M} \leq_{wscx} \mathbf{N}$*) iff*

$$E[\phi(\mathbf{M})] \leq E[\phi(\mathbf{N})], \text{ for all weak Schur-convex functions } \phi$$

Remark. In the case that $K = 1$, this reduces to the standard stochastic ordering among real-valued random variables (r.v.'s), i.e., for r.v.'s X and Y we write $Y \leq_{st} X$ iff $\Pr(X \leq x) \leq \Pr(Y \leq x)$, $x \in I\!R$.

3 Optimality of the Shortest Non-Full Queue (SNQ) Policy

In this section we establish the optimality of the SNQ policy. We first consider the case where all queues have the same capacity and then extend our results to queues with unequal capacities. The proofs involve sample-path arguments on forward event-driven inductions.

3.1 Queues with equal capacities

We consider a system of K queues, each with its own server, labelled $k = 1, 2, \cdots, K$, which are fed by a single arrival stream. For simplicity, we initially assume that all queues have equal capacities and each queue can store at most B jobs. Let $0 < a_1 < \cdots < a_n < \cdots$ be the sequence of arrival times, i.e., the n-th job arrives at time a_n, and let $\{\tau_n\}_{n=1}^{\infty}$ denote the interarrival times, $\tau_n = a_n - a_{n-1}$, $n = 1, 2, \cdots$, $a_0 = 0$. The customers arrive at a controller which routes them to the different queues. We assume that the service times at each queue are i.i.d. exponential r.v.'s independent of the arrival times as well as the decisions made by the controller. Furthermore, we also assume that service rates in different queues are all equal.

We consider a class of routing policies, Σ, that have instantaneous queue length information available to them and that are required to route jobs to some queue that has available space, if one exists. Define SQ to be the policy that always routes a job to the queue with the least number of jobs. In case of a tie, any rule can be used to choose the destination queue. The characterization *Non-Full* is redundant in the case of queues with equal capacities, since a *full* queue always has the largest queue length; hence, never does the SQ policy route a job to a *full* queue.

Let $\mathbf{N}^\pi(t) = (N_1^\pi(t), \cdots, N_K^\pi(t))$ denote the joint queue lengths at time $t > 0$ under policy $\pi \in \Sigma$. Let $L^\pi(t)$ denote the number of jobs lost due to buffer overflow under policy π by time t. In the following theorem we prove that under SQ the number of jobs that are rejected by any time t is minimized (in a stochastic sense). Moreover, the vector $\mathbf{N}^\pi(t)$ is shown to be larger than $\mathbf{N}^{SQ}(t)$ in the sense of weak Schur-convex order, for any $\pi \in \Sigma$ and all times t. Based on this last result one can immediately conclude that the total number of jobs present in the system at any time t is minimized under the SQ policy.

Theorem 1

$$L^{SQ}(t) \quad \leq_{st} \quad L^\pi(t), \tag{5}$$
$$\mathbf{N}^{SQ}(t) \quad \leq_{wscx} \quad \mathbf{N}^\pi(t). \tag{6}$$

for all $\pi \in \Sigma$, $t > 0$ *provided that* $\mathbf{N}^\pi(0) =_{st} \mathbf{N}^{SQ}(0)$.

Proof. We condition on the arrival times, service times, and initial queue lengths. The proof is by induction on event times (i.e. arrival times or departure times), $t_0 = 0, t_1, t_2, \cdots$. Specifically, we will show that

$$L^{SQ}(t) \quad \leq \quad L^\pi(t), \tag{7}$$
$$\mathbf{N}^{SQ}(t) \quad \prec_w \quad \mathbf{N}^\pi(t). \tag{8}$$

on the given sample path. We consider $L^\pi(0) = L^{SQ}(0) = 0$. Further, we can take initial queue lengths such that $\mathbf{N}^\pi(0) = \mathbf{N}^{SQ}(0)$. Although capital letters are usually reserved to denote random variables, within the proof of the theorem, as well as within all proofs to follow, they also indicate the values of the variables at specific time instants, on *a single sample path.*

To carry on a forward induction we need to couple the systems in the following manner. First, we couple the service completion times at the k-th largest queue under both policies, $k = 1, \cdots, K$. Furthermore, if any queue is empty, we assume that the server serves a fictitious customer and that a customer that arrives to that queue receives the remainder of this service time. The exponential assumption is required here to guarantee that all true service times form a sequence of i.i.d. r.v.'s.

Basis step. By the statement of the theorem, the relations hold for $t = t_0$.

Inductive step. Assume that the relations hold up through $t = t_n$. Clearly they hold for $t_n < t < t_{n+1}$. For $t = t_{n+1}$ we consider the following two cases.

Case 1. Service completion. Suppose that the next event on the given sample path is a completion from the k-th largest queue under both policies. Equation (7) follows easily at t_{n+1}, since,

$$L^\pi(t_{n+1}) = L^\pi(t_n) \geq L^{SQ}(t_n) = L^{SQ}(t_{n+1}).$$

Property 1 of Lemma 1 and the inductive hypothesis ensure that Equation (8) holds at $t = t_{n+1}$.

Case 2. Arrival. Suppose that the next event on the given sample path is an arrival of a customer. SQ routes the customer to the smallest queue and π routes the customer to some arbitrary queue. Clearly the inductive hypothesis $\sum_{i=1}^{K} \hat{N}_i^{\pi}(t_n) \geq \sum_{i=1}^{K} \hat{N}_i^{SQ}(t_n)$ guarantees that $L^{\pi}(t_{n+1}) \geq L^{SQ}(t_{n+1})$. Hence, (7) holds at t_{n+1}. Moreover, if a job is admitted under π, it will also be admitted under SQ. In this case, relation (8) at t_{n+1} is ensured by property 2 of Lemma 1 with $f = K \geq g$, since under SQ the incoming job is always routed to the K-th largest queue. On the other hand, if a job is rejected under π then the π-system is full at time t_{n+1} which ensures that (8) holds trivially at $t = t_{n+1}$. This completes the inductive step.

Removal of the conditioning on arrival times and service times completes the theorem. ∎

Remark. We emphasize the fact that the SQ policy *minimizes the expected number of jobs that are present in the system* at any time instant t, while *maximizing the total number of jobs that were admitted in the system* by t. This is an unusual phenomenon in *finite capacity* queueing networks, where trade-offs between performance measures such as throughput and delay are often encountered (see for example [18], pp. 286-288).

Next, define a cost function of the form

$$\begin{aligned} V_{\alpha}^{\pi}(\mathbf{N}) &= E\left[\int_0^{\infty} e^{-\alpha t}\phi(\mathbf{N}^{\pi}(t))dt|\mathbf{N}(0)=\mathbf{n}\right] \\ &\quad + E\left[\int_0^{\infty} e^{-\beta t}(L^{\pi}(t) - L^{\pi}(t^-))dt|\mathbf{N}(0)=\mathbf{n}\right] \end{aligned} \tag{9}$$

for any weak Schur-convex function ϕ, $\alpha, \beta > 0$, $\mathbf{n} \in \{0, \cdots, B\}^K$, and $\pi \in \Sigma$. Here, the first term accounts for α-discounted holding costs for jobs that are buffered in the system, whereas the second term accounts for β-discounted loss penalties for jobs that are rejected. Holding costs are appropriate to express both throughput and delay in systems with infinite queues. However, in finite capacity systems this is not true. In particular, a policy that minimizes holding costs does in fact maximize the mean delay for all admitted customers, only provided that it also maximizes the throughput.

The discounting factors $e^{-\alpha t}, e^{-\beta t}$ guarantee that the above cost function is well defined over an *infinite* horizon (see [3] for example). We assume that the sequence $\{\tau_n\}_{n=1}^{\infty}$ is such that (9) is finite for at least a policy in Σ. The optimality of the SQ policy is established in the following corollary.

Corollary 1 *SQ minimizes the cost function in (9) over all policies in* Σ.

Proof. The proof follows from the definition of $\leq_{st}$, $\leq_{wscx}$, and Theorem 1. ∎

Remark. Corollary 1 guarantees that the SQ policy is socially optimal. Note however that the same policy is individually optimal at the same time, since each admitted job is routed to the queue that minimizes its individual expected waiting time.

The above two remarks indicate that the SQ policy remains optimal for different - often conflicting - performance criteria. This optimality is due to the symmetry of the system, i.e., the homogeneity of the service rates in the queues. When the service rates are different, the optimal policy is characterized by a general, service-rate-dependent switching curve. Moreover, there is no reason to expect that the policy that minimizes losses due to buffer overflow will be identical to one that minimizes buffer occupancies or job delays when the service rates are different.

3.2 Queues with unequal capacities

In this subsection we determine the optimal routing policy when queues have unequal capacities. Let SNQ (*Shortest Nonfull Queue*) be a policy that always routes a job to the ***non-full*** queue with the least number of customers. In the following lemma we prove that ties can be broken in any arbitrary fashion. This is no longer obvious, since queues with the same queue lengths may have unequal residual capacities. Let $\boldsymbol{B} = (B_1, \ldots, B_K)$ denote the buffer allocation, i.e., queue k has the capacity to store B_k jobs, $1 \leq k \leq K$. Define $\Sigma_{SNQ}(\boldsymbol{B})$ to be the class of SNQ policies for this buffer allocation. When there is no confusion, we omit the argument $\boldsymbol{B}$. The following result states that all policies in Σ_{SNQ} behave the same.

Lemma 3

$$L^{\pi_1}(t) \;=_{st}\; L^{\pi_2}(t), \tag{10}$$

$$\mathbf{N}^{\pi_1}(t) \;=_{st}\; \mathbf{N}^{\pi_2}(t) \tag{11}$$

for any $\pi_1, \pi_2 \in \Sigma_{SNQ}$, $t > 0$ *provided that* $\mathbf{N}^{\pi_1}(0) =_{st} \mathbf{N}^{\pi_2}(0)$.

Proof. Let $\mathbf{N}, \mathbf{M} \in \{0, 1, .., \hat{B}_1\}^K$ be two arbitrary vectors. Throughout the rest of the paper, when considering queue lengths, we break ties by defining *smaller elements to correspond to queues with less buffers*, i.e., if $\hat{N}_{i+1} = \hat{N}_i$ then the queue with $\hat{N}_{i+1}$ jobs (which for simplicity we refer to as 'queue $\hat{N}_{i+1}$') has capacity greater than or equal to the queue $\hat{N}_i$.

The proof is similar to that of Theorem 1. Again we condition on arrival times, service times, and initial queue lengths, and use the same coupling argument as in Theorem 1. Specifically we show that

$$L^{\pi_1}(t) \;=\; L^{\pi_2}(t), \tag{12}$$

$$\mathbf{N}^{\pi_1}(t) \;=\; \mathbf{N}^{\pi_2}(t). \tag{13}$$

for all sample paths. The notation $\mathbf{N} = \mathbf{M}$ implies that $\sum_{i=1}^{k} \hat{N}_i = \sum_{i=1}^{k} \hat{M}_i$, for all $k = 1, .., K$. We take $L^{\pi_1}(0) = L^{\pi_2}(0) = 0$. Since the initial queue lengths are equal in distribution we assume that $\mathbf{N}^{\pi_1}(0) = \mathbf{N}^{\pi_2}(0)$. Assuming that relations (12), (13) hold at $t = t_n$ we show that they also hold at $t = t_{n+1}$. Note that by the induction hypothesis, i.e., relation (13) at time t_n, it follows that

$$\hat{N}_j^{\pi_1}(t_n) = \hat{N}_j^{\pi_2}(t_n) \;\; j = 1, .., K \tag{14}$$

Due to the above relation, if t_{n+1} is a service completion epoch, then relations (12), (13) follow immediately for $t = t_{n+1}$.

Consider now the case of an arrival at time t_{n+1}. Let $\hat{N}_g^{\pi_1}, \hat{N}_f^{\pi_2}$ be the queues (also queue lengths at time t_{n+1}^-) in which the job is admitted under π_1, π_2 respectively, at time t_{n+1}. Throughout this paper the time variable is occasionally omitted at time $t = t_n$. For instance we sometimes write $\hat{N}_l^{\pi}$ instead of $\hat{N}_l^{\pi}(t_n)$.

Let $g^* = \max(j \geq g : \hat{N}_j^{\pi_1} = \hat{N}_g^{\pi_1}, \hat{N}_j^{\pi_1}$ is non-full). Likewise, let $f^* = \max(j \geq f : \hat{N}_j^{\pi_2} = \hat{N}_f^{\pi_2}, \hat{N}_j^{\pi_2}$ is non-full). Then all queues $\hat{N}_j^{\pi_1}$, $j > g^*$ are full. To prove this suppose that some $\hat{N}_l^{\pi_1}$, $l \in \{g^*+1, \ldots, K\}$ is not full. Recalling that $\hat{N}_{g^*}^{\pi_1} \geq \hat{N}_l^{\pi_1}$ it is seen that the case $\hat{N}_{g^*}^{\pi_1} = \hat{N}_l^{\pi_1}$ contradicts the definition of g^*, whereas the case $\hat{N}_{g^*}^{\pi_1} > \hat{N}_l^{\pi_1}$ contradicts the fact that $\pi_1 \in \Sigma_{SNQ}$. Likewise, all queues $\hat{N}_j^{\pi_2}$, $j > f^*$ are full.

Now assume that $f^* \neq g^*$; in particular consider without loss of generality (w.l.g.) that $f^* < g^*$. Next, we prove that this assumption leads to a contradiction. Since $f^* < g^*$ we know that queue $\hat{N}_{g^*}^{\pi_2}$ is full. This implies that there exist at least $(K - g^* + 1)$ queues in the system (i.e. queues $\hat{N}_{g^*}^{\pi_2}, \ldots, \hat{N}_K^{\pi_2}$), with capacity less than or equal to $\hat{N}_{g^*}^{\pi_2}$. On the other hand, since $\hat{N}_{g^*}^{\pi_1}$ is not full, we conclude that there exist at most $(K - g^*)$ queues in the system (i.e. queues $\hat{N}_{g^*+1}^{\pi_1}, \ldots, \hat{N}_K^{\pi_1}$) with capacity less than or equal to $\hat{N}_{g^*}^{\pi_1}$. This is because the $\hat{N}_{g^*}^{\pi_1}$ queue has capacity greater than $\hat{N}_{g^*}^{\pi_1}$ and all queues $\hat{N}_1^{\pi_1}, .., \hat{N}_{g^*-1}^{\pi_1}$ have either strictly more than $\hat{N}_{g^*}^{\pi_1}$ customers (therefore they have capacity greater than $\hat{N}_{g^*}^{\pi_1}$) or exactly $\hat{N}_{g^*}^{\pi_1}$ customers, in which case they have capacity greater than or equal to the capacity of the $\hat{N}_{g^*}^{\pi_1}$ queue (due to our convention for breaking ties on queue lengths by ordering queues in decreasing capacity). Since $\hat{N}_{g^*}^{\pi_1} = \hat{N}_{g^*}^{\pi_2}$, we arrive at a contradiction.

We have therefore proved that $f^* = g^*$. Then it follows that $\hat{N}_{g^*}^{\pi_1} = \hat{N}_{f^*}^{\pi_2}$ by the induction hypothesis (14) at t_n. Since $\hat{N}_g^{\pi_1} = \hat{N}_{g^*}^{\pi_1}$ and $\hat{N}_f^{\pi_2} = \hat{N}_{f^*}^{\pi_2}$, we get $\hat{N}_g^{\pi_1} = \hat{N}_f^{\pi_2}$ which proves (14) at time t_{n+1}. This implies (13) at time t_{i+1}. Finally, (12) is implied at t_{n+1} by the induction hypothesis (i.e., (13) at t_i) and the definition of Σ_{SNQ}. This completes the inductive step.

Removal of the conditioning on arrival times, service times and initial queue lengths completes the proof of the lemma. ∎

The following theorem generalizes Theorem 1 to the case of unequal buffer capacities.

Theorem 2

$$L^{\gamma}(t) \leq_{st} L^{\pi}(t), \tag{15}$$

$$\mathbf{N}^{\gamma}(t) \leq_{wscx} \mathbf{N}^{\pi}(t). \tag{16}$$

for all $\pi \in \Sigma$, $\gamma \in \Sigma_{SNQ}$, $t > 0$ *provided that* $\mathbf{N}^{\pi}(0) =_{st} \mathbf{N}^{\gamma}(0)$.

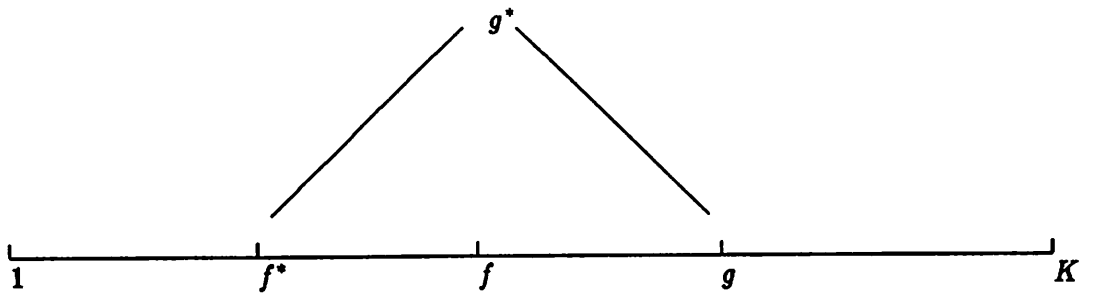

Figure 2: Index ordering for Theorem 2.

Proof. We couple the two systems and carry a forward induction on event times, on a single arbitrary sample path, exactly as in Theorem 1. The proof of Theorem 2 is the same as that of Theorem 1 for service completions. However, for arrival instants it becomes more complicated. Let t_{n+1} be an arrival instant, with the induction hypothesis holding at time t_n. Next, we consider the case in which a job is admitted into the system under both γ and π. The cases in which the arriving job is rejected by π, or by both γ and π can be treated as in Theorem 1. Clearly, since $\mathbf{N}^\gamma(t_i) \prec_w \mathbf{N}^\pi(t_i)$ the incoming job cannot be admitted only under π.

Regarding the number of jobs that are rejected by time t_{n+1}, it is seen that

$$L^\pi(t_{n+1}) = L^\pi(t_n) \geq L^\gamma(t_n) = L^\gamma(t_{n+1})$$

where the inequality above follows from the induction hypothesis. It remains to show that $\mathbf{N}^\gamma(t_{n+1}) \prec_w \mathbf{N}^\pi(t_{n+1})$.

Let $\hat{N}_f^\gamma, \hat{N}_g^\pi$ be the queues (at time t_{n+1}^-) to which the job is routed, under γ and π respectively, at time t_{n+1}. Due to Lemma 3 we assume w.l.g. that ties under γ are broken by sending the job to the queue with the largest index. This implies that all queues $\hat{N}_{f+1}^\gamma, .., \hat{N}_K^\gamma$ are full. Let $g^* = \min\{j \leq g : \hat{N}_j^\pi(t_n) = \hat{N}_g^\pi(t_n)\}$ and likewise $f^* = \min\{j \leq f : \hat{N}_j^\gamma(t_n) = \hat{N}_f^\gamma(t_n)\}$. It is seen that,

$$\sum_{i=1}^{l} \hat{N}_i^\pi(t_{n+1}) = \sum_{i=1}^{l} \hat{N}_i^\pi(t_n),\ l < g^*;\ \sum_{i=1}^{l} \hat{N}_i^\pi(t_{n+1}) = [\sum_{i=1}^{l} \hat{N}_i^\pi(t_n)] + 1,\ l \geq g^* \qquad (17)$$

Likewise,

$$\sum_{i=1}^{l} \hat{N}_i^\gamma(t_{n+1}) = \sum_{i=1}^{l} \hat{N}_n^\gamma(t_n),\ l < f^*;\ \sum_{i=1}^{l} \hat{N}_i^\gamma(t_{n+1}) = [\sum_{i=1}^{l} \hat{N}_n^\gamma(t_n)] + 1,\ l \geq f^* \qquad (18)$$

Due to the above equations if $f^* \geq g^*$ it immediately follows that $\mathbf{N}^\pi(t_{n+1}) \succ_w \mathbf{N}^\gamma(t_{n+1})$. We now consider the case $g^* > f^*$. If $g \leq f$ then property 2 of Lemma 1 can be used to complete the proof. Hence, we only have to consider the case $g > f$ and $g^* > f^*$ (see Figure 2). Clearly,

due to (17), (18) it suffices to prove the following strict inequality.

$$\sum_{i=1}^{l} \hat{N}_i^{\pi} > \sum_{i=1}^{l} \hat{N}_i^{\gamma} \quad l = f^*, .., g^* - 1 \tag{19}$$

Recall that throughout the paper we sometimes omit the time variable at t_n as in the previous relation.

First we prove that $\hat{N}_i^{\gamma} \geq \hat{N}_i^{\pi}$ for $i = f+1, \ldots, K$. Suppose that $\hat{N}_i^{\pi} > \hat{N}_i^{\gamma}$ for some $i \in \{f+1, \ldots, K\}$. The fact that queue $\hat{N}_i^{\gamma}$ is full implies that there exist at least $(K - i + 1)$ queues in the system (i.e. queues $\hat{N}_i^{\gamma}, \ldots, \hat{N}_K^{\gamma}$) with capacity less than or equal to $\hat{N}_i^{\gamma}$. Due to the assumption $\hat{N}_i^{\pi} > \hat{N}_i^{\gamma}$ it is also seen that there exist at most $(K - i)$ queues with capacity less than or equal to $\hat{N}_i^{\gamma}$ (because there are i queues, i.e., queues $\hat{N}_1^{\pi}, \ldots, \hat{N}_i^{\pi}$, with more than $\hat{N}_i^{\gamma}$ jobs). This leads to a contradiction.

Furthermore, it is not difficult to show that $\hat{N}_g^{\gamma}$ is strictly greater than $\hat{N}_g^{\pi}$. Suppose that $\hat{N}_g^{\gamma} = \hat{N}_g^{\pi}$. Then, the fact that queue $\hat{N}_g^{\gamma}$ is full implies that there exist at least $(K - g + 1)$ queues in the system with capacity less or equal than $\hat{N}_g^{\gamma}$. But queue $\hat{N}_g^{\pi}$ is not full, which implies that there exist at most $(K - g)$ queues with capacity less or equal than $\hat{N}_g^{\pi}$. In view of $\hat{N}_g^{\gamma} = \hat{N}_g^{\pi}$ we arrive at a contradiction.

Therefore, we have already proved

$$\begin{aligned} \hat{N}_i^{\gamma} &\geq \hat{N}_i^{\pi}, \quad i = f+1, \ldots, K \\ \hat{N}_g^{\gamma} &> \hat{N}_g^{\pi}, \quad g > f \end{aligned}$$

The last two equations imply that

$$\sum_{i=k}^{K} \hat{N}_i^{\gamma} > \sum_{i=k}^{K} \hat{N}_i^{\pi} \quad k = f+1, \ldots, g \tag{20}$$

Using $\sum_{i=1}^{K} \hat{N}_i^{\pi} \geq \sum_{i=1}^{K} \hat{N}_i^{\gamma}$ (induction hypothesis) and (20), we can support the first and second inequality respectively, in the following relation:

$$\sum_{i=1}^{k} \hat{N}_i^{\pi} + \sum_{i=k+1}^{K} \hat{N}_i^{\pi} \geq \sum_{i=1}^{k} \hat{N}_i^{\gamma} + \sum_{i=k+1}^{K} \hat{N}_i^{\gamma} > \sum_{i=1}^{k} \hat{N}_i^{\gamma} + \sum_{i=k+1}^{K} \hat{N}_i^{\pi} \quad k = f, .., g-1 \tag{21}$$

This implies

$$\sum_{i=1}^{k} \hat{N}_i^{\pi} > \sum_{i=1}^{k} \hat{N}_i^{\gamma}, \quad k = f, \ldots, g-1 \tag{22}$$

By Figure 2 it is seen that in order to prove (19) we must extend (22) to $k = f^*, .., f-1$. This is a straightforward application of Lemma 2, considering (22) for $k = f$ and noting that $\hat{N}_{f^*} = \hat{N}_{f^*+1} = ... = \hat{N}_f$, which completes the induction step.

Removal of the conditioning on arrival times, service times and initial queue lengths completes the proof. ∎

Formally, the optimality of a policy $\gamma \in \Sigma_{SNQ}$ is stated in the following corollary of Theoerem 2.

Corollary 2 *Any policy $\gamma \in \Sigma_{SNQ}$ minimizes the cost function in (9) over all policies in Σ.*

4 Extensions and Comments on the SNQ optimality

In this section we comment on the generality of the discussed model. Further, we extend our model to include systems with bulk arrivals and a class of state-dependent arrival processes.

4.1 Cost functions for SNQ-optimal routing policies

Clearly, Corollaries 1 and 2 are also valid if the cost function in (9) is replaced by another function, defined over a finite horizon, with or without discounting factors. In this case no stability-related assumption needs to be imposed on the interarrival times. Thus, our results include previous results, regarding both throughput and delay, which were proved for finite horizons. Throughput optimality, in the sense of maximizing the expected number of jobs that complete service by some time t, was established in [21,23]. Delay optimality, in the sense of minimizing the expected total time for the completion of service of all jobs that arrive within a finite period, was established in [7].

Furthermore, the model may include holding costs which are not necessarily linear on the queue lengths (a popular assumption in some classical queueing control problems, e.g., [4,13]) and can also be non-stationary. The weak Schur-convex nature of the function ϕ in (9) can be exploited to account for the designer's desire to keep the queue lengths at the stations below certain thresholds by adopting a proper dynamic routing policy. In computer systems for instance, the buffering space may be shared by other processing units. Overutilizing this common space may be discouraged by introducing sharply increasing holding costs when the queue lengths exceed certain thresholds.

4.2 Bulk arrivals

In the case of bulk arrivals, similar arguments to those used in section 2.2. can be invoked to show that the optimal policy is the natural generalization of the SQ (or SNQ) policy. Specifically, the optimal policy distributes the bulk in such a way that the workload in the queues 'balances'. For instance, if $\mathbf{N}(t^-) = (5, 2, 1)$, all queues have capacity greater than 5, and a bulk of 7 jobs arrives at time t, the optimal assignment of the jobs to the queues is the one under

which $\mathbf{N}(t) = (5,5,5)$. Formally, the proof copies that of Theorem 2, on a sample path with 7 consecutive arrivals and no service completions in between. In general, given any $\mathbf{N}(t^-)$ and any bulk size, the optimal routing policy γ is such that $\mathbf{N}^\gamma(t) \prec_w \mathbf{N}^\pi$, for any feasible policy π.

4.3 A class of state-dependent arrival processes

We consider arrival processes with decreasing rates, with respect to the total number of jobs in the system. Let $P(t) = \sum_{i=1}^{K} \hat{N}_i(t)$. We focus on arrival processes $a(P(t))$, such that if $P_1(t) > P_2(t)$ then $a(P_1(t))$ can be obtained from $a(P_2(t))$ via the so-called *thinning process* (see [6] for example). Under this reducing process, sample paths of $a(P_1(t))$ can be obtained from sample paths of $a(P_2(t))$, using Bernoulli rejection with some parameter p, $0 \leq p \leq 1$. For instance a Poisson process with rate $\lambda(P_2(t)) = \lambda_2$ can be thinned to a Poisson process with rate $\lambda(P_1(t)) = \lambda_1$, whenever $\lambda_2 > \lambda_1$.

In view of Theorem 2, whenever $P^\pi(t) > P^\gamma(t)$ there may occur additional arrivals under γ until some time $t' > t$ at which it becomes $P^\pi(t') = P^\gamma(t')$. The induction, however, can be carried out without any major modification. Arrival rates with decreasing state-dependent rates can better model communication systems with flow control and congestion-avoidance mechanisms.

5 The Optimal Buffer Allocation Problem

In this section we determine the optimal allocation of B buffers to K parallel queues ($B \geq K$), which are all fed by a single arrival stream. In general, the allocation of buffers to stations in queueing networks is a difficult and sometimes analytically intractable problem. Given the complexity of the problem, the solutions that are proposed are often approximate [16] or based on heuristics [10]. Perturbation analysis techniques have also been proposed towards gradient-like optimization [9].

Nevertheless, for the problem addressed in this paper we can take advantage of the fact that the optimal routing policy has been already determined, in order to specify a *unique* allocation scheme, which is optimal in the sense of minimizing (9). Let $B = (B_1, \cdots, B_K)$ be an allocation scheme such that $\sum_{i=1}^{K} B_i = B$, $B_i \geq 1$ for all $i = 1, .., K$. Let

$$\mathcal{B} = \{B = (B_1, \cdots, B_K) \; : \; \sum_{k=1}^{K} B_k = B, B_i \geq B_{i+1} \geq 1, i = 1, \cdots, K-1\} \tag{23}$$

denote the class of all feasible allocation schemes. Define the scheme $B^o = (B_1^o, \cdots, B_K^o)$ such that

$$B_i^o = \begin{cases} \lfloor B/K \rfloor + 1, & B \bmod K \neq 0, i = 1, .., B \bmod K, \\ \lfloor B/K \rfloor & \text{otherwise,} \end{cases} \tag{24}$$

i.e., the B_i's can differ by one at most. We will show subsequently that B^o is the optimal allocation scheme. We begin the analysis in this section with a preliminary lemma.

Lemma 4

$$B^o \prec_w B, \quad \forall B \in \mathcal{B}$$

Proof. Follows from the definition of B^o and "$\prec_w$". ∎

Next, we only consider systems that employ SNQ *optimal* policies. We modify our earlier notation so that whenever we are interested in the behavior of a system under SNQ, when the buffer allocation is determined by some scheme $B^b \in \mathcal{B}$, we will use the superscript of B and write $L^b(t)$, $\mathbf{N}^b(t)$.

Lemma 5 *If* $B^2 \prec_w B^1$, *then*

$$L^2(t) \quad \leq_{st} \quad L^1(t), \tag{25}$$

$$\mathbf{N}^2(t) \quad \leq_{wscx} \quad \mathbf{N}^1(t) \tag{26}$$

for all $B^1, B^2 \in \mathcal{B}$, $t > 0$ *provided that* $\mathbf{N}^2(0) =_{st} \mathbf{N}^1(0)$.

Proof. The proof of this lemma is similar to that of Theorem 2. We remind the reader that we have adopted the convention that $\hat{N}^i_j = \hat{N}^i_{j+1}$ implies that $B^i_{r^i(j)} \geq B^i_{r^i(j+1)}$, where r^i is a mapping of the ordered queue lengths into the buffer capacities of those queues when the buffer allocation is B^i, i.e., the queue lengths are ordered in decreasing value, and in case of ties, decreasing value of buffer capacity. Furthermore, due to Lemma 3 we may assume that whenever there are two or more queues, each with the smallest queue length, customers are routed to the one with the largest index as in Theorem 2. The proof is based on conditioning on the service times and arrival times and showing

$$L^2(t) \quad \leq \quad L^1(t), \tag{27}$$

$$\mathbf{N}^2(t) \quad \prec_w \quad \mathbf{N}^1(t). \tag{28}$$

by induction on the different event times, $t_0 = 0, t_1, \ldots, t_n, \ldots$.

Basis step. Follows from the initial conditions.

Inductive step. Assume that relations (27) and (28) hold for $t_0, \ldots, t_n$. We show that they hold for t_{n+1}. The case of a departure is handled as in Theorem 1. We consider the case of an arrival. Here there are two cases according to whether there is a loss under B^1. The case of a loss under B^1 is handled as in Theorem 1. In the case of no loss, let the arrival be to the queue with the g-th largest queue length under B^1, and the queue with the f-th largest queue length under B^2. Let $g^* = \min\{j \leq g : \hat{N}^1_j(t_n) = \hat{N}^1_g(t_n)\}$ and likewise $f^* = \min\{j \leq f : \hat{N}^2_j(t_n) = \hat{N}^2_f(t_n)\}$. As in Theorem 2, the non-trivial case is when $g > f$ and $g^* > f^*$, in which it suffices to prove, as in (19) that

$$\sum_{i=1}^{l} \hat{N}^1_i > \sum_{i=1}^{l} \hat{N}^2_i, \quad f^* \leq l \leq g^* - 1, \tag{29}$$

Next, we show that

$$\sum_{i=1}^{l} \hat{N}_i^1 > \sum_{i=1}^{l} \hat{N}_i^2, \quad f \leq l < g, \tag{30}$$

Then (29) can be proved by extending (30) over $\{f^*, .., f-1\}$ via Lemma 2.

Due to $\sum_{i=1}^{K} \hat{N}_i^1 \geq \sum_{i=1}^{K} \hat{N}_i^2$, (30) can be proved by showing that

$$\sum_{i=l}^{K} \hat{N}_i^1 < \sum_{i=l}^{K} \hat{N}_i^2, \quad f < l \leq g. \tag{31}$$

Recall that $B^2 \prec_w B^1$ is equivalent (by the definition of B in (23)) to $\sum_{i=j}^{K} \hat{B}_i^1 \leq \sum_{i=j}^{K} \hat{B}_i^2$, $1 \leq j \leq K$. According to the definitions of f and g,

$$\begin{aligned} \sum_{i=l}^{K} \hat{N}_i^1 &= \sum_{i=l}^{K} \hat{B}_i^1, \quad g < l \leq K, \\ \sum_{i=l}^{K} \hat{N}_i^2 &= \sum_{i=l}^{K} \hat{B}_i^2, \quad f < l \leq K. \end{aligned}$$

Consider the following cases.

Case 1: Assume that, for $f < l \leq g$, the queues with the $(K-l+1)$ smallest queue lengths under B^1 also correspond to the queues with the $(K-l+1)$ smallest buffer capacities under B^1. It follows

$$\sum_{i=l}^{K} \hat{N}_i^1 < \sum_{i=l}^{K} \hat{B}_i^1 \leq \sum_{i=l}^{K} \hat{B}_i^2 = \sum_{i=l}^{K} \hat{N}_i^2 \tag{32}$$

The first inequality is due to the fact that the queue with the g-th largest queue length under B^1 is not full. This yields (31).

Case 2: The second case occurs when there is not a one-to-one correspondence between the queues with the $(K-l+1)$ smallest capacities and the $(K-l+1)$ smallest queue lengths under B^1. Then the first strict inequality in (32) is not straightforward. Let S_l denote the subset of the queues with the $(K-l+1)$ smallest capacities which are not in the set of queues with the $(K-l+1)$ smallest queue lengths, and $\mathcal{T}_l$ denote the subset of queues with the $(K-l+1)$ smallest queue lengths which do not belong in the set of queues with the $(K-l+1)$ smallest capacities. Then it follows that $B^1_{\tau^1(i)} \geq \hat{N}_i^1 \geq \hat{N}_j^1$, for all i, j s.t. $\hat{N}_i^1 \in S_l$ and $\hat{N}_j^1 \in \mathcal{T}_l$, since $i < l \leq j$.

Furthermore, it can be proved that $\hat{N}_y^1 > \hat{N}_z^1$ for some $\hat{N}_y^1 \in S_l, \hat{N}_z^1 \in \mathcal{T}_l$. Specifically, by the definition of $S_l, \mathcal{T}_l$ it follows that there exists some pair of queues $\hat{N}_y^1 \in S_l, \hat{N}_z^1 \in \mathcal{T}_l$ such that $B^1_{\tau^1(y)} < B^1_{\tau^1(z)}$, i.e., there exists a queue in $\mathcal{T}_l$ with capacity strictly greater than some queue in S_l. Then we cannot have $\hat{N}_y^1 = \hat{N}_z^1$, since in that case our convention for breaking ties on queue lengths would be contradicted. Since $\hat{N}_y^1 \geq \hat{N}_z^1$ (because $y < l \leq z$) it follows that $\hat{N}_y^1 > \hat{N}_z^1$.

In summary we have shown that $B^1_{\tau^1(i)} \geq \hat{N}^1_j$ for all i, j s.t. $\hat{N}^1_i \in S_l$ and $\hat{N}^1_j \in \mathcal{T}_l$, and in particular $B_{\tau^1(y)^1} \geq \hat{N}^1_y > \hat{N}^1_x$ for some $\hat{N}^1_y \in S_l$, $\hat{N}^1_x \in \mathcal{T}_l$. Thus

$$\sum_{i=l}^{K} \hat{N}^1_i < \sum_{i=l}^{K} B^1_i$$

which then leads to (32) again. This establishes (31), which completes the induction.

Removal of the conditioning on arrival and service times completes the proof of the lemma. ∎

An immediate consequence of Lemmas 4 and 5 is the following theorem.

Theorem 3 *For any feasible allocation scheme $B^b \in \mathcal{B}$ it follows*

$$L^o(t) \leq_{st} L^b(t) \tag{33}$$

$$\mathbf{N}^o(t) \leq_{wscx} \mathbf{N}^b(t) \tag{34}$$

$t > 0$ provided that $\mathbf{N}^o(0) \leq_{wscx} \mathbf{N}^b(0)$.

Therefore the buffer allocation scheme B^o provides the optimum performance in the following sense.

Corollary 3 *B^o minimizes the cost function in (9) over the class $\mathcal{B}$.*

Note that B^o is the most 'balanced' of all schemes in $\mathcal{B}$. The optimality of B^o is intuitive in the sense that all queues have the same service rate. Last, note that the same scheme was shown to provide the optimal throughput in a two-station cyclic network, in [17].

6 A System with a Queueing Facility at the Controller

In this section we consider a system in which buffer space is available at the controller. This allows the controller the option of queueing a customer after his arrival to the system for a period of time, before he is routed to one of the K queueing/service stations (see Figure 3).

In this system, routing desicions can be taken not only at arrival and departure instants, but also at arbitrary time instants, i.e., whenever there are some customers at the controller. A similar problem was studied by Lin and Kumar in [13]. In their model a controller with infinite buffer capacity precedes two single servers which have no queueing facilities. The servers have different rates. It was shown that the optimal policy, with respect to holding costs, always keeps the slow server idle as long as the queue at the controller is below a certain threshold.

We continue to study policies that never allow the controller to reject a job if there exists space at one or more of the K stations or at the controller. Let Σ^* denote this class of routing

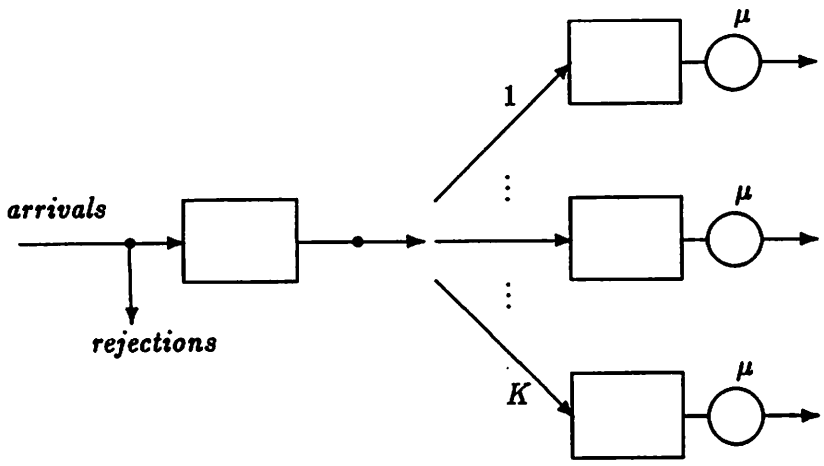

Figure 3: A system with a queueing facility and K parallel stations

policies. We show that the optimal policy always delays making routing decisions, i.e. holds the customers at the controller's facility, as long as all K servers are busy. If all of the controller's buffers are occupied then a new incoming job is routed to the station which has the shortest queue length. Let Σ^*_{SNQD} be this class of optimal policies. Clearly, $\Sigma^*_{SNQD} \subset \Sigma^*$.

We continue to label the queues associated with the servers as $k = 1, \ldots, K$ and the new queue at the controller as $k = 0$. Let B_k denote the queue capacity at the k-th queue, $0 \leq k \leq K$. Let $\mathbf{N}^\pi(t)$, $N^\pi_k(t)$ and $\hat{N}^\pi_k(t)$, $k = 1, .., K$, $\pi \in \Sigma^*(t)$, be defined as in the previous sections. Furthermore, let $N^\pi_0(t)$ denote the number of customers that occupy the controller at time t, when the system operates under policy π and let $N^\pi(t) = \sum_{k=0}^{K} N^\pi_k(t)$ denote the total number of customers in the system.

The intuition behind the optimality of a $SNQD$ policy is that if customers were allowed to be routed to non-empty stations (at times in which there is available buffering space at the controller) then the total time in which servers remain idle would be increased, as a result of the statistical fluctuations of the service times. In other words, if a customer is routed to a non-empty station, then he may remain in queue for a long time interval during which other stations may become empty. Hence, customers may not receive service although there exist available idle servers in the system.

Let $\Sigma^*_{NI} \subset \Sigma^*$ denote the class of non-idling routing policies, where a non-idling policy is one that never leaves work in the controller queue while there is an idle server. We begin our study of the system by establishing that the optimal policy belongs to Σ^*_{NI}.

Lemma 6 *For any policy $\pi \in \Sigma^*$ there exists a policy $\gamma \in \Sigma^*_{NI}$ such that*

$$\begin{aligned} L^\gamma(t) &\leq_{st} L^\pi(t), \\ N^\gamma(t) &\leq_{st} N^\pi(t). \end{aligned}$$

for $t > 0$, provided that $\mathbf{N}^\gamma(0) =_{st} \mathbf{N}^\pi(0)$, and $N^\gamma_0(0) =_{st} N^\pi_0(0)$.

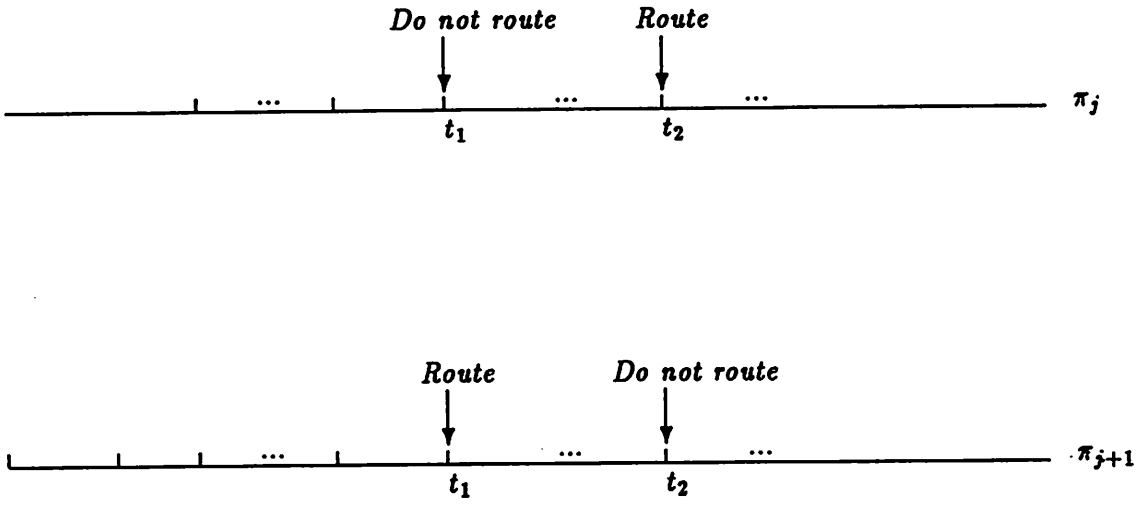

Figure 4: Construction of π_{j+1} in Lemma 6.

Proof. As before, we condition on initial queue lengths, service times and arrival times. We will construct a sequence of policies $\pi_0 = \pi, \pi_1, \ldots, \pi_j, \ldots$ such that

$$L^{\pi_{j+1}}(t) \leq_{st} L^{\pi_j}(t), \tag{35}$$
$$N^{\pi_{j+1}}(t) \leq_{st} N^{\pi_j}(t). \tag{36}$$

for all times $t > 0$, and either $\gamma = \pi_j$ for some $j < \infty$ or $\gamma = \lim_{j\to\infty} \pi_j$. We now describe how π_{j+1} is constructed from π_j.

Let t_1 be the first event time at which π_j deviates from a non-idling policy, i.e., a service completion occurs at some server k such that $N_k^{\pi_j}(t_1^+) = 0$ and $N_0^{\pi_j}(t_1^+) > 0$. First, we let π_{j+1} 'copy' π_j at all routing instants in the interval of time $[0, t_1)$. At time t_1, π_{j+1} routes a customer to server k. Suppose that the first routing event after time t_1 under π_j occurs at time t_2 and suppose that π_j routes to server l at that time. Policy π_{j+1} is not allowed to route any customers during the interval $(t_1, t_2]$. Hence it does not route a customer at time t_2 (see also Figure 4). Clearly,

$$L^{\pi_j}(t) = L^{\pi_{j+1}}(t), \tag{37}$$
$$N^{\pi_j}(t) \geq N^{\pi_{j+1}}(t), \tag{38}$$

for all $t \leq t_2$ and

$$\mathbf{N}^{\pi_{j+1}}(t_2^+) \prec_w \mathbf{N}^{\pi_j}(t_2^+), \tag{39}$$
$$N_0^{\pi_{j+1}}(t_2^+) = N_0^{\pi_j}(t_2^+). \tag{40}$$

This is because server k had no customers prior to the customer routing by π_{j+1} at time t_1 and server l had zero, one, or more customers prior to the customer routing by π_j at time t_2. Also note that the customer which was routed at t_1 under π_{j+1} may complete service by time t_2. Finally, let π_{j+1} copy π_j at all routing instants occuring in the interval (t_2, ∞). This construction along with equation (40) ensures that

$$N_0^{\pi_j}(t) = N_0^{\pi_{j+1}}(t), \;\; t > t_2 \tag{41}$$

Moreover, relations (39), (41) can be used to initiate an induction argument similar to the one contained in Theorem 2 to establish

$$\mathbf{N}^{\pi_{j+1}}(t) \prec_w \mathbf{N}^{\pi_j}(t), \;\; t > t_2 \tag{42}$$

Equations (41) and (42) imply,

$$L^{\pi_j}(t) \geq L^{\pi_{j+1}}(t), \;\; t > t_2$$

and so (35), (36) follow immediately.

We have constructed policy π_{j+1} that satisfies relations (35) and (36). Furthermore, if we let t_j denote the first time at which π_j deviates from a non-idling policy, then $t_{j+1} > t_j$, for $0 \leq j$. Consequently, we have constructed a sequence of policies such that either π_j is a non-idling policy for some j or there is an infinite sequence of non-idling policies such that the times at which the policies deviate from a non-controller idling policy form an increasing, unbounded sequence (the unboundedness is due to the exponential service time assumption). In this case, $\gamma = \lim_{j \to \infty} \pi_j$. Removal of the conditioning on the service times, arrival times, and initial queue lengths yields the desired result. ∎

As a consequence of Lemma 6, we can now restrict our attention to policies in Σ^*_{NI}. We show that all policies in Σ^*_{SNQD} outperform any policy π in Σ^*_{NI}.

Theorem 4

$$L^\gamma(t) \;\; \leq_{st} \;\; L^\pi(t), \tag{43}$$
$$N^\gamma(t) \;\; \leq_{st} \;\; N^\pi(t). \tag{44}$$

for all $\pi \in \Sigma^*_{NI}, \gamma \in \Sigma^*_{SNQD}$, $t > 0$ *provided that* $\mathbf{N}^\pi(0) =_{st} \mathbf{N}^\gamma(0)$.

Proof. As in the previous theorems we condition on initial queue lengths, service times, and arrival times. Specifically, we show that

$$L^\gamma(t) \;\; \leq \;\; L^\pi(t), \tag{45}$$
$$\mathbf{N}^\gamma(t) \;\; \prec_w \;\; \mathbf{N}^\pi(t), \tag{46}$$
$$N^\gamma(t) \;\; \leq \;\; N^\pi(t) \tag{47}$$

for any sample path and for all $t > 0$.

Since routing decisions can be taken at arbitrary times under policy π, we carry the induction on the union of the event times under π and the event times under γ. For convenience we treat routing events due to service completions as separate events from the service completions themselves. For instance, if the n-th event is a service completion at time t followed immediately by the routing of a customer to that station, then the routing is the $(n+1)$-th event and $t_n = t$, $t_{n+1} = t^+$.

Clearly (45)-(47) hold at time $t = 0$. Let us assume that they hold at time t_n and we will prove that they also hold at time t_{n+1}. We consider the following cases.

Case 1. Service completion. Suppose that the next event at time t_{n+1} is a service completion from the l-th largest queue. Equation (45) holds trivially at time t_{n+1}. Equation (46) follows from the induction hypothesis, i.e. (46) at time t_n and property 1 of Lemma 1.

As to equation (47) it follows easily if the service completion is real under γ. If it is fictitious under γ then by the definition of γ it follows that $N_0^\gamma(t_n) = 0 = N_0^\gamma(t_{n+1})$. Hence,

$$N^\pi(t_{n+1}) \geq \sum_{i=1}^{K} \hat{N}_i^\pi(t_{n+1}) \geq \sum_{i=1}^{K} \hat{N}_i^\gamma(t_{n+1}) = N^\gamma(t_{n+1})$$

Case 2. Routing event, no arrival. Suppose that the next event at time t_{n+1} is a routing event under π (where t_{n+1} is not an arrival instant), or a routing event under γ (following a service completion), or both. Equations (45), (47) follow trivially at time t_{n+1}.

As to equation (46) the only case in which it does not follow readily is when a customer is routed only under γ. This only occurs when $\hat{N}_K^\gamma(t_n) = 0$. Let $f^* = \min\{j \geq 1 : \hat{N}_j^\gamma(t_n) = 0\}$. Consider the following subcases.

2.1. $\hat{N}_K^\pi(t_n) = 0$. Since a customer is routed under γ we have $N_0^\gamma(t_n) > 0$. On the other hand, since no customer is routed under π and $\hat{N}_K^\pi(t_n) = 0$ it follows that $N_0^\pi(t_n) = 0$ (by the definition of Σ_{NI}^*). Therefore, using (47) at t_n we get,

$$\sum_{i=1}^{K} \hat{N}_i^\pi(t_n) \geq \sum_{i=1}^{K} \hat{N}_i^\gamma(t_n) + N_0^\gamma(t_n) > \sum_{i=1}^{K} \hat{N}_i^\gamma(t_n)$$

and by Lemma 2,

$$\sum_{i=1}^{k} \hat{N}_i^\pi(t_n) > \sum_{i=1}^{k} \hat{N}_i^\gamma(t_n), \quad k = f^*, .., K$$

From the last equation (46) follows easily at t_{n+1}.

2.2. $\hat{N}_K^\pi(t_n) > 0$. By the induction hypothesis, i.e., $\mathbf{N}^\gamma(t_n) \prec_w \mathbf{N}^\pi(t_n)$, it follows that

$$\sum_{i=1}^{k} \hat{N}_i^\pi(t_n) \geq \sum_{i=1}^{k} \hat{N}_i^\gamma(t_n), \quad k \leq f^* - 1 \tag{48}$$

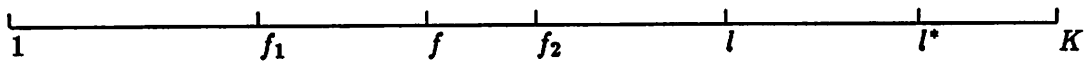

Figure 5: Index ordering for Theorem 4, case 3.

and

$$\sum_{i=1}^{k} \hat{N}_i^{\pi}(t_n) > \sum_{i=1}^{k} \hat{N}_i^{\gamma}(t_n), \;\; k \geq f^* \qquad (49)$$

since $\hat{N}_k^{\pi}(t_n) > 0 = \hat{N}_k^{\gamma}(t_n)$ for $k \geq f^*$. From the last two equations it follows easily that $\mathbf{N}^{\gamma}(t_{n+1}) \prec_w \mathbf{N}^{\pi}(t_{n+1})$.

Case 3. Arrival. Suppose the next event is an arrival. Equations (45), (47) follow easily at time t_{n+1} by the induction hypothesis, i.e., (45), (47) at time t_n.

As to equation (46) it follows as in *case 2* when the customer is routed under π or when it is routed only under γ and $\hat{N}_K^{\gamma}(t_n) = 0$. Next, we consider the only remaining case, i.e., the customer is routed only under γ because $N_0^{\gamma}(t_n) = B_0$, $\hat{N}_K^{\gamma}(t_n) > 0$.

Since the incoming customer is kept at the controller at time t_{n+1} under π we have $N_0^{\pi}(t_n) < B_0 = N_0^{\gamma}(t_n)$, which via (47) at t_n implies

$$\sum_{i=1}^{K} \hat{N}_i^{\pi}(t_n) > \sum_{i=1}^{K} \hat{N}_i^{\gamma}(t_n) \qquad (50)$$

Let $\hat{N}_f^{\gamma}$ be the queue to which the customer is routed under γ and define $f_1 = \min\{j \leq f : \hat{N}_j^{\gamma}(t_n) = \hat{N}_f^{\gamma}(t_n)\}$ (see also Figure 5). Clearly it suffices to prove that

$$\sum_{i=1}^{k} \hat{N}_i^{\pi}(t_n) > \sum_{i=1}^{k} \hat{N}_i^{\gamma}(t_n), \;\; k = f_1, .., K \qquad (51)$$

Let $f_2 = \max\{j \geq f : \hat{N}_j^{\gamma}(t_n) = \hat{N}_f^{\gamma}(t_n)\}$. We only prove (51) for $k \in \{f_2, .., K\}$. Then by Lemma 2 we can extend the result on $\{f_1.., f_2\}$. We proceed by contradiction. Suppose that

$$\sum_{i=1}^{l} \hat{N}_i^{\pi}(t_n) = \sum_{i=1}^{l} \hat{N}_i^{\gamma}(t_n)$$

for some $l \in \{f_2, .., K\}$. Let $l^* = \min\{j > l : \sum_{i=1}^{j} \hat{N}_i^{\pi}(t_n) > \sum_{i=1}^{j} \hat{N}_i^{\gamma}(t_n)\}$. Note that $l^* \leq K$ exists, due to (50). This implies, $\sum_{i=1}^{l^*-1} \hat{N}_i^{\pi}(t_n) = \sum_{i=1}^{l^*-1} \hat{N}_i^{\gamma}(t_n)$. Therefore, $\hat{N}_{l^*}^{\pi}(t_n) > \hat{N}_{l^*}^{\gamma}(t_n)$.

It is now seen that there exist at least l^* queues with capacity strictly greater that $\hat{N}^{\gamma}_{l^*}$, i.e, queues $\hat{N}^{\pi}_1, .., \hat{N}^{\pi}_{l^*}$. On the other hand, since $\gamma \in \Sigma^*_{SNQD}$ it follows that all queues $\hat{N}^{\gamma}_{l^*}, .., \hat{N}^{\gamma}_K$ are full. Therefore, there exist at most $l^* - 1$ queues, i.e. queues $\hat{N}^{\gamma}_1, .., \hat{N}^{\gamma}_{l^*-1}$, with capacity strictly greater than $\hat{N}^{\gamma}_{l^*}$. Hence, we arrive at a contradiction.

This completes the induction.

Removal of the conditioning on arrival times, service times and initial queue lengths completes the theorem ∎

Define a cost function of the form

$$V^{\pi}_{\alpha}(\mathbf{N}) = E\left[\int_0^{\infty} e^{-\alpha t}\phi(N^{\pi}(t))dt|\mathbf{N}(0)=\mathbf{n}\right] + E\left[\int_0^{\infty} e^{-\beta t}(L^{\pi}(t) - L^{\pi}(t^-))dt|\mathbf{N}(0)=\mathbf{n}\right] \quad (52)$$

for a nondecreasing function ϕ, $\alpha, \beta > 0$, $\mathbf{n} \in \{0, \cdots, B\}^K$, and $\pi \in \Sigma^*$. The optimality of any policy in Σ^*_{SNQD} is formally established in the following corollary.

Corollary 4 *Any policy $\gamma \in \Sigma^*_{SNQD}$ minimizes the cost function in (52) over all policies in Σ.*

Note that the cost function in (9) has to be modified. This is because under a policy $\gamma \in \Sigma^*_{SNQD}$ a long queue may be formed at the controller, whereas under some other policy $\pi \in \Sigma^*$ customers might be equally distributed among the K service stations. Then, the previous corollary might fail to hold if instead of (52) we maintained (9).

We have not discussed the buffer allocation problem for a system with buffer space associated with the controller. However, it should be clear that similar arguments as those in Section 5 can be used to show that the optimal buffer allocation will allocate the controller as many buffers as possible (as constrained by design limitations), and that the parallel queues will receive the remaining buffers in the most balanced fashion, i.e., as if there were no controller.

7 Conclusions

We have considered the problem of dynamic routing for a class of finite capacity queueing systems which consist of a number of parallel queues with identical exponential servers. We have treated both symmetric and non-symmetric systems, in which queues have equal or unequal capacities respectively. Furthermore, we considered systems in which buffering space is available at the controller, i.e., the routing decision maker. In these systems routing decisions should be delayed in order to diminish the time during which servers stay idle, improving the overall system's performance.

Using a partial order relation on the queue lengths we have constructed a sample path comparison framework, which enabled us to establish stochastic dominance relations on critical random quantities, such as the total number of customers in the system at any time instant and the number of customers that are rejected. We have based our proofs on forward event-driven inductions by coupling arrival and service times between systems that employ different routing mechanisms or have different configurations (specified by different allocation schemes). In particular, we have shown that the Shortest Non-Full Queue (SNQ) policy is optimal for a wide variety of performance measures, such as throughput and delay.

It is noteworthy that the SNQ policy minimizes the expected number of jobs that are present in the system at any time instant t, while it also maximizes the total number of customers that are admitted in the system by t. This is an unusual phenomenon in finite capacity systems, where trade-offs between metrics such as throughput and delay often appear. Moreover, the SNQ strategy presents another unique characteristic, namely, it is both individually and socially optimal. In the former sense, it is equally fair to all arriving customers; in the latter, it outperforms any other policy with respect to an overall cost index.

The problem of allocating a fixed number of buffers to the different queues has been also analyzed. Using a similar sample-path-dominance framework we have shown that the optimal allocation scheme is the one in which the difference between the maximum and minimum queue capacities is minimized, i.e., becomes either 0 or 1.

Last, it is of practical interest to study systems in which the queue lengths are not observed. In this case a static (as opposed to dynamic) routing policy should be determined prior to the system's initiation time and be employed thereafter. A simple intuitive argument is enough to reveal a very interesting trade-off between minimizing the total number of customers in the system and minimizing the total number of customers that are rejected. For instance, consider a system that operates under heavy load conditions, i.e., $\lambda > K\mu$. Then under a Round Robin (RR) policy that alternates between queues there are more customers present in the system than under a policy that routes all incoming customers to a single station. Clearly, however, the second policy rejects more customers out of the system. Using a different framework from the one presented in section 2 we have proved that the RR policy is optimal with respect to throughput. Results will be reported elsewhere.

References

[1] S.A.Banawan and J.Zahorjan, 'Load sharing in heterogeneous queueing systems', *Proc. IEEE Infocom'89 Conf.*, pp. 731-739, 1989.

[2] C.E.Bell and S.Stidham, 'Individual and social optimization in the allocation of customers to alternative servers', *Management Science*, vol. 29, pp. 831-839, 1983.

[3] D.P.Bertsekas, *Dynamic Programming*, Prentice Hall, chapter 5, 1987.

[4] C.Buyukkoc, P.Varaiya and J.Warland, 'The $c\mu$ rule revisited', *Advances on Applied Prob.*, vol. 17, pp. 237-238, 1985.

[5] Y.C.Chow and W.H.Kohler, 'Models for dynamic load balancing in a heterogeneous multiple processor system', *IEEE Trans. on Computers*, vol. C-28, pp. 354-361, 1979.

[6] D.R.Cox and V.Isham, *Point processes*, Chapman and Hall, London, 1980.

[7] A.Ephremides, P.Varaiya and J.Walrand, 'A simple dynamic routing problem', *IEEE Trans. on Aut. Control*, vol. AC-25, 1980.

[8] B.Hajek, 'Optimal control of two interactive service stations', *IEEE Trans. on Aut. Control*, vol. AC-29, pp. 491-498, 1984.

[9] Y.C.Ho, M.A.Eyler and T.T.Chien, 'A gradient technique for general buffer storage design in a production line', *Int. J. Production Res.*, vol. 17, pp. 557-580, 1979.

[10] M.A.Jafari and J.G.Shanthikumar, 'Determination of optimal buffer capacities and optimal allocation in multistage automatic transfer lines', to appear in *IEE Transactions.*

[11] P.K.Johri, 'Optimality of the shortest line discipline with state-dependent service times', *European J. of Operational Research*, vol. 41, pp. 157-161, 1989.

[12] K.R.Krishnan, 'Joining the right queue: A state-dependent decision rule', *IEEE Trans. on Aut. Control*, vol. 35, pp.104-108, 1990.

[13] W.Lin and P.R.Kumar, 'Optimal control of a queueing system with two heterogeneous servers', *IEEE Trans. on Aut. Control*, vol. AC-29, pp.696-703, 1984.

[14] A.W. Marshall and I. Olkin, *Inequalities: Theory of Majorization and Its Applications*, Academic Press, 1979.

[15] S.M.Ross, *Stochastic processes*, John Wiley and Sons, 1983.

[16] T.J.Sheskin, 'Allocation of interstage storage along an automatic transfer line' *AIEE Transactions*, vol. 8, pp. 146-152, 1976.

[17] P.D.Sparaggis and W.Gong, 'A generalized semi-Markov process model for the buffer allocation problem in cyclic networks', submitted for publication, 1990.

[18] A.Tanenbaum, *Computer networks*, Prentice Hall, 1988.

[19] D. Towsley, S. Fdida, H. Santoso, "Design and evaluation of flow control protocols for Metropolitan Area Networks", to appear in *Proceedings of NATO Workshop on High Speed Networks*, Sophia-Antipolis, France, June 1990.

[20] J.Walrand, *An introduction to queueing networks*, Prentice Hall, 1988.

[21] R.R.Weber, 'On the optimal assignment of customers to parallel queue', *J. of Applied Prob.*, vol. 15, pp. 406-413, 1978.

[22] W.Whitt, 'Deciding which queue to join', *Oper. Res.*, vol. 34, pp. 55-62, 1986.

[23] W.Winston, 'Optimality of the shortest line discipline', *J. of Applied Prob.*, vol. 14, pp. 181-189, 1977.