Probability in the Engineering and Informational Sciences, **16**, 2002, 185–203. Printed in the U.S.A.

ON PARALLEL QUEUING WITH RANDOM SERVER CONNECTIVITY AND ROUTING CONSTRAINTS

NICHOLAS BAMBOS
Departments of Electrical Engineering and Management Science & Engineering
Stanford University
Stanford, CA
E-mail: bambos@leland.stanford.edu

GEORGE MICHAILIDIS
Department of Statistics
The University of Michigan
Ann Arbor, MI
E-mail: gmichail@umich.edu

We study systems of parallel queues with finite buffers, a single server with *random connectivity* to each queue, and arriving job flows with *random* or *class-dependent accessibility* to the queues. Only currently connected queues may receive (preemptive) service at any given time, whereas an arriving job can only join one of its accessible queues. Using the *coupling method*, we study three key models, progressively building from simpler to more complicated structures.

In the first model, there are only random server connectivities. It is shown that allocating the server to the Connected queue with the Fewest Empty Spaces (C-FES) stochastically minimizes the number of lost jobs due to buffer overflows, under conditions of independence and symmetry.

In the second model, we additionally consider random accessibility of queues by arriving jobs. It is shown that allocating the server to the C-FES and routing each arriving job to the currently Accessible queue with the Most Empty Spaces (C-FES/A-MES) minimizes the loss flow stochastically, under similar assumptions.

In the third model (addressing a target application), we consider *multiple classes* of arriving job flows, each allowed access to a deterministic subset of the queues. Under analogous assumptions, it is again shown that the C-FES/A-MES policy minimizes the loss flow stochastically.

© 2002 Cambridge University Press 0269-9648/02 $12.50

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

The random connectivity/accessibility aspect enhances significantly the structure and application scope of the classical parallel queuing model. On the other hand, it introduces essential additional dynamics and considerable complications. It is interesting that a simple policy like FES/MES, known to be optimal for the classical model, extends to the C-FES/A-MES in our case.

1. INTRODUCTION

We address certain novel issues of resource allocation in *random environments*, which pertain to systems of parallel queues with finite buffers and a server with *random connectivity* to each queue. We also investigate job routing issues under *random/class-dependent accessibility* constraints to the queues. Such issues arise in several application areas, like wireless communication networks with extraneous interference, flexible manufacturing systems with failing components, and so forth. We analyze three key models capturing the essential features of various application scenarios.

The *first model* we consider is the following. There are K parallel first-come first-served (FCFS) queues, each having a positive finite buffer of capacity B. All queues are served by a single server, whose connectivity to each queue is randomly modulated (see Sect. 2). Only connected ones may be served. The service is *preemptive*, which implies that the job currently receiving service can be suspended and the server could be allocated to another queue. Jobs arriving to a queue and finding its buffer full are blocked and rejected; otherwise, they join the queue for service. The problem is how to dynamically allocate the server to the *connected* queues so as to stochastically minimize the flow of jobs that are lost due to buffer overflows. We provide a couple of key applications of this model.

In intelligent cellular communication networks, downlink messages destined to different mobiles are stored in separate buffers at the base station. The downlinks (base-to-mobile) are stochastically enabled/disabled, due to extraneous random interference. A transmitter has to decide which receiver (mobile) to transmit to, among the connected (enabled downlink) ones, in order to minimize buffer overflows and maximize the aggregate throughput. Note that the notion of enabled/disabled service is analogous to that of connectivity and such systems can be cast into our modeling framework.

In a flexible manufacturing system, the operator of a workstation uses several tools to execute various jobs, which arrive at random times and are queued up in separate finite buffers, according to the tools required for their execution. Jobs have random service times. The tools fail and are repaired at random times. The jobs of a particular buffer cannot be executed if any of the required tools are not operational. Hence, to maximize the workstation throughput, the operator must decide which queue to serve, among those for which all required tools are operational.

The *second model* we study has the same buffer and service structure as the first model; however, there is now a single job arrival stream with randomly modulated accessibility to the queues (see Sect. 4). An arriving job must join one of the cur-

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

rently connected queues at its arrival time, assuming there is a nonfull one; otherwise, it is blocked and rejected. The problem is to jointly allocate the server among connected queues and route incoming jobs to accessible buffers, so as to minimize the job loss flow.

Such is the situation in satellite networks, where the transmitting Earth station uploads (routes) messages to the satellites it can access in a given time period while the receiving Earth station downloads (serves) messages from the satellites to which it can connect, as random interference permits.

The *third model* we analyze also has the same buffer and service structure as the first model. In this case, however, there are multiple classes of jobs. Each class is characterized by the subset of queues to which its jobs may be routed (see Sect. 5). Therefore, we have *class-dependent* deterministic accessibility (routing) constraints. The problem now is to route each arriving job to a queue permitted by its class and allocate the server to a connected queue in order to minimize buffer overflows.

An application of this model (which has primarily motivated this work) is in the area of selective transmission in multichannel wireless communication. Suppose that there are K communication channels and a tunable transmitter that can transmit messages in any of them. Each channel has a separate finite buffer, where messages to be transmitted in the channel are queued up. There are several sources of messages (users of the communication system), each of which may only transmit in a subset of the channels (those to which it has leased access). Each source corresponds to a class of messages. At any given time, the transmitter can serve only those channels (queues) whose interference is below some acceptable threshold, in order to guarantee some quality of service. These active channels change in time due to random extraneous interference patterns. Each user chooses the channel buffers to place its messages in, and the transmitter chooses the channel to serve, among the enabled ones. The objective is to maximize the aggregate throughput.

The classical parallel queuing problem with job routing and/or server scheduling has received considerable attention in the literature, being a key queuing paradigm. It has been shown for both finite and infinite buffers that "routing an incoming job to the shortest queue" (RSQ) and "assigning a server to the longest queue" (SLQ) is optimal under various assumptions (see [9,11] and references therein). In [6], the *joint* problem of routing incoming jobs and scheduling a server has been considered and the optimality of the RSQ/SLQ policy proved, under state-dependent service rates. Typically, dynamic programming arguments and/or stochastic comparison methods have been used for establishing results in this area, mostly under assumptions of statistical symmetry associated with the systems. Random queue–server connectivities have been considered in [10], where the stability problem of infinite buffer queues has been addressed in a discrete-time Markovian framework. The results have been extended in [1] to general stationary ergodic arrival and connectivity processes. In [4], the optimality of index policies is investigated for a model of parallel queues and many servers with random server connectivities. Finally, in [12], the connectivity mechanism (on or off) is generalized to a multistate one and the stability properties of the system are investigated.

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

The introduction of random server connectivity and random/class-dependent queue accessibility provides a significant conceptual/structural extension of the standard parallel queuing model (and widens its application scope considerably) in a useful direction that has been little explored in the past. On the other hand, this extension generates additional essential dynamics which must be considered.

The organization of the article is as follows. In Section 2, the first model (baseline) is precisely defined. It is then shown in Section 3 that allocating the server to the currently Connected queue with the Fewest Empty buffer Spaces (C-FES) stochastically minimizes the loss flow, under *symmetric* memoryless arrivals, service and connectivities, equal-size buffers, and preemptive service. In Section 4, the second model is addressed. It is shown that allocating the server to the currently C-FES and routing each arriving job to the currently Accessible queue with the Most Empty Spaces (C-FES/A-MES) minimizes the loss flow in *symmetric* Markovian systems. In Section 5, the third model is addressed. The C-FES/A-MES policy is again shown to be optimal under assumptions of symmetry. In Section 6, selected simulation results are presented, indicating that the system performance is considerably robust with respect to buffer sizes and structural asymmetry. Finally, several current attempts at generalizing the results are discussed in Section 7 and some open problems identified.

To show the results, we use the coupling method [3] and the standard structural ordering provided by the submajorization framework [5,8,9]. Building on this basis, the article contributes in the following two directions. First, it extends substantially the standard parallel queuing paradigm by introducing the powerful modeling attribute of randomly modulated connectivity/accessibility. Second, in the technical direction, the main issue becomes the careful pathwise comparison of alternative policies under various coupling structures. The random connectivity/accessibility introduces two additional layers in the coupling construction, compared to the standard parallel queuing model. Moreover, in the third model, a novel type of coupling construction has to be used. Indeed, the matching has to be done at the level of *sets* of queues rather than between individual queues, as earlier. The emerging complications are then successfully resolved, and the method seems to have wider applicability to problems in this area.

The first model is a special case of the third; however, it is leveraged in the analysis of both the second and third. In the interest of keeping the proofs short, when arguments analogous to those used in previous proofs arise again, they are not repeated but simply referred to with minimal explanations. Analyzing the models in the order introduced allows us to use the above strategy most effectively. We gradually build up the investigation toward the third (target) model, at each step simply addressing the novel problems arising.

2. PARALLEL QUEUES WITH RANDOM SERVER CONNECTIVITIES

Let us now precisely define our first model. The job arrival process to the kth queue is a *Poisson* flow $\mathbf{A}_k = \{A_k(t); t \in \mathcal{R}_+\}$, $k = 1, 2, \ldots, K$, where $A_k(t)$ is the number of

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

attempted arrivals to the kth queue in the time interval $(0, t]$. The Poisson processes $\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_K$ are mutually independent and have equal rates (*symmetric arrivals*). The service times of all jobs (in any queue of the system) are independent and identically distributed (i.i.d.) *exponential* random variables (*symmetric service*). Each server–queue connectivity process $\mathbf{C}_k = \{C_k(t); t \in \mathcal{R}_+\}$, $k = 1, 2, \ldots, K$, is modeled as a two-state continuous-time Markov chain, where $C_k(t)$ is 1 if the server is connected to the kth queue at time t, and 0 otherwise. The connectivity processes are mutually independent and have identical statistics (*symmetric connectivities*). It is assumed that all connectivity processes are in stationarity (in any case, they do couple with their stationary versions in finite time). All random processes and variables are defined on a common probability space $(\Omega, \mathcal{F}, \mathcal{P})$. The arrival flows, service times, and connectivity processes are assumed to be mutually independent.

A *server allocation policy* is used to decide which queue to serve among the currently connected ones. Decision instants are the service completion, connectivity switching, and job arrival epochs. Service is *preemptive* (actually, this is the only possibility when a queue receiving service suddenly becomes disconnected). Let Γ be the set of all allocation policies which base their decisions only on current and past system state information. We define $\mathbf{Q}^\gamma(t) = \{Q_1^\gamma(t), Q_2^\gamma(t), \ldots, Q_K^\gamma(t); t \geq 0\}$ to be the joint queue length process, where $Q_k^\gamma \in \{1, 2, \ldots, B\}$ is the number of jobs in the kth queue at time t (including that in service), under the server allocation policy $\gamma \in \Gamma$. We also define the residual capacity process $\mathbf{R}^\gamma(t) = \{R_1^\gamma(t), R_2^\gamma(t), \ldots, R_K^\gamma(t); t \geq 0\}$, where $R_k^\gamma = B - Q_k^\gamma(t)$ is the number of empty buffer spaces in the kth queue. Finally, we define the loss process $\mathbf{L}^\gamma = \{L^\gamma(t); t \geq 0\}$ and the departure process $\mathbf{D}^\gamma = \{D^\gamma(t); t \geq 0\}$, where $L^\gamma(t)$ and $D^\gamma(t)$ denote the number of lost jobs (due to buffer overflows) and the number of departing jobs (after receiving service), respectively, in the time interval $(0, t]$, under the server allocation policy γ.

Denote by $\pi \in \Gamma$ the policy which at every decision epoch allocates the server to the connected queue with the fewest number of empty buffer spaces. We call it *Connected Fewest Empty Spaces* policy or C-FES. Ties (equal number of empty spaces in two or more connected queues) are arbitrarily broken; for example, giving priority to higher index queue or by choosing any connected queue equiprobably. Note that C-FES allocates the server based only on current system state (stationary policy). It is shown (in Sect. 3) to stochastically minimize the loss flow or equivalently maximize the departure flow (throughput).

3. LOSS FLOW MINIMIZATION UNDER C-FES

We briefly review the definition and key properties of *weak submajorization*, which is used later in the proofs. We mainly follow the discussion in [9], where the statements of Fact 3.1 are proved. Given a K-dimensional vector $\mathbf{G} = (G_1, G_2, \ldots, G_k, \ldots, G_K) \in \mathcal{R}^K$, define $G_{(i)}$ to be the ith largest component of $\mathbf{G}$ and assign *order index* i to it (ties are arbitrarily broken; e.g., according to the actual indices of the vector components); that is, if i is the order index of G_k, then $(i) = k$, which is the *actual index* of the component. Denote by $\mathbf{G}_{(\cdot)} = (G_1, G_2, \ldots, G_{(k)}, \ldots, G_{(K)})$ the *ordered ver-*

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

sion of the vector $\mathbf{G}$. For example, for $\mathbf{G} = (G_1, G_2, G_3) = (7,10,2)$, we have $\mathbf{G}_{(\cdot)} = (G_{(1)}, G_{(2)}, G_{(3)}) = (10,7,2)$; hence, $(1) = 2$, $(2) = 1$, and $(3) = 3$.

We say that a K-dimensional vector $\mathbf{H} \in \mathcal{R}^K$ *weakly submajorizes* a K-dimensional vector $\mathbf{G} \in \mathcal{R}^K$ and write $\mathbf{G} \prec_w \mathbf{H}$, iff $\sum_{i=1}^{l} G_{(i)} \leq \sum_{i=1}^{l} H_{(i)}$ for every $l = 1,2,\ldots,K$, where $G_{(i)}$ ($H_{(i)}$) is the ith *largest* component of $\mathbf{G}$ ($\mathbf{H}$), as previously defined.

Fact 3.1 (Properties of Weak Submajorization): If $\mathbf{G} \prec_w \mathbf{H}$, then for $i,j \in \{1,2,\ldots,K\}$, the following hold:

1. $\{G_{(1)},\ldots,(G_{(i)} - 1)^+,\ldots,G_{(K)}\} \prec_w \{H_{(1)},\ldots,(H_{(j)} - 1)^+,\ldots,H_{(K)}\}$, $i \leq j$, $x^+ = \max(x,0)$.
2. $\{G_{(1)},\ldots,(G_{(i)} + 1),\ldots,G_{(K)}\} \prec_w \{H_{(1)},\ldots,(H_{(j)} + 1),\ldots,H_{(K)}\}$, $i \geq j$.
3. $\{G_{(1)},\ldots,G_{(i)},\ldots,G_{(K)}\} \prec_w \{H_{(1)},\ldots,(H_{(j)} - 1)^+,\ldots,H_{(K)}\}$ if $\sum_{i=1}^{k} G_{(i)} < \sum_{i=1}^{k} H_{(i)}$, for every $k \geq j$, with $k, j \in \{1,2,\ldots,K\}$.

We employ the following definition of the stochastic ordering $\mathbf{X} \leq_{\text{st}} \mathbf{Y}$ of random processes [7], writing $\{X(t); t \geq 0\} \leq_{\text{st}} \{Y(t); t \geq 0\}$ if $E[f(X(t_1), X(t_2), \ldots, X(t_n))] \leq E[f(Y(t_1), Y(t_2), \ldots, Y(t_n))]$, for all $n \in \mathcal{Z}_+$, $t_1,\ldots,t_n \in \mathcal{R}_+$ and all bounded, nondecreasing functions $f: \mathcal{R}^n \to \mathcal{R}$.

Proposition 3.1 (Optimality of the C-FES Policy): *The C-FES policy $\pi \in \Gamma$ stochastically minimizes the loss flow and maximizes the throughput; that is,*

$$\{\mathbf{L}^{\pi}(t); t \geq 0\} \leq_{\text{st}} \{\mathbf{L}^{\gamma}(t); t \geq 0\}, \tag{1}$$

$$\{\mathbf{D}^{\pi}(t); t \geq 0\} \geq_{\text{st}} \{\mathbf{D}^{\gamma}(t); t \geq 0\} \tag{2}$$

for all policies $\gamma \in \Gamma$, given that $\mathbf{Q}^{\pi}(0) = \mathbf{Q}^{\gamma}(0)$, $\mathbf{D}^{\pi}(0) = \mathbf{D}^{\gamma}(0) = 0$, and $\mathbf{L}^{\pi}(0) = \mathbf{L}^{\gamma}(0) = 0$.

Proof: For an arbitrary policy $\gamma \in \Gamma$, we simply need to construct couplings $(\hat{\mathbf{L}}^{\pi}, \hat{\mathbf{L}}^{\gamma})$ and $(\hat{\mathbf{D}}^{\pi}, \hat{\mathbf{D}}^{\gamma})$ of the random processes $(\mathbf{L}^{\pi}, \mathbf{L}^{\gamma})$ and $(\mathbf{D}^{\pi}, \mathbf{D}^{\gamma})$, respectively, such that $\hat{\mathbf{L}}^{\pi} \stackrel{\text{d}}{=} \mathbf{L}^{\pi}$, $\hat{\mathbf{L}}^{\gamma} \stackrel{\text{d}}{=} \mathbf{L}^{\gamma}$ and $\hat{L}^{\pi}(t) \leq \hat{L}^{\gamma}(t)$ a.s. for $t \geq 0$, and $\hat{\mathbf{D}}^{\pi} \stackrel{\text{d}}{=} \mathbf{D}^{\pi}$, $\hat{\mathbf{D}}^{\gamma} \stackrel{\text{d}}{=} \mathbf{D}^{\gamma}$ and $\hat{D}^{\pi}(t) \geq \hat{D}^{\gamma}(t)$ a.s. for $t \geq 0$. The "hat" symbol denotes the coupled versions of the processes, and $\stackrel{\text{d}}{=}$ denotes the equality of their finite-dimensional distributions. Relations (1) and (2) then follow immediately [2,3]. Denote by $\mathcal{S}^{\gamma}$ ($\mathcal{S}^{\pi}$) the system operating under policy γ (π). Let $\sigma_n^{\gamma} \in \mathcal{R}_+$ ($\sigma_n^{\pi} \in \mathcal{R}_+$) be the service time of the nth job to complete service in $\mathcal{S}^{\gamma}$ ($\mathcal{S}^{\pi}$).

The simplest coupling is one which leverages directly on weak submajorization, tracking the systems $\mathcal{S}^{\gamma}$ and $\mathcal{S}^{\pi}$ at all decision instants (arrival times, service completions, and connectivity switchings). We construct new systems $\hat{\mathcal{S}}^{\gamma}$ and $\hat{\mathcal{S}}^{\pi}$ whose dynamics are probabilistically coupled to those of $\mathcal{S}^{\gamma}$ and $\mathcal{S}^{\pi}$ correspondingly, on the common probability space, as follows. Letting $\hat{\sigma}_n^{\gamma} \in \mathcal{R}_+$ ($\hat{\sigma}_n^{\pi} \in \mathcal{R}_+$) be the service time of the nth job ($n \in \mathcal{Z}_+$) to complete service in $\hat{\mathcal{S}}^{\gamma}$ ($\hat{\mathcal{S}}^{\pi}$), set $\hat{\sigma}_n^{\pi} = \hat{\sigma}_n^{\gamma}$ (since the service times are i.i.d. random variables with identical statistics in all queues). If all currently connected queues in $\hat{\mathcal{S}}^{\gamma}$ ($\hat{\mathcal{S}}^{\pi}$) are empty, the server

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

engages a fictitious job of identical statistics, whose service time does get registered in the sequence $\{\hat{\sigma}_n^{\gamma}\}$ ($\{\hat{\sigma}_n^{\pi}\}$); however, departures of fictitious jobs are not registered in $\hat{\mathbf{D}}^{\gamma}$ ($\hat{\mathbf{D}}^{\pi}$). This does not affect the system statistics (due to the preemptiveness of service), but it preserves the synchronization of the indices of the two sequences which is used later. Now, at every time $t \in \mathcal{R}_+$, set (match) the arrival and connectivity processes of the queue having the ith largest residual capacity ($i \in \{1,2,\ldots,K\}$) in $\hat{\mathcal{S}}^{\gamma}$ with the arrival and connectivity processes (correspondingly) of the ith largest residual capacity queue in $\hat{\mathcal{S}}^{\pi}$. Hence, the ith largest residual capacity queue in $\hat{\mathcal{S}}^{\gamma}$ receives jobs and switches connectivities at the same times as that in $\hat{\mathcal{S}}^{\pi}$. Ordering the queues at decision instants according to their empty buffer spaces and matching the arrival and connectivities of the ith largest residual capacity ones under the two policies does not alter the statistics of $\mathcal{S}^{\gamma}$ and $\mathcal{S}^{\pi}$, since the interepoch times of the arrival, service, and connectivity processes are i.i.d exponential (memoryless) random variables with identical (symmetric) statistics. This is the *intuition* behind the coupling of $\hat{\mathcal{S}}^{\gamma}$ with $\mathcal{S}^{\gamma}$, and $\hat{\mathcal{S}}^{\pi}$ with $\mathcal{S}^{\pi}$.

The dynamics of the systems $\hat{\mathcal{S}}^{\gamma}$ and $\hat{\mathcal{S}}^{\pi}$ induce the required couplings $\hat{\mathbf{L}}$ and $\hat{\mathbf{D}}$. Indeed, let $\hat{\mathbf{R}}^{\gamma}(t)$ ($\hat{\mathbf{R}}^{\pi}(t)$) be the residual capacity process of $\hat{\mathcal{S}}^{\gamma}$ ($\hat{\mathcal{S}}^{\pi}$). We assume that $\hat{\mathbf{R}}^{\pi}(0) = \hat{\mathbf{R}}^{\gamma}(0)$. We simply need to show that for the coupled versions of the processes, we have pathwise

$$\hat{L}^{\pi}(t) \leq \hat{L}^{\gamma}(t), \tag{3}$$

$$\hat{D}^{\pi}(t) \geq \hat{D}^{\gamma}(t), \tag{4}$$

and

$$\hat{\mathbf{R}}^{\pi}(t) \prec_w \hat{\mathbf{R}}^{\gamma}(t) \tag{5}$$

for all $t \in \mathcal{R}_+$. The proof proceeds by induction on decision epochs (i.e., job arrival, service completion, and connectivity switching times $t_0 = 0 < t_1 < t_2 < \cdots < t_n < t_{n+1} \cdots$ in $\hat{\mathcal{S}}^{\gamma}$ and $\hat{\mathcal{S}}^{\pi}$). Note that simultaneous events in the same system can occur only with probability zero. From $\hat{\mathbf{R}}^{\pi}(0) = \hat{\mathbf{R}}^{\gamma}(0)$, relations (3)–(5) follow immediately for $t = t_0 = 0$. Assume that all three relations hold at $t = t_n$ (the *induction hypothesis*). Clearly, they also hold for all $t \in [t_n, t_{n+1})$. We will show that they continue to hold at $t = t_{n+1}$, considering the following cases:

1. *Connectivity switchings*. Suppose that at t_{n+1} there is a connectivity switching at some queue. Due to the preemptiveness of service, there will be no change in the queue lengths in $\hat{\mathcal{S}}^{\gamma}$ or $\hat{\mathcal{S}}^{\pi}$; hence, (3)–(5) trivially hold at $t = t_{n+1}$.
2. *Service completions*. Consider now the situation where a service completion occurs at $t = t_{n+1}$. Relation (3) obviously holds (no arrivals occur). For the other two, we examine the following four subcases:
   a. Suppose that there is some nonempty queue in $\hat{\mathcal{S}}^{\gamma}$ and some in $\hat{\mathcal{S}}^{\pi}$ at $t = t_n$. Then, using the induction hypothesis, we easily obtain $\hat{D}^{\pi}(t_{n+1}) = \hat{D}^{\pi}(t_n) + 1 \geq \hat{D}^{\gamma}(t_n) + 1 = \hat{D}^{\gamma}(t_{n+1})$, since service times are coupled in

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

$\hat{\mathcal{S}}^\gamma$ and $\hat{\mathcal{S}}^\pi$. Recall that *fictitious* jobs, which have possibly been served in the past (when all connected queues were empty), have preserved the synchronization of the indices in $\{\hat{\sigma}_n^\gamma\}$ and $\{\hat{\sigma}_n^\pi\}$, resulting in currently matched values. Regarding the residual capacities, note that, in $\hat{\mathcal{S}}^\pi$, the queue served has the largest order-index, say i, among the currently connected ones (because C-FES serves the connected queue with the fewest empty spaces). Since the connectivities in $\hat{\mathcal{S}}^\pi$ are also matched with those in $\hat{\mathcal{S}}^\gamma$, we see that the order-index, say j, of the queue served in $\hat{\mathcal{S}}^\gamma$ is such that $i \geq j$. Therefore, from Fact 3.1(2), we get $\hat{\mathbf{R}}^\pi(t_{n+1}) \prec_w \hat{\mathbf{R}}^\gamma(t_{n+1})$, using the induction hypothesis and noting that, at t_{n+1}, both $\hat{\mathcal{S}}^\gamma$ and $\hat{\mathcal{S}}^\pi$ see their residual capacities increasing by 1 at queues (i) and (j) correspondingly.

b. Suppose now that in $\hat{\mathcal{S}}^\pi$ all connected queues are empty, whereas in $\hat{\mathcal{S}}^\gamma$, there is a nonempty one at $t = t_n$. Let i_*^π be the largest order-index among those of connected queues in $\hat{\mathbf{R}}^\pi(t_n)$. Since all such queues have residual capacity B (being empty in $\hat{\mathcal{S}}^\pi$) and due to the decreasing sizes of $\hat{R}_{(i)}^\pi(t_n)$ as i increases, we see that $\hat{R}_{(i)}^\pi(t_n) = B$ for every queue $i \in \{1,2,3,\ldots,i_*^\pi\}$. However, due to the matching of connectivities in $\hat{\mathcal{S}}^\pi$ and $\hat{\mathcal{S}}^\gamma$ and the fact that, in $\hat{\mathcal{S}}^\gamma$, there is a nonempty connected queue, there must be some queue of order-index i_0^γ such that $i_0^\gamma \in \{1,2,\ldots,i_*^\pi\}$ and $\hat{R}_{(i_0^\gamma)}^\gamma(t_n) < B$. Hence, $\sum_{i=1}^{i_*^\pi} \hat{R}_{(i)}^\pi(t_n) = i_*^\pi B > \sum_{i=1}^{i_*^\pi} \hat{R}_{(i)}^\gamma(t_n)$, which contradicts the induction hypothesis $\hat{\mathbf{R}}^\pi(t_n) \prec_w \hat{\mathbf{R}}^\gamma(t_n)$, rendering this case *impossible*.

c. If, in $\hat{\mathcal{S}}^\gamma$, all connected queues are empty and there is a nonempty one in $\hat{\mathcal{S}}^\pi$ at $t = t_n$, we immediately get $\hat{D}^\pi(t_{n+1}) = \hat{D}^\pi(t_n) + 1 > \hat{D}^\gamma(t_n)$, proving (4). Concerning the residual capacities, define k_*^γ and k_0^π analogously to i_*^π and i_0^γ in Step 2b (correspondingly), reversing the previous argument. We then get $\sum_{i=1}^{k_*^\gamma} \hat{R}_{(i)}^\gamma(t_n) = k_*^\gamma B$ and $\hat{R}_{(k_0^\pi)}^\pi(t_n) < B$, whereas $\hat{R}_{(i)}^\pi(t_n) \leq B$ for all $i \in \{1,2,3,\ldots,K\}$. These facts, combined with the induction hypothesis, immediately give $\hat{\mathbf{R}}^\pi(t_{n+1}) \prec_w \hat{\mathbf{R}}^\gamma(t_{n+1})$.

d. If all connected queues in both $\hat{\mathcal{S}}^\gamma$ and $\hat{\mathcal{S}}^\pi$ are empty at $t = t_n$, the proof is trivial.

3. *Job arrivals*. Let now t_{n+1} be the arrival time of a job in both $\hat{\mathcal{S}}^\gamma$ and $\hat{\mathcal{S}}^\pi$. Recall that arrivals are matched in the ith largest residual capacity queues in the two systems. Relation (4) is trivially true (no departures occur). To show (5), we consider the following four subcases:

a. If the job is admitted in both $\hat{\mathcal{S}}^\pi$ and $\hat{\mathcal{S}}^\gamma$, then $\hat{\mathbf{R}}^\pi(t_{n+1}) \prec_w \hat{\mathbf{R}}^\gamma(t_{n+1})$ from Fact 3.1(1), due to $\hat{\mathbf{R}}^\pi(t_n) \prec_w \hat{\mathbf{R}}^\gamma(t_n)$ and the matching of arrivals.

b. If the job is admitted in $\hat{\mathcal{S}}^\pi$, but rejected in $\hat{\mathcal{S}}^\gamma$, then $\hat{\mathbf{R}}^\gamma(t_{n+1}) = \hat{\mathbf{R}}^\gamma(t_n)$. Using $\hat{\mathbf{R}}^\pi(t_n) \prec_w \hat{\mathbf{R}}^\gamma(t_n)$ and Fact 3.1(1), we immediately get $\hat{\mathbf{R}}^\pi(t_{n+1}) \prec_w \hat{\mathbf{R}}^\gamma(t_{n+1})$.

c. If the job is rejected in $\hat{\mathcal{S}}^\pi$, but admitted in $\hat{\mathcal{S}}^\gamma$, then $\hat{\mathbf{R}}^\pi(t_{n+1}) = \hat{\mathbf{R}}^\pi(t_n)$. Let i_a be the order-index of the queue that the arriving job joins in $\hat{\mathcal{S}}^\gamma$. Obviously, $R_{(i_a)}^\gamma(t_n) > 0$. Since the job was rejected in $\hat{\mathcal{S}}^\pi$ and due to the matching of arrivals in the ith largest queues in $\hat{\mathcal{S}}^\pi$ and $\hat{\mathcal{S}}^\gamma$ and the de-

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

creasing sizes of $\hat{R}^{\pi}_{(i)}(t_n)$ as i increases, we must have $\hat{R}^{\pi}_{(i)}(t_n) = 0$ for every i such that $i \in \{i_a, i_a + 1, \ldots K\}$. However, from $\hat{\mathbf{R}}^{\pi}(t_n) \prec_w \hat{\mathbf{R}}^{\gamma}(t_n)$, we have $\sum_{i=1}^{k} \hat{R}^{\pi}_{(i)}(t_n) \leq \sum_{i=1}^{k} \hat{R}^{\gamma}_{(i)}(t_n)$ for every $k \in \{1,2,\ldots,K\}$. Combining the above three facts, we get $\sum_{i=1}^{k} \hat{R}^{\pi}_{(i)}(t_n) < \sum_{i=1}^{k} \hat{R}^{\gamma}_{(i)}(t_n)$, for every $k \in \{i_a, i_a + 1, \ldots, K\}$. Using Fact 3.1(3), we immediately get $\hat{\mathbf{R}}^{\pi}(t_{n+1}) \prec_w \hat{\mathbf{R}}^{\gamma}(t_{n+1})$.

d. If the job is rejected in both $\hat{\mathcal{S}}^{\pi}$ and $\hat{\mathcal{S}}^{\gamma}$, the proof is trivial.

The above arguments show that $\hat{D}^{\pi}(t) \geq \hat{D}^{\gamma}(t)$ and $\hat{\mathbf{R}}^{\pi}(t) \prec_w \hat{\mathbf{R}}^{\gamma}(t)$ for all $t \geq 0$ pathwise. However, they do not fully cover (actually in case 3c) the proof for the loss process (3). Instead, the following arguments need to be used.

Note that relation (5), which has already been proven, implies that $\sum_{k=1}^{K} \hat{Q}^{\pi}_{k}(t) \geq \sum_{k=1}^{K} \hat{Q}^{\gamma}_{k}(t) \geq 0$. Moreover, the following structural relations hold:

$$\sum_{k=1}^{K} \hat{Q}^{\pi}_{k}(0) + \sum_{k=1}^{K} \hat{A}^{\pi}_{k}(t) = \hat{D}^{\pi}(t) + \hat{L}^{\pi}(t) + \sum_{k=1}^{K} \hat{Q}^{\pi}_{k}(t) \tag{6}$$

and

$$\sum_{k=1}^{K} \hat{Q}^{\gamma}_{k}(0) + \sum_{k=1}^{K} \hat{A}^{\gamma}_{k}(t) = \hat{D}^{\gamma}(t) + \hat{L}^{\gamma}(t) + \sum_{k=1}^{K} \hat{Q}^{\gamma}_{k}(t). \tag{7}$$

Since both $\hat{\mathcal{S}}^{\gamma}$ and $\hat{\mathcal{S}}^{\pi}$ have matched arrivals, we have $\sum_{k=1}^{K} \hat{A}^{\pi}_{k}(t) = \sum_{k=1}^{K} \hat{A}^{\gamma}_{k}(t)$, so subtracting (7) from (6), we get

$$\hat{L}^{\gamma}(t) - \hat{L}^{\pi}(t) = (\hat{D}^{\pi}(t) - \hat{D}^{\gamma}(t)) + \left(\sum_{k=1}^{K} \hat{Q}^{\pi}_{k}(t) - \sum_{k=1}^{K} \hat{Q}^{\gamma}_{k}(t) \right) \tag{8}$$

for all $t \in \mathcal{R}_+$. Using the already proven relation (4), we immediately get relation (3). This completes the proof of the proposition. ■

Remark 3.1 (Discrete-Time Dynamics): A result analogous to that of Proposition 3.1 holds in the case where time is slotted (discrete) and the interarrival, service, and connectivity interswitching times are geometrically distributed. The proof of optimality of C-FES is basically the same, except that now we can have multiple arrivals, a service completion, and multiple connectivity switchings occurring simultaneously (in the same time slot) with positive probability (that cannot happen under continuous-time Markovian dynamics). To handle this additional complication, we need to introduce an ordering in registering simultaneous events; that is, we first register the departure, then the arrivals and connectivity switchings, and, finally, (at the end of the slot) the C-FES decides for the new allocation. The proof then proceeds along the lines of the continuous-time one.

Remark 3.2 (Other Couplings): There are other ways to construct a coupling to prove the result; we have considered a few of them. For example, we can couple the arrival and connectivity processes, as well as the service times, on the *actual* queues (instead of reordering them first, according to their residual capacities). We can then

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

check whether forcing the C-FES policy for a single decision instant on an arbitrary server allocation policy (interchange) leads to improvement almost surely. However, the simplest proof seems to result from the coupling employed in Proposition 3.1, in particular with respect to the extension of the result in the enhanced models analyzed in the following sections. Unfortunately, the symmetry of arrivals, service, and connectivities is of crucial importance for the proof under all the couplings we have considered, as is typically the case in proofs based on pathwise arguments.

Remark 3.3 (Asymmetric Systems): In the case of asymmetric arrival, service, and connectivity processes in the various queues, we can still formulate the optimal server allocation problem within the framework of dynamic programming. However, the situation becomes very complicated and it is not clear how to resolve it. Actually, even in the symmetric case, the dynamic programming approach leads to a more involved scheme. In the particular case of unequal buffers, the proof of Proposition 3.1 collapses at step 2b. In Section 6, we investigate asymmetric systems experimentally, and in Section 7, we discuss some related open problems.

4. RANDOM QUEUE ACCESSIBILITY AND JOB ROUTING

Let us now move on to the second model, which includes job routing. We consider a single stream of incoming jobs with randomly modulated access to the queues. Each job joins (is routed to) one of the queues that are accessible upon its arrival time, provided not all of them are full; otherwise, it is blocked and rejected. As before, there are K parallel FCFS queues with finite buffers of equal size B and a single server with randomly modulated server–queue connectivities. In addition to allocating the server, this model requires routing decisions for placing jobs in accessible queues, so as to minimize the loss flow.

Let $\mathbf{A} = \{A(t); t \in \mathcal{R}_+\}$ be the job arrival flow of the system, where $A(t)$ is the number of attempted arrivals in the time interval $(0, t]$. Let $O_k(t)$ take the value 1 if the kth queue is accessible by (open to) an incoming job at time $t \in \mathcal{R}_+$, and 0 otherwise. $\mathbf{O}_k = \{O_k(t); t \in \mathcal{R}_+\}$, $k \in \{1,2,3,\ldots,K\}$, is the *access process* of the kth queue. We model $\mathbf{O}_k$ as a two-state (1 and 0 or open and closed) continuous-time Markov chain. We assume that the access processes of all queues are independent with identical statistics (*symmetric access*). If all accessible queues are full upon the arrival of a job or if no queue is accessible, then that job is immediately blocked and rejected. The statistics of the job service times and the server–queue connectivity processes are as in Section 2 (symmetric service and connectivities).

In the present setting, we need a *joint server allocation/job routing policy* to decide which queue to serve, among those that are currently connected, and where to route an arriving job, among the queues that are currently accessible. Again, decision instants are the job arrival, service completion, and connectivity switching times, and service is preemptive. Let $\tilde{\Gamma}$ be the set of policies that base decisions on past and present state information. We are once again interested in characterizing an alloca-

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

tion policy which stochastically minimizes the loss process $\mathbf{L}$ and maximizes the departure process $\mathbf{D}$ (defined as in Sect. 2).

Denote by $\pi^* \in \tilde{\Gamma}$ the policy which at every decision instant allocates the server to the *connected* queue with the *fewest empty spaces* and routes an incoming job to the accessible queue with the *most empty spaces* (if there are any; otherwise, it rejects it). When arrivals occur, the routing decision (action) is completed before the server allocation one is made. Ties are broken based on some priority scheme or equiprobably, as in the C-FES policy. This policy uses only current state information (stationary). We call it C-FES/A-MES.

PROPOSITION 4.1 (Optimality of the C-FES/A-MES Policy): *The C-FES/A-MES policy $\pi^* \in \tilde{\Gamma}$ stochastically minimizes the loss flow and maximizes the throughput; that is,*

$$\{\mathbf{L}^{\pi}(t); t \geq 0\} \leq_{\text{st}} \{\mathbf{L}^{\gamma}(t); t \geq 0\}, \tag{9}$$

$$\{\mathbf{D}^{\pi}(t); t \geq 0\} \geq_{\text{st}} \{\mathbf{D}^{\gamma}(t); t \geq 0\} \tag{10}$$

for all policies $\gamma \in \tilde{\Gamma}$, given that $\mathbf{Q}^{\pi^*}(0) = \mathbf{Q}^{\gamma}(0)$, $\mathbf{D}^{\pi^*}(0) = \mathbf{D}^{\gamma}(0) = 0$, and $\mathbf{L}^{\pi^*}(0) = \mathbf{L}^{\gamma}(0) = 0$.

PROOF: The structure of the proof is analogous to that of Proposition 3.1. We fully borrow the notation from the former proof. We simply address here the points that are different, due to the change in the model structure.

We compare the evolution of the system under policies π^* (C-FES/A-MES) and γ in $\tilde{\Gamma}$. The following coupling needs to be used here. First, the service times are coupled as before ($\hat{\sigma}_n^{\pi^*} = \hat{\sigma}_n^{\gamma}$). Also, the server connectivities of the ith largest residual capacity queue under π^* is coupled with the server connectivities of the ith largest residual capacity queue under γ (for all i's). However, the coupling employed here additionally matches the job arrival times and the queue access processes in the ith largest residual capacity queues under policies π^* and γ correspondingly. The proof proceeds by induction on decision instants.

Steps 1 (connectivity switchings) and 2 (service completions) of the induction are identical to those in Proposition 3.1. Step 3 (job arrivals) obviously needs some additional analysis, addressing the enhancements in the model structure and the nature of the C-FES/A-MES policy. Steps 3b and 3d carry over to the present situation as they are.

Step 3a needs to be altered as follows. Suppose the job is admitted in both $\hat{\mathcal{S}}^{\pi^*}$ and $\hat{\mathcal{S}}^{\gamma}$. Note that in $\hat{\mathcal{S}}^{\pi^*}$, the queue where the job is routed to must have the smallest order-index, say i, among the currently accessible ones (because C-FES/A-MES routes it to the accessible queue with the most empty spaces). Since the queue accessibilities in $\hat{\mathcal{S}}^{\pi^*}$ are matched to those in $\hat{\mathcal{S}}^{\gamma}$, we see that the order-index, say j, of the queue where the job is routed to in $\hat{\mathcal{S}}^{\gamma}$ is such that $i \leq j$. Therefore, from Fact 3.1(1), we get $\hat{\mathbf{R}}^{\pi^*}(t_{n+1}) \prec_w \hat{\mathbf{R}}^{\gamma}(t_{n+1})$, using the induction hypothesis $\hat{\mathbf{R}}^{\pi^*}(t_n) \prec_w \hat{\mathbf{R}}^{\gamma}(t_n)$

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

and noting that at t_{n+1}, both $\hat{\mathcal{S}}^{\pi^*}$ and $\hat{\mathcal{S}}^{\gamma}$ see their residual capacities decrease by 1 at queues (i) and (j) correspondingly.

Finally, step 3c needs to be slightly amended as follows. If the job is rejected in $\hat{\mathcal{S}}^{\pi^*}$, but admitted in $\hat{\mathcal{S}}^{\gamma}$, then $\hat{\mathbf{R}}^{\pi^*}(t_{n+1}) = \hat{\mathbf{R}}^{\pi^*}(t_n)$. Let i_a be the order-index of the queue that the arriving job joins in $\hat{\mathcal{S}}^{\gamma}$. Obviously, $R^{\gamma}_{(i_a)}(t_n) > 0$ [there is some empty space in (i_a) at t_n^+ in $\hat{\mathcal{S}}^{\gamma}$]. Since the job was rejected in $\hat{\mathcal{S}}^{\pi^*}$ and due to the matching of the job arrivals and queue accessibilities in the ith largest queues in $\hat{\mathcal{S}}^{\pi^*}$ and $\hat{\mathcal{S}}^{\gamma}$, we must have $\hat{R}^{\pi^*}_{(i_a)}(t_n) = 0$ [there was no empty space in any accessible queue, so also in the (i_a) one]. Due to the decreasing sizes of $\hat{R}^{\pi^*}_{(i)}(t_n)$ as i increases, we must have $\hat{R}^{\pi^*}_{(i)}(t_n) = 0$ for every $i \in \{i_a, i_a + 1, \ldots K\}$. As earlier, from $\hat{\mathbf{R}}^{\pi^*}(t_n) \prec_w \hat{\mathbf{R}}^{\gamma}(t_n)$, we have $\sum_{i=1}^{k} \hat{R}^{\pi^*}_{(i)}(t_n) \leq \sum_{i=1}^{k} \hat{R}^{\gamma}_{(i)}(t_n)$ for every $k \in \{1,2,\ldots,K\}$. From all of the above facts, we get $\sum_{i=1}^{k} \hat{R}^{\pi^*}_{(i)}(t_n) < \sum_{i=1}^{k} \hat{R}^{\gamma}_{(i)}(t_n)$, for every $k \in \{i_a, i_a + 1, \ldots, K\}$. Again, using Fact 3.1(3), we get $\hat{\mathbf{R}}^{\pi^*}(t_{n+1}) \prec_w \hat{\mathbf{R}}^{\gamma}(t_{n+1})$.

The rest of the proof is as that of Proposition 3.1. ■

Remark 4.1: In the standard model, where all queues are always accessible, an alternative proof can be based on showing that

$$(L^{\pi^*}(t) + B, Q_1^{\pi^*}(t), Q_2^{\pi^*}(t), \ldots, Q_K^{\pi^*}(t)) \prec_w (L^{\gamma}(t) + B, Q_1^{\gamma}(t), Q_2^{\gamma}(t), \ldots, Q_K^{\gamma}(t)) \quad \textbf{(11)}$$

for every t. Unfortunately, this avenue does not work in the case of modulated queue accessibility. Specifically, a careful look at the details of the induction shows that this approach does not go through in the case where a job is admitted by γ but not by π^*. The reason is that the rejection of a job by π^* provides information only for the accessible queues in both systems, whereas the states of the inaccessible ones remain obscure, which is not an issue under full accessibility.

5. MULTICLASS JOBS WITH DETERMINISTIC QUEUE ACCESSIBILITY

The third model, where the C-FES/A-MES policy can be shown to minimize the loss flow under assumptions of symmetry, is the following. Again, we have the standard structure of K parallel FCFS queues with finite buffers of equal size B and a single server with randomly modulated server–queue connectivities. However, there are now various classes (flows) of jobs arriving at the system. A job belongs to class $\mathbf{J}_U$, $U \subseteq \{1,2,3,\ldots,K\}$, if it can only be routed (has access to) and join for service (provided there is an empty buffer space) any one of the queues in the set U. We call U the *access set* of class $\mathbf{J}_U$. Note that job classes are differentiated by their access sets.

Let $\mathbf{U}_m$, $m \in \{1,2,3,\ldots,K\}$ be the set of all subsets of $\{1,2,3,\ldots,K\}$ of cardinality m. There are $\binom{K}{m}$ elements in $\mathbf{U}_m$. For an arbitrarily fixed $m \in \{1,2,3,\ldots,K\}$, we consider the collection of traffic flows of job classes $\mathbf{J}_U$, where $U \in \mathbf{U}_m$. Let the arrival process of the job class $\mathbf{J}_U$ $(U \in \mathbf{U}_m)$ be $\tilde{\mathbf{A}}_U = \{\tilde{A}_U(t); t \in \mathcal{R}_+\}$, where $\tilde{A}_U(t)$ is the number of attempted arrivals of class $\mathbf{J}_U$ jobs in the time interval $(0, t]$. The

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

processes $\tilde{\mathbf{A}}_U$ are assumed to be mutually independent Poisson processes of equal rate for all $U \in \mathbf{U}_m$. Note that all classes can access the same number of queues m (although not the same queues) and have identical statistics (*load-balanced input*). The statistics of the service times and connectivity processes are as in Section 2.

Define the set of joint server allocation/job routing policies $\tilde{\Gamma}$ as in Section 4 and denote by $\pi^* \in \tilde{\Gamma}$ the C-FES/A-MES policy in this new setup.

PROPOSITION 5.1: *For the model described in the present section, the C-FES/A-MES policy $\pi^* \in \tilde{\Gamma}$ stochastically minimizes the loss flow and maximizes the throughput, analogously to the situation described in Section 4 (Proposition 4.1).*

PROOF: The structure of the proof is partly analogous to that of Proposition 3.1, enhanced with the additional arguments of the proof of Proposition 4.1, which uses the C-FES/A-MES policy. However, the coupling here has to be *significantly* changed to deal with the structure of the current extended model. We mostly borrow the notation from the previous proofs and simply discuss the new coupling here; the results follow using exactly the same steps as earlier, except for the particular cases treated below.

We compare the evolution of the system under policies π^* (C-FES/A-MES) and γ in $\tilde{\Gamma}$. The couplings of service times and connectivity processes are the same as those in the Propositions 3.1 and 4.1. The coupling of arrivals is subtler and is constructed on the job classes as follows.

At any given time, we first order the queues (in decreasing order) according to their residual capacities in $\mathcal{S}^{\pi^*}$ and $\mathcal{S}^{\gamma}$ and then order the job classes as follows. Let $(i)^{\gamma}$ ($(i)^{\pi^*}$) be the queue with the ith largest residual capacity in $\mathcal{S}^{\gamma}$ ($\mathcal{S}^{\pi^*}$) at time t [hence, that with order-index i in $\mathbf{R}^{\gamma}(t)$ ($\mathbf{R}^{\pi^*}(t)$)]. For every $i \in \{1,2,3,\ldots,K\}$, define the *class block* $\mathbf{F}_i^{\gamma} = \{\mathbf{J}_U, U \in \mathbf{U}_m : (i)^{\gamma} \in U, (i-1)^{\gamma} \notin U, (i-2)^{\gamma} \notin U, \ldots, (2)^{\gamma} \notin U, (1)^{\gamma} \notin U\}$ to be the set of classes whose access sets contain the ith largest residual capacity queue in $\mathcal{S}^{\gamma}$, but no queue of larger residual capacity. Define $\mathbf{F}_i^{\pi^*}$ analogously. A simple example makes the situation clear. Consider a system of three parallel queues 1, 2, 3 and job classes $\mathbf{J}_{\{1,2\}}$, $\mathbf{J}_{\{1,3\}}$, $\mathbf{J}_{\{2,3\}}$. Let the residual capacity vector in $\mathcal{S}^{\gamma}$ be $\mathbf{R}^{\gamma} = (R_1^{\gamma}, R_2^{\gamma}, R_3^{\gamma}) = (7,10,2)$; hence, $\mathbf{R}_{(\cdot)}^{\gamma} = (R_{(1)}^{\gamma}, R_{(2)}^{\gamma}, R_{(3)}^{\gamma}) = (10,7,2)$. Then, we have $\mathbf{F}_1^{\gamma} = \{\mathbf{J}_{\{1,2\}}, \mathbf{J}_{\{2,3\}}\}$, $\mathbf{F}_2^{\gamma} = \{\mathbf{J}_{\{1,3\}}\}$ and $\mathbf{F}_3^{\gamma} = \varnothing$. On the other hand, letting $\mathbf{R}^{\pi^*} = (R_1^{\pi^*}, R_2^{\pi^*}, R_3^{\pi^*}) = (1,5,8)$ be the residual capacity vector in $\mathcal{S}^{\pi^*}$, we have $\mathbf{R}_{(\cdot)}^{\pi^*} = (R_{(1)}^{\pi^*}, R_{(2)}^{\pi^*}, R_{(3)}^{\pi^*}) = (8,5,1)$. Then, we get $\mathbf{F}_1^{\pi^*} = \{\mathbf{J}_{\{2,3\}}, \mathbf{J}_{\{1,3\}}\}$, $\mathbf{F}_2^{\pi^*} = \{\mathbf{J}_{\{1,2\}}\}$, and $\mathbf{F}_3^{\pi^*} = \varnothing$.

Note that $\mathbf{F}_i^{\gamma}$ and $\mathbf{F}_i^{\pi^*}$ have the same *number* of elements for every $i \in \{1,2,\ldots,K\}$ (although not the same classes), because all sets $U \in \mathbf{U}_m$ have equal cardinality m. Moreover, note that the $\mathbf{F}$ blocks form disjoint partitions of $\mathbf{U}_m$; that is, $\bigcup_{i=1}^{K} \mathbf{F}_i^{\gamma} = \mathbf{U}_m = \bigcup_{i=1}^{K} \mathbf{F}_i^{\pi^*}$. Based on the above we can arbitrarily index by l_i the classes in each block $\mathbf{F}_i^{\gamma}$ ($\mathbf{F}_i^{\pi^*}$), $i \in \{1,2,3,\ldots,K\}$. The *coupling* is now done on the classes $\mathbf{J}_U$, $U \in \mathbf{U}_m$, by matching the job arrival times of the l_ith class in $\hat{\mathcal{S}}^{\gamma}$ with those of the l_ith class in $\hat{\mathcal{S}}^{\pi^*}$. In the previous example, we could employ the following matching of job arrivals with respect of classes: $\mathbf{J}_{\{2,3\}}$ in $\hat{\mathcal{S}}^{\gamma}$ with $\mathbf{J}_{\{1,3\}}$ in $\hat{\mathcal{S}}^{\pi^*}$,

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

$\mathbf{J}_{\{1,2\}}$ in $\hat{\mathcal{S}}^{\gamma}$ with $\mathbf{J}_{\{2,3\}}$ in $\hat{\mathcal{S}}^{\pi^*}$, and $\mathbf{J}_{\{1,3\}}$ in $\hat{\mathcal{S}}^{\gamma}$ with $\mathbf{J}_{\{1,2\}}$ in $\hat{\mathcal{S}}^{\pi^*}$. A key feature of this coupling is that matched arrivals have access to the ith largest residual capacity queue both in system $\hat{\mathcal{S}}^{\gamma}$ and $\hat{\mathcal{S}}^{\pi^*}$, due to the construction of the $\mathbf{F}$ blocks and the matching of classes. That enables the coupling of arrival times to be consistent with that of connectivities. This is an important piece of *intuition* behind the previous constructions. In our example, queue 1 in $\hat{\mathcal{S}}^{\gamma}$ (the second largest in $\mathbf{R}^{\gamma} = (7,10,2)$) is accessible by $\mathbf{J}_{\{1,2\}}$, which is matched to $\mathbf{J}_{\{2,3\}}$ in $\hat{\mathcal{S}}^{\pi^*}$; the latter job class can access queue 2 (the second largest in $\mathbf{R}^{\pi^*} = (5,1,8)$).

Steps 1 (connectivity switchings) and 2 (service completions) in the induction on decision instants are identical to those in Proposition 3.1. Only steps 3a and 3c need to be slightly amended as follows.

In step 3a, we examine the system at the arrival instant of a job of some class J_U and assume that it is admitted in both $\hat{\mathcal{S}}^{\pi^*}$ and $\hat{\mathcal{S}}^{\gamma}$. Note that, in $\hat{\mathcal{S}}^{\pi^*}$, the C-FES/A-MES policy routes the job to the largest residual capacity queue among those in U, which has some order-index i. This implies that all other queues in U have order-indices larger than i in $\hat{\mathbf{R}}^{\pi^*}$. The previously constructed coupling of arrivals based on the $\mathbf{F}$ blocks *ensures* that the largest residual capacity queue in U in $\hat{\mathcal{S}}^{\gamma}$ also has order-index i. Therefore, in $\hat{\mathcal{S}}^{\gamma}$, the job is routed to some queue with order-index j, such that $j \geq i$ (actually, trying to capture this effect has provided the *main intuition* for the construction of the coupling). From Fact 3.1(1), we get $\hat{\mathbf{R}}^{\pi^*}(t_{n+1}) \prec_w \hat{\mathbf{R}}^{\gamma}(t_{n+1})$, using the induction hypothesis $\hat{\mathbf{R}}^{\pi^*}(t_n) \prec_w \hat{\mathbf{R}}^{\gamma}(t_n)$ and noting that, at t_{n+1}, both $\hat{\mathcal{S}}^{\pi^*}$ and $\hat{\mathcal{S}}^{\gamma}$ see their residual capacities decrease by 1 at queues (i) and (j) correspondingly.

Finally, in step 3c, we assume that the job is rejected in $\hat{\mathcal{S}}^{\pi^*}$, but admitted in $\hat{\mathcal{S}}^{\gamma}$. Then, $\hat{\mathbf{R}}^{\pi^*}(t_{n+1}) = \hat{\mathbf{R}}^{\pi^*}(t_n)$. Let i_a be the order-index of the queue (i_a) that the arriving job joins in $\hat{\mathcal{S}}^{\gamma}$. Obviously, $R^{\gamma}_{(i_a)}(t_n) > 0$ (there must be some empty space in (i_a) at t_n^+ in $\hat{\mathcal{S}}^{\gamma}$). Since the job was rejected in $\hat{\mathcal{S}}^{\pi^*}$ and due to the matching of job arrivals and classes in the ith largest queues in $\hat{\mathcal{S}}^{\pi^*}$ and $\hat{\mathcal{S}}^{\gamma}$, we must have $\hat{R}^{\pi^*}_{(i_a)}(t_n) = 0$ (there was no empty space in any accessible queue, so also in the (i_a)th one). Due to the decreasing sizes of $\hat{R}^{\pi^*}_{(i)}(t_n)$ as i increases, we must have $\hat{R}^{\pi^*}_{(i)}(t_n) = 0$ for every $i \in \{i_a, i_a+1, \ldots, K\}$. The proof of this case can then be clinched as in Proposition 4.1.

The rest of the proof is as that of Proposition 3.1. ■

Remark 5.1 (Special Cases): Consider the following two extreme cases of the model (as an honesty check on it). If $m = 1$, then there are K arrival streams, each of which can access only one queue; so, we recapture the basic model of Section 2. On the other hand, if $m = K$, there is only one class of jobs which can access any of the queues. Then, we get a special case of the model in Section 4, with all the queues being always accessible. The coupling also reduces appropriately.

Remark 5.2: Note that the *symmetric* Poisson multiclass job flows $\tilde{\mathbf{A}}_U$ can be equivalently generated by a single (joint) Poisson flow of rate $\binom{K}{m}\lambda$, which is Bernoulli-split into $\binom{K}{m}$ independent Poisson flows of rate λ each. From this point of view, we can consider the system as having a single class input with random queue accessi-

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

bility of a special kind (i.e., the set of accessible queues has always cardinality m). We can then couple the two systems so that the ordered queues available to an arriving job are the same in both. Unfortunately, this results in no simplification in the proof of Proposition 5.1, since structures analogous to the class blocks $\mathbf{F}_i^\gamma$ have to be used again to clinch the result.

Remark 5.3 (Structural Asymmetry): If we do not include all of the classes of $\mathbf{U}_m$ in the model (resulting in structural asymmetry), the proof of Proposition 5.1 collapses under the used coupling. To see why, assume that there are three parallel queues 1, 2, 3 and only two job classes $\mathbf{J}_{\{1,2\}}$ and $\mathbf{J}_{\{2,3\}}$; the $\mathbf{J}_{\{1,3\}}$ class is missing. Suppose that at some time the queues are ranked 2, 3, 1 in $\hat{\mathcal{S}}^\gamma$, according to their residual capacities, whereas in $\hat{\mathcal{S}}^{\pi^*}$, the ranking is 1, 2, 3. Given the coupling employed in the proof, assume that a job arrives on $\mathbf{J}_{\{1,2\}}$ in $\hat{\mathcal{S}}^\gamma$, which is matched to $\mathbf{J}_{\{2,3\}}$ in $\hat{\mathcal{S}}^{\pi^*}$, because $\mathbf{J}_{\{1,3\}}$ is missing. This job can be routed to either of the queues 1 or 2 in $\hat{\mathcal{S}}^\gamma$ but has to be routed to queue 2 in $\hat{\mathcal{S}}^{\pi^*}$. If the job is routed to queue 2 in $\hat{\mathcal{S}}^\gamma$, then the proof of step 3a collapses (Fact 3.1(1) is not applicable). However, had the class $\mathbf{J}_{\{1,3\}}$ been present, it could have been matched to $\mathbf{J}_{\{1,2\}}$ in $\hat{\mathcal{S}}^\gamma$ and the job would have been routed to queue 1 in $\hat{\mathcal{S}}^{\pi^*}$, which has the largest residual capacity; application of Fact 3.1(1) would have established the result.

6. C-FES ROBUSTNESS: SOME CASE STUDIES

An interesting practical question is how large the buffers should be in order for the loss flow to be negligible [i.e., the blocking (rejection) probability of incoming jobs to be close to zero]. We have simulated the basic model of Section 2 in slotted time (as described in Remark 3.1), with four queues and buffer sizes $B = 1/5/10/100/500/1000$, using the C-FES policy. The results are presented in Figures 1 and 2. The service times have geometric distributions of rate 0.8. The probability of each queue being connected to the server is 0.5 during each time slot. We have made simulation runs for arrival rates $\lambda = 0.10/0.12/0.14/0.18/0.185/0.19/0.20/0.25/0.40/0.70$ (probability of a job arrival at each queue in a time slot). Each simulation run is 10^7 steps long.

Assuming that the buffers are *infinite*, the region of arrival rates for which the above system can be stable (queue lengths be ergodic under some server allocation policy) is $\lambda \in [0, 0.187)$. This can be computed from the formulas in [10] (allocating the server to the connected queue with maximum length stabilizes the system throughout this region). Therefore, the system *capacity* is $\lambda^* = 0.187$. For any $\lambda > \lambda^* = 0.187$, the queue lengths eventually blow up to infinity under *any* server allocation policy (infinite buffers).

Of interest in Figure 1 is that buffer sizes as small as $B = 10$ (100) are adequate for reducing the blocking probability of arriving jobs below 0.01 (0.001), even under heavy traffic of $\lambda = 0.185$ ($\lambda/\lambda^* = 0.185/0.187 = 99\%$ close to capacity). A small degradation of 4% can be observed for five buffer places, around the critical arrival rate 0.187, and a significant one of 30% for a single buffer space. Regarding the

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

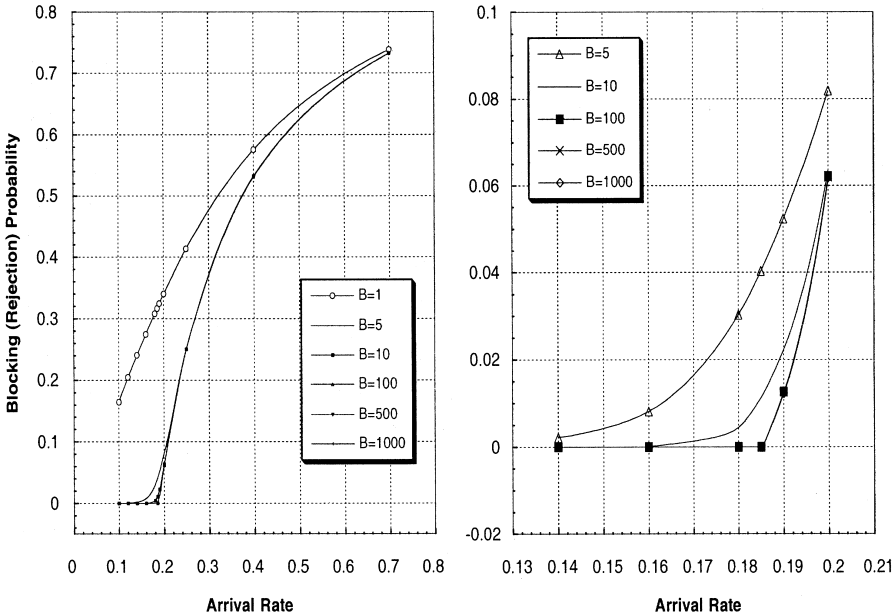


FIGURE 1. Dependence of the blocking (rejection) probability of an incoming job on the arrival rate per queue, for several buffer sizes of B. The second graph is a magnification of the $(0.14, 0.19) \times (0, 0.1)$ region of the first.

average delay of admitted jobs, Figure 2 clearly shows its steep rise when the arrival rate approaches the critical value 0.187, especially for large buffers (>100). The result is consistent with intuition.

Another issue we have explored experimentally is the sensitivity of the aggregate throughput of the C-FES policy to buffer and connectivity asymmetries. The results are presented in Figure 3. We have simulated (in slotted time) a system of *two* queues. A total of 100 buffer places are distributed over the two queues. We plot the admission probability for various allocations of buffer spaces to the two queues. The service times have geometric distributions of rate 0.8 and the arrival rates are the same at both queues (symmetric arrivals). We have experimented with two scenarios. In Case 1, the connectivities are symmetric, each queue having probability 0.55 of being connected during each time slot. In Case 2, the first queue has probability 0.42146 of being connected during a time slot, whereas the second has probability 0.65 (so $0.65000/0.42146 - 1 = 0.542 \approx 50\%$ asymmetry in favor of the second queue). These probabilities have been chosen so that the two cases have the same load capacity $\lambda^* = 0.319$ (critical arrival rate computed from [10] under infinite buffers). In both cases, we have run the simulations (10^7 steps) with arrival rate $\lambda = \lambda^* - 10^{-7} = 0.319 - 10^{-7}$ at each queue (extremely heavy traffic).

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

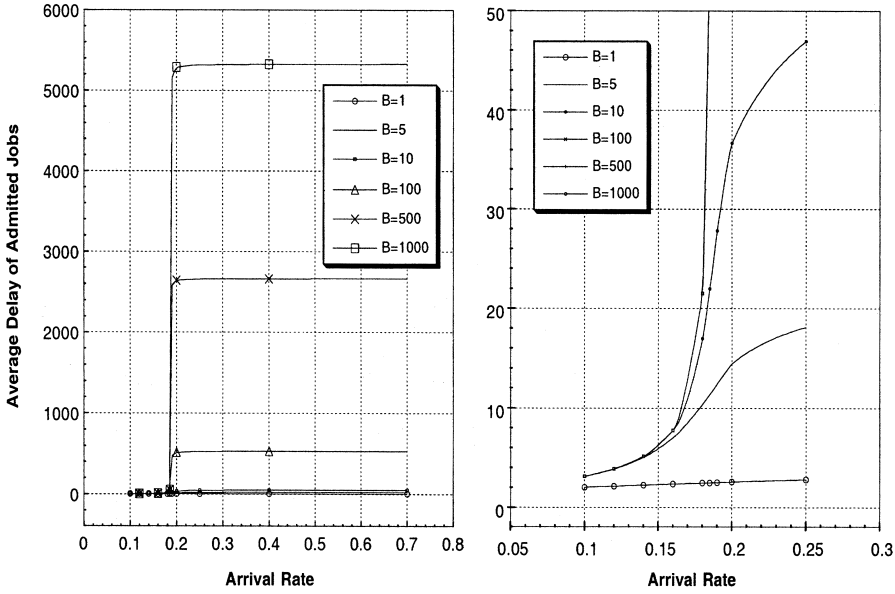


FIGURE 2. Dependence of the average delay of admitted jobs on the arrival rate per queue for several buffer sizes. The second graph is a magnification of the (0.1, 0.25) × (0, 0.50) region of the first.

A couple of interesting observations can be made from Figure 3, indicating that the aggregate throughput is rather robust with respect to asymmetries. Considering first the case of symmetric connectivities, observe that the variation of admission probability is less than 1% throughout the range $B_1 = \{20,21,22,\ldots,78,79,80\}$ for the buffer size of the first queue ($B_2 = 100 - B_1$), hence, up to $(B_2/B_1) - 1 = (80/20) - 1 = 300\%$ buffer asymmetry. Next, observe that for asymmetric connectivities (50% tilt in favor of the second queue), the discrepancies in admission probabilities in the region $B_1 = \{20,21,\ldots 79,80\}$ are below 1%, compared to those of symmetric connectivities. This behavior can be explained by arguing that 20 buffer places (per queue) are enough for admitting almost every arriving job, even in this highly load-stressed system (just 10^{-7} from capacity). This is consistent with our previous observations.

7. CONCLUSIONS AND FINAL REMARKS

The policy C-FES (C-FES/A-MES) has been shown to stochastically minimize the loss flow and maximize the throughput for symmetric Markovian systems of parallel queues with finite buffers and random server connectivities (and job routing). Experimental evidence, emerging from a couple of key case studies, indicates that buffer sizes as small as 10 are adequate for admitting almost 99% of the arriving jobs

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

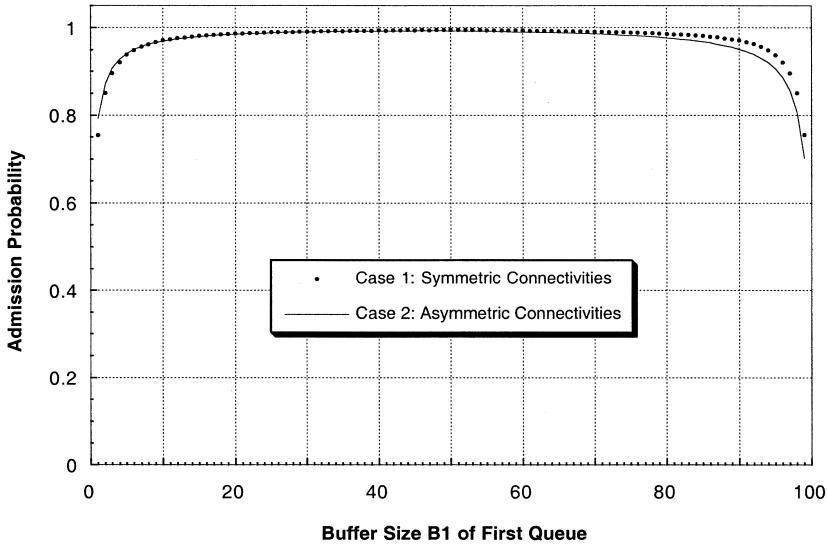


FIGURE 3. Dependence of the admission probability of incoming jobs (aggregate throughput) on the buffer size B_1 of the first queue (the buffer size of the second queue is $B_2 = 100 - B_1$), for the cases of symmetric and asymmetric connectivities.

(even under heavy traffic) and that the C-FES policy is quite robust with respect to asymmetries of buffers and connectivities.

There are several other issues of interest, which we are currently exploring. For asymmetric systems, the couplings used here collapse. We are investigating several simple preemptive job routing/server allocation policies for optimizing the dynamics under lack of symmetry. Certain issues regarding the successful stochastic comparison of policies need to be resolved in order to establish analogous results for asymmetric structures. A related issue is the allocation of buffer places to the queues in order to minimize the loss flow.

Recall that the first model is a special case of the third. We would like to extend the results to a new model which supersedes all three, by allowing class-dependent accessibility sets and, additionally, random modulation within each one in a meaningful way. Unfortunately, we have not been successful in constructing a coupling for this model, except in degenerate cases. The existing couplings collapse and cannot be salvaged. What is probably needed is an extension of the coupling concept introduced in the analysis of the third model. We are currently working in this direction, but certain technical issues still remain unresolved.

Another interesting problem is the identification of optimal *nonpreemptive* policies under *nonmemoryless* arrivals and/or connectivities and/or service. The intuition in this case changes significantly. Indeed, at some instants, it may be better for the server to idle, in anticipation that a currently nonconnected queue with an almost

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press

full buffer will soon become connected, instead of becoming engaged to a large job of a currently connected queue with many empty buffer spaces. Such *anticipative idling* can potentially prevent an overflow in the congested queue. We are currently studying the parallel queue model under renewal interarrival, service, and connectivity switching times. Certain technical issues still need to be resolved before we can characterize the server allocation scheme which minimizes the loss flow in the set of nonpreemptive policies which permit the server to idle.

Acknowledgments

The authors would like to thank an anonymous referee for the helpful comments and suggestions.

The work of the first author was supported in part by the National Science Foundation, that of the second author by grant IIS-9988095.

References

1. Bambos, N. & Michailidis, G. (1995). On the stationary dynamics of parallel queues with random server connectivities. Proceedings of the 34th Conference on Decision and Control, New Orleans, LA, pp. 3638–3643.
2. Kamae, T., Krengel, V., & O'Brien, G.L. (1978). Stochastic inequalities on partially ordered spaces. *Annals of Probability* 6: 1044-1049.
3. Lindvall, T. (1992). *Lectures on the coupling method*. New York: Wiley.
4. Lott, C. & Teneketzis, D. (2000). On the optimality of an index rule in multichannel allocation for single-hop mobile networks with multiple service classes. *Probability in the Engineering and Information Sciences* 14: 259–297.
5. Marshall, A.W. & Olkin, I. (1979). *Inequalities: Theory of majorization and its applications*. New York: Academic Press.
6. Menich, R. & Serfozo, R.D. (1991). Optimality of routing and servicing in dependent parallel processing systems. *Queueing Systems: Theory and Applications* 9: 403-418.
7. Ross, S.M. (1995). *Stochastic processes*, 2nd ed. New York: Wiley.
8. Shaked, M. & Shanthikumar, J.G. (1994). *Stochastic orders and their applications*. New York: Academic Press.
9. Sparragis, P.D., Towsley, D., & Cassandras, C.G. (1993). Extremal properties of the shortest/longest non-full queue policies in finite-capacity systems with state-dependent service rates. *Journal of Applied Probability* 30: 223–236.
10. Tassiulas, L. & Ephremides, A. (1993). Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Transactions on Information Theory* IT-39: 466–478.
11. Wasserman, K.M. & Bambos, N. (1996). Optimal server allocation to parallel queues with finite capacity buffers. *Probability in the Engineering and Information Sciences* 10: 279–285.
12. Wasserman, K.M. & Lennon-Olsen, T. (2001). On mutually interfering parallel servers subject to external disturbances. *Operations Research* 49: 700–709.

https://doi.org/10.1017/S0269964802162048 Published online by Cambridge University Press