A tutorial on some new methods for performance evaluation of queueing networks

P. R. Kumar*
Dept. of Electrical and Computer Engineering, and
Coordinated Science Laboratory
University of Illinois
1308 West Main Street
Urbana, IL 61801/USA

Abstract

In the 1970s, Baskett, Chandy, Muntz and Palacios, Kelly, and others, generalized the earlier results of Jackson and obtained explicit solutions for the steady–state distributions of some restricted queueing networks. These queueing networks are called "product–form networks," due to the structure of their explicit solutions. The class of such tractable networks is quite small, however. For example, if customers require different mean service times on different revisits to the same server, or if customers on a later visit are given higher priority, then very little is known concerning whether the network is even stable, or what form the steady–state distribution has if it exists.

Recently, some new methods have been developed for establishing the stability of a system, and for obtaining bounds on key performance measures such as mean delay, mean number in system, or mean throughput. Since they are based on the well developed computational tool of linear programming, these methods can be widely employed in diverse applications in communication networks, computer systems, and manufacturing systems. In this paper, we provide a tutorial exposition of some of these recent developments.

1 Introduction

Due to the important works of Jackson [1, 2], Baskett, Chandy, Muntz and Palacios [3], Kelly [4] and others, explicit product form formulae are available for the steady–state distribution of some queueing networks. However, this class of tractable queueing networks is quite restrictive. For example, if a class of customers visits the same server twice, and the mean service times are different for the two visits, then very little is known about whether there is in fact a steady–state distribution, or what it is if it exists. The situation is similarly intractable if the server gives preemptive priority to customers on their second visit.

Networks where services at different stages have different mean values are however important in applications, e.g., semiconductor manufacturing systems (see Wein [5]). Also, in most systems, one would like

*The research reported here has been supported in part by the USARO under Grant No. DAAH04-95-1-0090, in part by the National Science Foundation under Grant No. ECS-9403571, and in part by the Joint Services Electronics Program under Contract No. N00014-90-J1270.

to control the behavior of a network by making scheduling decisions which give priority to some customers over others. Thus, for these and other reasons, it is of significant interest to develop methods to study the performance of systems which fall outside the previously studied class of "product form networks."

We will address two issues in this tutorial. First, we will address the issue of determining when a network has a steady–state distribution with a finite mean number of customers in steady–state. When this is so, we will say that the system is "stable." In Section 2 we give an example to show that some systems can be unstable. In Sections 6 and 7 we provide a sufficient condition for stability, which consists simply of determining whether a certain linear program is feasible. The second issue we address is how one may obtain lower and upper bounds for the mean number of customers or throughput in steady–state, by simply solving a linear program. In Sections 8-10, we develop this procedure for open queueing networks, i.e., networks with exogenous arrivals, and in Section 11 we deal with closed queueing networks, buffer priority policies, and others systems with special structure.

Due to space limitations, we only develop here those ideas that are fundamental to these results. Throughout we provide some additional references which allow the reader to further explore this emerging field.

2 An unstable queueing network

The following example shows that even simple queueing networks can be unstable.

Example 1

Consider the system shown in Figure 1, drawn from Lu and Kumar [6]. There are two servers. Server 1 serves two "buffers" b_1 and b_4, while Server 2 servers the other two buffers b_2 and b_3. Customers arrive periodically with interarrival time $\frac{1}{\lambda}$ to buffer b_1. At buffer b_i they require s_i time units of service from the server serving that buffer. Customers leave buffer b_i after securing the required service, and proceed to b_{i+1}, with the exception that after b_4 they leave the system.

A necessary condition for Servers 1 and 2 to cope with the arrival rate λ is that

$$\lambda(s_1 + s_4) \leq 1 \text{ and } \lambda(s_2 + s_3) \leq 1. \tag{1}$$

However, condition (1), even with strict inequalities, is *not* sufficient to ensure that the number of customers in the system remains bounded. That will also depend on how the servers *schedule* their operations.

Let us suppose that Server 1 gives higher preemptive priority to customers in b_4 over customers in b_1. Whenever b_4 is not empty, Server 1 picks the customer at the head of buffer b_4 for service. When b_4 is empty, Server 1 works on the customer (if any) at the head of buffer b_1. If b_1 is also empty, the server remains idle. In a similar way, let us suppose that Server 2 gives higher pre-emptive priority to buffer b_2 over b_3. Such a policy is a *buffer priority policy*, using a priority ordering $\{b_4, b_1, b_2, b_3\}$.

Suppose that $\lambda_1 = 1, s_1 = s_3 = 0$, and $s_2 = s_4 = 2/3$. We note that even if a service time is zero, a customer must still wait its turn for service.

Let $x_i(t)$ denote the number of customers in buffer b_i at time t, and denote by $x(t)$ the vector $(x_1(t), x_2(t), x_3(t), x_4(t))$. Let us consider an initial condition $x(0^-) = (N, 0, 0, 0)$ at time 0^-, and trace the future evolution of the system. We will now show that this queueing network is unstable in the sense that the number of customers in the system is unbounded.

Instantaneously (since $s_1 = 0$) the N customers in b_1 are transferred to b_2, and the system state becomes $(0, N, 0, 0)$. Hence Server 2 commences working on the customers in b_2. While it is working, new customers arrive to b_1. Since b_4 is empty, Server 1 immediately attends to these new arrivals as they come, and transfers them to b_2. At time $2N^-$, b_2 is empty once again. The total of N initial customers, as well as the $2N$ customers that subsequently arrived, have been transferred to b_3. Hence the system state is $x(2N^-) = (0, 0, 3N, 0)$. This being the first time at which b_2 is empty, Server 2 can work on b_3. Instantaneously it transfers these $3N$ customers to b_4 (recall that $s_3 = 0$). Hence, the system state becomes $(0, 0, 0, 3N)$. This is the first time at which b_4 is non-empty. Server 1 now has to serve b_4 rather than b_1, since b_4 has higher priority. It spends $2N$ time units serving these $3N$ parts. Meanwhile the arriving customers to b_1 are left unattended. Thus, at time $4N^-$, the new state of the system is $x(4N^-) = (2N, 0, 0, 0)$. We see that at time $4N^-$, the system state is the double of the initial condition $x(0^-) = (N, 0, 0, 0)$. This process repeats itself, and at time $12N^-$, the system state is again doubled, to $x(12N^-) = (4N, 0, 0, 0)$. Thus, we see that the number of customers in the system is unbounded. □

A similar system and others are shown to be unstable in Kumar and Seidman [7] when the servers incur set–up times when switching between buffers. The well known First Come First Serve (FCFS) policy can also be unstable. Consider a system of the form shown in Figure 2. It is almost the same as the system in Figure 1, except that customers revisit Server 2 many times before returning to Server 1. Suppose now that the FCFS service discipline is employed at each server. This discipline tends to shepherd the customers in b_2 together as one bunch, before letting them move to b_8. To see this, consider a number of customers in b_2, when b_3, b_4, b_5, b_6 and b_7 are empty. Suppose no arrivals come to b_2 from b_1. Then all the

customers at b_2 will move to b_3, then they all move to b_4, etc., under the FCFS discipline. Thus one can regard the system of Figure 2 with many many revisits to Server 2, operating under FCFS, as basically approximating the buffer priority policy in Figure 1. This line of argument is made precise in Bramson [8], who shows that such a stochastic system with exponentially distributed interarrival and service times is unstable when there are many revisits. Seidman [9] provides a different example of a deterministic system which is unstable under FCFS.

What positive results are available for deterministic systems? Lu and Kumar [6] have shown that the Last Buffer First Serve (LBFS) buffer priority policy which uses the priority order $\{b_L, b_{L-1}, b_{L-2}, \ldots, b_1\}$ is stable for all systems whenever the capacity condition (1) holds with strict inequalities. They also show that the FBFS buffer priority policy, which uses the priority ordering $\{b_1, b_2, \ldots, b_L\}$, the reverse of LBFS, is also similarly stable. For systems with many customer types each following a possible different route, they also identify some stable buffer priority policies. For systems which incur "set-up times," Kumar and Seidman [7] have identified stable classes of scheduling policies.

3 The transition probabilities of the controlled queueing network

In much of this paper, for simplicity of exposition only, we will focus on networks with a single fixed route that may involve visiting some servers more than once. Such networks are called "re-entrant lines" (see Kumar [10]). The results here are drawn from Kumar and Meyn [11] to which we refer the reader for further results and details.

Consider a system of the form shown in Figure 3. It consists of S servers labelled 1, 2, ..., S. Customers arrive from an external source as a Poisson process of rate λ to buffer b_1 served by Server $\sigma(1) \in \{1, 2, \ldots, S\}$. After having their service completed at buffer b_i served by Server $\sigma(i)$, customers travel next to buffer b_{i+1} served by Server $\sigma(i+1)$. After being served at b_L, they leave the system. The random service times of customers at buffer b_i are exponentially distributed with mean $\frac{1}{\mu_i}$. We suppose that all interarrival and service times are independent random variables. Let $\rho_\sigma := \sum_{\{i:b_i \text{ is served by Server } \sigma\}} \frac{\lambda}{\mu_i}$ be the *nominal load* on Server σ. Throughout we suppose that the arrival rate λ is within the *capacity* of the system, i.e.,

$$\rho_\sigma < 1 \text{ for all servers } \sigma. \tag{2}$$

Let us define the state $x(t) := (x_1(t), x_2(t), \ldots, x_L(t))^T$, where $x_i(t)$ is the number of customers in the system at time t. The evolution of $x(t)$ is guided by the *scheduling policy*. Let us define "control actions"

$\{w_i(t) : 1 \leq i \leq L\}$ which determine the choice of which buffer is worked on at time t, by

$$\begin{aligned} w_i(t) &:= 1 \text{ if buffer } b_i \text{ is being served by server } \sigma(i), \\ &:= 0 \text{ otherwise.} \end{aligned}$$

We assume that all stochastic processes are right continuous with left hand limits.

Denote by e_i the unit vector $(0, \ldots, 0, 1, 0, \ldots, 0)^T$ with a 1 in the ith place. Under the control actions $w(t)$, the stochastic process $\{x(t)\}$ is a controlled Markov chain with the following transition probabilities (where $o(h)$ is a term such that $o(h)/h \to 0$ as $h \searrow 0$):

$$\begin{aligned} \text{Prob } (x(t+h) = y | x(t) = x, w(t) = w) &= \lambda h + o(h) \quad \text{if } y = x + e_1, \\ &= \mu_i w_i h + o(h) \text{ if } y = x - e_i + e_{i+1} \text{ and } 1 \leq i \leq L-1, \\ &= \mu_L w_L h + o(h) \text{ if } y = x - e_L. \end{aligned}$$

Instead of studying the continuous time controlled Markov chain $\{x(t) : 0 \leq t < +\infty\}$, it is more convenient to study an equivalent sampled discrete-time controlled Markov chain $\{x(\tau_n) : n = 0, 1, 2, \ldots\}$. Let us rescale time so that $\lambda + \sum_{i=1}^{L} \mu_i = 1$. If a buffer b_i is not being served by its server, let us suppose that there is a fictitious customer in the buffer who is being served. We sample the system at the times $\{\tau_n\}$, which comprise of all arrival times, and real or fictitious service completion times. Let us make the assumption that the scheduling policy does not change its decisions in between the sampling times, i.e.,

$$w_i(t) = w_i(\tau_n) \quad \text{for } \tau_n \leq t < \tau_{n+1}. \tag{3}$$

We shall call such a scheduling policy *non-interruptive*. Then the resulting discrete time controlled Markov chain is a faithful replica of the original continuous time Markov chain $\{x(t)\}$ in the following sense. If one has a steady–state distribution, so does the other, and the steady state distributions are the same. (This process is called "uniformization"; see Lippman [12]). The transition probabilities of this sampled controlled Markov chain are:

$$\begin{aligned} \text{Prob } (x(\tau_n + 1) = y | x(\tau_n) = x, w(\tau_n) = w) &= \lambda \text{ if } y = x + e_1, \\ &= \mu_i \text{ if } y = x - e_i + e_{i+1} \text{ and } w_i = 1, \\ &\qquad \text{for } 1 \leq i \leq L-1, \\ &= \mu_L \text{ if } y = x - e_L \text{ and } w_L = 1. \end{aligned} \tag{4}$$

4 Scheduling policies

From now on, we focus on scheduling policies that in addition to being non–interruptive are also *non-idling*. By this we mean that a server never stays idle when there is a customer waiting for service. This can be

quantitatively expressed as:

$$\sum_{\{i:\sigma(i)=\sigma\}} w_i(\tau_n) = 1 \Leftrightarrow \sum_{\{i:\sigma(i)=\sigma\}} x_i(\tau_n) \geq 1. \tag{5}$$

Note that one consequence is

$$x_j(\tau_n) = \sum_{\{i:b_i \text{ is served by } \sigma(j)\}} w_i(\tau_n) x_j(\tau_n). \tag{6}$$

Two examples of non-idling and non-interruptive scheduling policies are buffer priority policies and the FCFS policy. A *buffer priority policy* θ is a scheduling policy employing a *priority ordering* $\theta = \{\theta(1), \theta(2), \ldots, \theta(L)\}$, which is a permutation of $\{1, 2, \ldots, L\}$. If buffers b_i and b_j are both served by the same server, i.e., $\sigma(i) = \sigma(j)$, then b_j is given higher pre-emptive priority over b_i if $\theta^{-1}(j) < \theta^{-1}(i)$. So the control actions satisfy,

$$\begin{aligned} w_i(t) &= 1 \text{ if } x_i(t) \geq 1 \text{ and } x_j(t) = 0 \text{ for all } j \text{ with } \sigma(j) = \sigma(j) \text{ and } \theta^{-1}(j) < \theta^{-1}(i), \\ &= 0 \text{ otherwise.} \end{aligned} \tag{7}$$

One consequence is that the stochastic processes $\{x(t)\}$ and $\{x(\tau_n)\}$ are time–homogeneous Markov chains, since the control $w(t)$ is purely a function of $x(t)$. (In the language of Markov Decision Processes, the scheduling policy is a *stationary* policy).

If, instead, the scheduling policy employed is FCFS, then $w_i(t)$ is not determined purely from $x(t)$. It depends on the times of past events. Hence $\{x(t)\}$ and $\{x(\tau_n)\}$ are *not* Markov chains. They are controlled Markov chains with control actions dependent on past history.

5 A fundamental identity

We will study the evolution of products of the form $x_i(\tau_{n+1})x_j(\tau_{n+1})$. The exact formulas depend on whether i or j are equal to 1, and whether $|i-j| = 0, 1$ or more.

To illustrate the main ideas, consider the case $i = 1$ and $3 \leq j \leq L$. Then, given $x(\tau_n)$ and $w(\tau_n)$, from (4) one has

$$\begin{aligned} x_1(\tau_{n+1})x_j(\tau_{n+1}) &= (x_1(\tau_n)+1)x_j(\tau_n) \text{ with probability } \lambda, \\ &= (x_1(\tau_n)-1)x_j(\tau_n) \text{ with probability } \mu_1 w_1(\tau_n), \\ &= x_1(\tau_n)(x_j(\tau_n)+1) \text{ with probability } \mu_{j-1} w_{j-1}(\tau_n), \\ &= x_1(\tau_n)(x_j(\tau_n)-1) \text{ with probability } \mu_j w_j(\tau_n), \\ &= x_1(\tau_n)x_j(\tau_n) \text{ with probability } (1 - \lambda - \mu_1 w_1(\tau_n) - \mu_{j-1} w_{j-1}(\tau_n) - \mu_j w_j(\tau_n)). \end{aligned}$$

So

$$E[x_1(\tau_{n+1})x_j(\tau_{n+1})|x(\tau_n),w(\tau_n)] = x_1(\tau_n)x_j(\tau_n)+(\lambda-\mu_1 w_1(\tau_n))x_j(\tau_n) +(\mu_{j-1}w_{j-1}(\tau_n)-\mu_j w_j(\tau_n))x_1(\tau_n).$$

In a similar way, one can evaluate $E[x_i(\tau_{n+1})x_j(\tau_{n+1})|x(\tau_n),w(\tau_n)]$ for all i, j.

Lemma 1: The fundamental identity. *Let $Q=[q_{ij}]$ be a symmetric matrix. Then,*

$$(i)\quad E[x^T(\tau_{n+1})Qx(\tau_{n+1})|x(\tau_n),w(\tau_n)] = x^T(\tau_n)Qx(\tau_n)+f(Q,x(\tau_n),w(\tau_n)) +g(Q,w(\tau_n)),\ \text{where} \tag{8}$$

$$f(Q,x(\tau_n),w(\tau_n)) := 2\lambda e_1^TQx(\tau_n)+2\textstyle\sum_{i=1}^{L-1}\mu_i w_i(\tau_n)(e_{i+1}-e_i)^TQx(\tau_n) -2\mu_L w_L(\tau_n)e_L^TQx(\tau_n),\ \text{and} \tag{9}$$

$$g(Q,w(\tau_n)) := \lambda e_1^TQe_1+\textstyle\sum_{i=1}^{L-1}\mu_i w_i(\tau_n)(e_{i+1}-e_i)^TQ(e_{i+1}-e_i)+\mu_L w_L(\tau_n)e_L^TQe_L. \tag{10}$$

ii) $|g(Q,x(\tau_n),w(\tau_n))|\le M$ for all n.

6 Definition and consequences of stability

We now state precisely what we will mean by the word "stability" in this paper. Let us denote by $|x(\tau_n)|:=\sum_{i=1}^L x_i(\tau_n)$ the number of customers in the system at time τ_n. We shall say that a queueing network is *stable* under a given scheduling policy if the long run average of the expected number of customers in the system is bounded for all *deterministic* initial conditions, i.e.,

$$\text{limsup}_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}E|x(\tau_n)|<+\infty,\ \text{for all } x(\tau_o)=x\in Z_+^L. \tag{11}$$

Lemma 2: Implication of stability for stationary scheduling policies. *Suppose the scheduling policy u is* stationary, *i.e., there is a function u such that $w(\tau_n)=u(x(\tau_n))$.*

i) *Then $\{x(\tau_n)\}$ is a time-homogeneous Markov chain. with a single closed communicating class.*

ii) *If the system is stable, i.e., (11) holds, then the Markov chain has a steady–state probability distribution π, i.e., for every initial condition $x(\tau_0)$,*

$$\lim_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}\ Prob\ (x(\tau_n)=x)=\pi(x).$$

iii) *If the system is stable, then the mean number of customers in steady–state is finite, i.e., $\sum_{x\in Z_+^L}|x|\pi(x)<+\infty$.*

Proof The origin 0 is reachable from every state since the scheduling policy is non-idling. Hence the result is immediate from (4). If the Markov chain has no steady–state distribution, then (see [13]) $\lim_{n\to\infty}$ Prob$(x(\tau_n) = i) = 0$ for every i. Then, however, for every M, $\lim_{n\to\infty} P(|x(\tau_n)| \leq M) = 0$. Hence $\liminf_{N\to\infty} \frac{1}{N}\sum_{n=0}^{N-1} E|x(\tau_n)| \geq M$ for every M. This contradicts (11). So (iii) has to hold. Moreover $\lim_{N\to\infty} \frac{1}{N}\sum_{n=0}^{N-1} E\sum_{|x|\leq M}|x|$ Prob $(x(\tau_n) = x) = \sum_{|x|\leq M}|x|\pi(x)$. Hence, (iv) follows from (11) by invoking a limiting argument with $M \to \infty$. □

7 A procedure to establish the stability of queueing networks and scheduling policies

In this section we show that a queueing network is stable if we can find a feasible solution to a finite set of linear inequalities, i.e., a linear program.

Definition: Copositive matrix. *A symmetric matrix Q is called* copositive *if $x^TQx \geq 0$ for all vectors x with nonnegative components, i.e., for all $x \in \mathbb{R}_+^L$.*

Lemma 3. *If Q is a symmetric* nonnegative *matrix, i.e., $Q = [q_{ij}]$ with $q_{ij} \geq 0$, then it is copositive.*

Theorem 1: A sufficient condition for stability. *The system is stable if there is a copositive matrix Q and a $\gamma > 0$, such that f defined in (9) satisfies,*

$$f(Q, x(\tau_n), w(\tau_n)) \leq -\gamma|x(\tau_n)| \text{ for all } n. \tag{12}$$

Proof *The proof is based on (8). By taking expectations on both sides and summing, we obtain*

$$\sum_{n=0}^{N-1} Ex^T(\tau_{n+1})Qx(\tau_{n+1}) = \sum_{n=0}^{N-1} Ex^T(\tau_n)Qx(\tau_n) + \sum_{n=0}^{N-1} Ef(Q, x(\tau_n), w(\tau_n)) + \sum_{n=0}^{N-1} Eg(Q, w(\tau_n)).$$

By telescoping, and using (12) and Lemma 4.ii, we have

$$\begin{aligned} Ex^T(\tau_N)Qx(\tau_N) &= Ex^T(\tau_0)Qx(\tau_0) + \sum_{n=0}^{N-1} Ef(Q, x(\tau_n), w(\tau_n)) + \sum_{n=0}^{N-1} Eg(Q, w(\tau_n)) \\ &\leq x^T(\tau_0)Qx(\tau_0) - \gamma\sum_{n=0}^{N-1} E|x(\tau_n)| + NM. \end{aligned}$$

Since Q is copositive, and $x(\tau_n)$ is nonnegative, it follows that the left hand side above is nonnegative. Hence (11) holds, since by by dividing by $N\gamma$,

$$\frac{1}{N}\sum_{n=0}^{N-1} E|x(\tau_n)| \leq \frac{M}{\gamma} + \frac{1}{N\gamma}x^T(\tau_0)Qx(\tau_0) \text{ for all } N. \square$$

Thus we would like to have a procedure by which we can determine whether (12) holds for some copositive matrix Q.

Lemma 4: Upper bound on f for non-idling policies. *Consider a non-idling policy. Define* $a^+ := \max(a,0)$ *and* $q_{L+1,j} := 0$. *Then,*

$$\begin{aligned} f(Q,x(\tau_n),w(\tau_n)) \quad &\leq \quad 2\sum_{j=1}^{L} x_j(\tau_n)\left[\lambda q_{1j} + \max_{\{i:\sigma(i)=\sigma(j)\}} \mu_i(q_{i+1,j}-q_{ij})\right. \\ &\qquad \left. + \sum_{\{\sigma:\sigma\neq\sigma(j)\}} \max_{\{i:\sigma(i)=\sigma\}} \mu_i(q_{i+1,j}-q_{ij})^+\right]. \end{aligned}$$

Proof By the non-idling property (5), since one of the $w_i(\tau_n)$s with $\sigma(i)=\sigma(j)$ is 1 whenever $x_j(\tau_n)\geq 1$, we see that

$$\sum_{\{i:\sigma(i)=\sigma(j)\}} a_i w_i(\tau_n) x_i(\tau_n) \leq \max_{\{i:\sigma(i)=\sigma(j)\}} a_i x_i(\tau_n) \text{ if } x_j(\tau_n)\geq 1. \tag{13}$$

In addition, since $0\leq \sum_{\{i:\sigma(i)=\sigma\}} w_i(\tau_n)\leq 1$, one always has for every $\sigma\neq\sigma(j)$,

$$\sum_{\{i:\sigma(i)=\sigma\}} a_i w_i(\tau_n) x_j(\tau_n) \leq \max_{\{i:\sigma(i)=\sigma\}} a_i^+ x_j(\tau_n). \tag{14}$$

Now by grouping all indices in (9) corresponding to a specific server together, we have

$$\begin{aligned} f(Q,x(\tau_n),w(\tau_n)) \quad &= \quad 2\textstyle\sum_{j=1}^{L} x_j(\tau_n)\left[\lambda q_{1j} + \sum_{\{i:\sigma(i)=\sigma(j)\}} \mu_i(q_{i+1,j}-q_{ij}) w_i(\tau_n)\right. \\ &\qquad \left. \sum_{\{\sigma:\sigma\neq\sigma(j)\}} \sum_{\{i:\sigma(i)=\sigma\}} \mu_i(q_{i+1,j}-q_{ij}) w_i(\tau_n)\right]. \end{aligned} \tag{15}$$

The bound now follows from (13) and (14). □

Theorem 2: A linear programming sufficient condition for stability of *all* non-idling policies.

(i) *Suppose there exists a copositive matrix* Q $(=Q^T)$ *such that*

$$\lambda q_{1j} + \max_{\{i:\sigma(i)=\sigma(j)\}} \mu_i(q_{i+1,j}-q_{i,j}) + \sum_{\{\sigma:\sigma\neq\sigma(j)\}} \max_{\{i:\sigma(i)=\sigma\}} \mu_i(q_{i+1,,j}-q_{i,j})^+ \leq -1. \tag{16}$$

Then all non-idling scheduling polices are stable. (We define $q_{L+1,j}:=0$*).*

(ii) *If* $\{q_{ij}: 1\leq i,j\leq L\}$, $\{r_j: 1\leq j\leq L\}$ *and* $\{s_{\sigma,j}:$ *for all* $\sigma\neq\sigma(j)\}$ *satisfy the following linear inequalities, then all non-idling scheduling policies are stable:*

$$\lambda q_{1j} + \sum_{\sigma} s_{\sigma,j} \leq 1 \quad \textit{for all} \quad j \tag{17}$$

$$s_{\sigma(j),j} \geq \mu_i(q_{i+1,j}-q_{i,j}) \quad \textit{for all} \quad i \textit{ with } \sigma(i)=\sigma(j) \tag{18}$$

$$s_{\sigma,j} \geq \mu_i(q_{i+1,j} - q_{ij}) \quad \textit{for all} \quad \sigma \neq \sigma(j), \textit{ and all } i \textit{ with } \sigma(i) = \sigma \tag{19}$$

$$s_{\sigma,j} \geq 0 \quad \textit{for all} \quad \sigma \neq \sigma(j) \tag{20}$$

$$q_{ij} = q_{ji} \quad \textit{for all} \quad i,j \tag{21}$$

$$q_{ij} \geq 0 \quad \textit{for all} \quad i,j \tag{22}$$

$$q_{L+1,j} = 0 \quad \textit{for all} \quad j. \tag{23}$$

Proof The claim (i) follows from Theorem 1 and Lemma 4. For (ii), one only needs to observe that (21) and (22) guarantee that Q is a copositive matrix, and hence copositive. □

Thus, we see that the *linear program* (17-23) can be used to check the sufficient condition for the stability of *all* non-idling policies.

Suppose instead that we only wish to establish that a *particular* buffer priority policy θ is stable. In (15) we need to upper bound the term in [....] by (-1) only when $x_j(\tau_n) \geq 1$. But then, from (7), $w_i(\tau_n) = 0$ for all i with $\sigma(i) = \sigma(j)$ and $\theta^{-1}(j) < \theta^{-1}(i)$. Hence one can restrict the first maximum on the RHS in (15) to only those is with $\sigma(i) = \sigma(j)$ and $\theta^{-1}(i) \leq \theta^{-1}(j)$. This gives the following sufficient condition for stability of θ.

Theorem 3: A linear programming sufficient condition for stability of a buffer priority policy θ.

(i) The buffer priority policy θ is stable if there exists a copositive matrix Q such that

$$\lambda q_{1j} + \max_{\substack{\{i:\sigma(i)=\sigma(j) \\ and\ \theta^{-1}(i)\leq\theta^{-1}(j)\}}} \mu_i(q_{i+1,j} - q_{ij}) + \sum_{\{\sigma\neq\sigma(j)\}} \max_{\{i:\sigma(i)=\sigma\}} \mu_i(q_{i+1,j} - q_{ij})^+ \leq -1 \text{ (with } q_{L+1,j} := 0).$$

(ii) Let us modify the linear inequalities (17-23) by changing (18) to:

$$s_{\sigma(j),j} \geq \mu_i(q_{i+1,j} - q_{ij}) \text{ for all } i \text{ with } \sigma(i) = \sigma(j) \text{ and } \theta^{-1}(i) \leq \theta^{-1}(j). \tag{24}$$

If the modified linear program (17,24,19,20,21,22,23) is feasible, then the buffer priority policy θ is stable.

Example 2: LBFS is stable

Consider the system shown in Figure 4, with $\mu_1 = \mu_3$, and operating under the Last Buffer First Serve buffer priority policy. One needs to check whether

$$\lambda q_{1j} + \max_{\substack{\{i:\sigma(i)=\sigma(j) \\ and\ i\geq j\}}} \mu_i(q_{i+1,j} - q_{i,j}) + \sum_{\{\sigma\neq\sigma(j)\}} \max_{\{i:\sigma(i)=\sigma\}} \mu_i(q_{i+1,j} - q_{ij})^+ \leq -\gamma \tag{25}$$

has a feasible symmetric nonnegative solution $[q_{ij}]$ when $\gamma = 1$. Let us determine the largest $\gamma \leq 1$ for which (25) is feasible. This gives the linear program:

$$\text{Max } \gamma$$

subject to

$$\begin{aligned}
\lambda q_{11} + \max\{\mu_1(q_{12} - q_{11}), -\mu_3 q_{13}\} + \max\{\mu_2(q_{13} - q_{12}), 0\} + \gamma &\leq 0 \\
\lambda q_{12} + \mu_2(q_{23} - q_{22}) + \max\{\mu_1(q_{22} - q_{12}\}, -\mu_3 q_{23}, 0\} + \gamma &\leq 0 \\
\lambda q_{13} - \mu_3 q_{33} + \max\{\mu_2(q_{33} - q_{23}), 0\} + \gamma &\leq 0 \\
\gamma &\leq 1 \\
q_{11}, q_{12}, q_{13}, q_{22}, q_{23}, q_{33} &\geq 0.
\end{aligned}$$

Note that $Q = 0$ always gives a feasible solution for $\gamma = 0$. If there is a feasible solution (Q, γ), with $\gamma > 0$, then $\left(\frac{Q}{\gamma}, 1\right)$ is also a feasible solution. Hence the value of the above linear program is either 0 or 1. If it is 0, one can draw no conclusion about the stability of LBFS from this test. On the other hand, if the value of the LP is 1, then LBFS is stable.

Define ρ_1 = nominal load on Server 1 $= \frac{2\lambda}{\mu_1}$, and ρ_2 = nominal load on Server 2 $= \frac{\lambda}{\mu_2}$. Figure 5 plots the value of the LP as a function of (ρ_1, ρ_2). As can be seen, the LP has value 1 throughout, and so LBFS is stable for all (ρ_1, ρ_2) with $0 \leq \rho_i \leq 1$. □

The formulae developed here for establishing stability are easily extended to systems with multiple arrival streams, and random routing (as in Jackson networks); see Kumar and Meyn [11]. Instead of restricting attention to nonnegative q_{ij}s, as in (22), one can simply search for a Q with sign indefinite q_{ij}s. If a feasible solution exists, then one can subsequently check to see whether it is copositive, by using known algorithms (see [11]). On the other hand if Q is not copositive, then slight perturbations of the policy lead to an unstable system, as shown in Kumar and Meyn [14]. For stationary scheduling policies, our sufficient condition for stability actually yields the existence of an exponential moment, and geometric convergence; see [11]. Recently, Down and Meyn [15] have extended the test over quadratic Lyapunov functions given here to piecewise linear Lyapunov functions. A distinctly different and powerful approach to stability, consisting of examining the fluid limits of the stochastic processes in the network, has been developed by Dai [16]. If a system is proved stable by using the fluid limit approach, then the stability also holds for other distributions of interarrival and service times, with some mild restrictions; see Chen [17]. Using this approach Dai has established the stability of the FBFS policy. The stability of the LBFS policy

is established in Dai and Weiss [18] and Kumar and Kumar [19]. The Fluctuation Smoothing Policy for Mean Cycle-Time, developed for scheduling re-entrant lines in Lu, Ramaswamy and Kumar [20], is proved to be stable in Kumar and Kumar [21].

8 Equalities constraining performance

Stability is a yes or no question. Once stability has been established, one often wishes to address more fine-grained questions dealing with the level of performance that one can extract from the system. For specificity, we will focus on bounds on delay or response time. Given stability, Little's Theorem shows that the mean delay of customers is proportional to the mean number of customers in the system. Thus we shall seek bounds on the mean number of customers in the system.

We restrict attention to stationary non-idling scheduling policies. Suppose that the system is started in steady–state at time $\tau_0 = 0$. Denote $Ex_j(\tau_n)$ by x_j. Let us define the variables

$$z_{ij} := E[w_i(\tau_n)x_j(\tau_n)]. \tag{26}$$

From (6) we see that the mean number in buffer b_j can be written as,

$$x_j = \sum_{\{i:\sigma(i)=\sigma(j)\}} z_{ij}. \tag{27}$$

We thus wish to study the variables $\{z_{ij} : 1 \le i, j \le L\}$. Recall the definitions of $f(\cdot)$ and $g(\cdot)$ in Lemma 1.

Lemma 5: Linear equality constraints on $\{z_{ij}\}$.

(i) *For a stable, stationary scheduling policy,*

$$Ef(Q, x(\tau_n), w(\tau_n)) + Eg(Q, w(\tau_n)) = 0 \text{ for every matrix } Q. \tag{28}$$

(ii) *For a stable, stationary scheduling policy, the variables $\{z_{ij}\}$ satisfy the following* equality constraints*:*

$$\begin{aligned}
2\lambda x_1 + 2\lambda - 2\mu_1 z_{11} &= 0 \\
\lambda x_2 - \lambda - \mu_1 z_{12} + \mu_1 z_{11} - \mu_2 z_{21} &= 0 \\
\lambda x_j - \mu_1 z_{1j} - \mu_j z_{j1} + \mu_{j-1} z_{j-1,1} &= 0 \text{ for } j \ge 3 \\
2\mu_{j-1} z_{j-1,j} + 2\lambda - 2\mu_j z_{jj} &= 0 \text{ for } j \ge 2 \\
\mu_{i-1} z_{i-1,i+1} - \mu_i z_{i,i+1} - \lambda + \mu_i z_{ii} - \mu_{i+1} z_{i+1,i} &= 0 \text{ for } 2 \le i \le 1-1 \\
\mu_{i-1} z_{i-1,j} - \mu_i z_{ij} + \mu_{j-1} z_{j-1,i} - \mu_j z_{ji} &= 0 \text{ for } 2 \le i \le L-2 \text{ and } j \ge i+2.
\end{aligned}$$

Proof

(i) Let us employ the slightly stronger assumption that the network has a finite *second* moment in steady–state. Then $E[x_i(\tau_{n+1})x_j(\tau_{n+1})] - E[x_i(\tau_n)x_j(\tau_n)] = 0$ for all i, j. Hence $E[x^T(\tau_{n+1})Qx(\tau_{n+1})] = E[x^T(\tau_n)Qx(\tau_n)]$ for any symmetric matrix Q. Using this in (12) of Lemma (1.i) gives the desired result. In Kumar and Meyn [14] the same result is shown to hold even when the network only has a finite *first* moment.

(ii) Since (28) holds for all Q, it follows that $\frac{\partial}{\partial q_{ij}}\{Ef(Q, x(\tau_n), w(\tau_n)) + Eg(Q, w(\tau_n))\} = 0$ for all i, j. The result then follows by simple calculation using the formulae for f and g. These constraints are obtained independently in Bertsimas, Paschalidis and Tsitsiklis [22] and Kumar and Kumar [23]. □

9 Non-idling and nonnegativity constraints

To obtain upper and lower bounds on the mean number in the system, one would like to maximize or minimize $\sum_{j=1}^{L} x_j$ subject to the linear equality constraints of Lemma (5.ii) and (27). This gives rise to a linear program. Clearly, to obtain better bounds, we would like to append as many other constraints on $[z_{ij}]$ as we can discover, to the linear program.

Clearly, one has the *nonnegativity* constraints

$$z_{ij} \geq 0 \text{ for } 1 \leq i, j \leq L. \tag{29}$$

In addition, for any Server σ, $\sum_{\{i:\sigma(i)=\sigma\}} w_i(\tau_n) \leq 1$. Multiplying both sides by $x_j(\tau_n)$ and taking expectations, we obtain

$$\sum_{\{i:\sigma(i)=\sigma\}} z_{ij} \leq x_j \text{ for all } j, \sigma. \tag{30}$$

As we have seen earlier in (27), equality holds above when $\sigma = \sigma(j)$, due to the non-idling policy. We will call (30,27) the *non–idling inequality constraints.*

10 Performance bounds for all non-idling policies

From Sections 8 and 9, we obtain the following bounds on performance for the entire class of non-idling stationary policies.

Theorem 4: Performance bounds for all non-idling policies. *An upper bound on the mean number in the system under any non-idling stationary policy is the value of the linear program: Max* $\sum_{j=1}^{L} x_j$,

subject to the constraints (27,29,30) and those of Lemma 5.ii. A lower bound on the mean number in the system under any non–idling stationary policy is the value of the above linear program with a "Min" replacing the "Max."

Example 3

Consider the system shown in Figure 6. Let $\mu_1 = \mu_2 = \mu_3 = \mu_4 = 2$. Then the nominal load on each server is $\rho := \lambda$. A lower bound on the mean delay of all stationary non–idling scheduling policies is furnished by the linear program:

$$\text{Min} \frac{1}{\rho}(x_1 + x_2 + x_3 + x_4)$$

subject to: $x_1 = z_{11} + z_{31}$, $x_2 = z_{22} + z_{42}$, $x_3 = z_{13} + z_{33}$, $x_4 = z_{14} + z_{44}$, $\rho + \rho x_1 - 2z_{11} = 0$, $\rho x_2 - \rho - 2z_{12} + 2z_{11} - 2z_{21} = 0$, $\rho x_3 - 2z_{13} + 2z_{21} - 2z_{31} = 0$, $\rho x_4 - 2z_{14} + 2z_{31} - 2z_{41} = 0$, $\rho + 2z_{12} - 2z_{22} = 0$, $2z_{13} - 2z_{23} - \rho + 2z_{22} - 2z_{32} = 0$, $z_{14} - z_{24} + z_{32} - z_{42} = 0$, $\rho + 2z_{23} - 2z_{33} = 0$, $2z_{24} - 2z_{34} + 2z_{33} - \rho - 2z_{43} = 0$, $\rho + 2z_{34} - 2z_{44} = 0$, $z_{21} + z_{41} \leq x_1$, $z_{12} + z_{32} \leq x_2$, $z_{23} + z_{43} \leq x_3$, $z_{14} + z_{34} \leq x_4$, $z_{ij} \geq 0, x_i \geq 0$. Figure 7 plots the lower bound on mean delay for $0 \leq \rho < 1$. Figure 8 plots the lower bound on the normalized value, $(1 - \rho)$ Mean Delay. As can be seen, in heavy traffic no scheduling policy can attain a delay less than $\frac{3}{4(1-\rho)}$. □

11 Additional constraints for specific systems

Frequently, structural properties of a specific scheduling policy or a specific system provide additional constraints which can be appended to the linear programs to obtain better upper and lower bounds. We illustrate this for buffer priority policies, closed systems, and when modeling non-exponentially distributed interarrival or service times by the "method of stages."

11.1 Buffer priority policies

Consider a specific buffer priority policy θ. Then, if $x_j(\tau_n) \geq 1$, server $\sigma(j)$ cannot work on any buffer with lower priority than b_j. Hence, if $x_j(\tau_n) \geq 1$, then $w_i(\tau_n) = 0$ for all i with $\sigma(i) = \sigma(j)$ and $\theta^{-1}(i) > \theta^{-1}(j)$. Thus, from (26),

$$z_{ij} = 0 \text{ for all } i \text{ with } \sigma(i) = \sigma(j) \text{ and } \theta^{-1}(i) > \theta^{-1}(j). \tag{31}$$

Example 4: The last buffer first serve policy

Consider the same system as in Example 3, shown in Figure 6. To obtain lower or upper bounds on the mean delay under LBFS, we append the constraints

$$z_{14} = 0 \text{ and } z_{23} = 0$$

to the constraints shown in Example 3. Figure 9 plots the upper bound on delay under LBFS. Figure 10 plots the value of the ratio

$$\frac{\text{Upper Bound on LBFS Delay}}{\text{Lower Bound on Delay for All Non–idling Stationary Scheduling Policies}},$$

where the value of the denominator is obtained from Example 3. We see that even in heavy traffic, LBFS has a delay no more than 10/3 times the optimal delay, while it is nearly optimal in light traffic. □

11.2 Bounds on the throughput of closed networks

Consider a closed queueing network with N trapped customers. The system is as described in Section 3, except that there are no exogenous arrivals, and customers leaving b_L *return* to b_1. A key question concerning such systems is what is the throughput λ. Note that $E[w_i(\tau_n)] = \frac{\lambda}{\mu_i}$ for all i.

Assume that the system is in steady–state. Then one has

$$E[x^T(\tau_{n+1})Qx(\tau_{n+1})] - E[x^T(\tau_n)Qx(\tau_n)] = 0 \text{ for all matrices } Q.$$

By setting the partial derivatives to zero, i.e.,

$$\frac{\partial}{\partial q_{ij}}\{E[x^T(\tau_{n+1})Qx(\tau_{n+1})] - E[x^T(\tau_n)Qx(\tau_n)]\} = 0$$

one obtains the following equality constraints: $2\mu_{j-1}z_{j-1,j} + 2\lambda - 2\mu_j z_{j,j} = 0$ for all j, $\mu_{i-1}z_{i-1,i+1} - \mu_i z_{i,i+1} - \lambda + \mu_i z_{ii} - \mu_{i+1}z_{i+1,i} = 0$ for all i, $\mu_{i-1}z_{i-1,j} - \mu_i z_{ij} + \mu_{j-1}z_{j-1,i} - \mu_j z_{ji} = 0$ for all i and $j \neq i+1$. Above, we identify the subscript $(L+1)$ with 1.

The nonnegativity and non-idling constraints are the same as before: $z_{ij} \geq 0$, and $\sum_{\{i:\sigma(i)=\sigma\}} z_{ij} \leq \sum_{\{i:\sigma(i)=\sigma(j)\}} z_{ij}$ for all $\sigma \neq \sigma(j)$ and all j.

The first *additional* constraint for a closed network is that the total number of trapped customers is N:

$$\sum_{j=1}^{L} \sum_{\{i:\sigma(i)=\sigma(j)\}} z_{ij} = N.$$

The second additional constraint is more subtle. Since $E[w_i(\tau_n)] = \frac{\lambda}{\mu_i}$, and since $\sum_{j=1}^{L} x_j(\tau_n) = N$, we obtain

$$\sum_{j=1}^{L} z_{ij} = N\frac{\lambda}{\mu_i} \text{ for all } i.$$

If one is analyzing a specific buffer priority policy θ, then one has the buffer priority constraints (31).

Example 5: Throughput bounds for a closed network

Consider the closed network shown in Figure 11, operating under the buffer priority policy $\theta = \{1, 3, 4, 2\}$. Let $\mu_1 = 1/3, \mu_2 = 2/7, \mu_3 = 1, \mu_4 = 2$. For $N = 20$, the throughput lies in the interval $[0.238095, 0.240343]$. When the number of trapped customers is increased to $N = 50$, the throughout lies in the interval $[0.247525, 0.248007]$. □

11.3 Unreliable servers

In many systems, servers are unreliable. For example, in manufacturing systems machines often fail and have to be repaired before they can resume processing of parts. The following examples shows how to model and analyze such systems.

Example 6: Unreliable servers

Consider the system shown in Figure 12. Customers arrive as a Poisson process of rate λ. They visit buffers b_1, b_2 and b_3 before leaving the system. The service times at b_1, b_2 and b_3 are all exponentially distributed with mean values $\frac{1}{\mu_1}$, $\frac{1}{\mu_2}$ and $\frac{1}{\mu_3}$, respectively. The LBFS buffer priority policy $\theta = \{3, 2, 1\}$ is used. Server 1 fails randomly. The time to failure and time to repair are also exponentially distributed, with mean values $\frac{1}{\mu_0}$ and $\frac{1}{\mu_4}$, respectively.

The occasional failure of Server 1 is modeled by introducing a "high priority" customer. This high priority customer is either located at b_4 or at a buffer b_0 located at another Server 0. Whenever the high priority customer is at b_4, Server 1 can only work on it; thus b_4 has higher priority over b_3 and b_1. After completing service at b_4, which corresponds to repair of Server 1, the high priority customer moves to b_0, before returning to b_4. The closed subnetwork $\{b_4, b_0\}$ therefore always contains one customer. The service rates at b_4 and b_0 are μ_4 and μ_0, respectively.

The additional constraints due to the priority of the high priority customer are:

$$z_{14} = 0 \text{ and } z_{34} = 0.$$

Since there is only one customer trapped in the loop $\{b_4, b_0\}$, we have

$$z_{40} = 0, z_{04} = 0 \text{ and } z_{00} + z_{04} = 1.$$

Since either b_4 or b_0 is always working, one has $w_4(\tau_n) + w_0(\tau_n) = 1$. Hence we obtain

$$z_{02} + z_{42} = z_{22}, z_{01} + z_{41} = z_{11} + z_{31} + z_{41} \text{ and } z_{03} + z_{43} = z_{13} + z_{33} + z_{43}.$$

The remaining equality constraints, nonnegativity constraints and non-idling constraints are obtained by following the same procedure as earlier, with slight algebraic differences due to the presence of *two* subnetworks in the system. The equality constraints are, $2\lambda(z_{11} + z_{31} + z_{41}) + 2\lambda - 2\mu_1 z_{11} = 0$, $2\mu_1 z_{12} - 2\mu_2 z_{22} + 2\lambda = 0$, $2\mu_2 z_{23} - 2\mu_3 z_{33} + 2\lambda = 0$, $2\mu_0 z_{04} + 2\mu_0\mu_4/(\mu_0 + \mu_4) - 2\mu_4 z_{44} = 0$, $2\mu_4 z_{40} + 2\mu_0\mu_4/(\mu_0 + \mu_4) - 2\mu_0 z_{00} = 0$, $\lambda z_{22} + \mu_1 z_{11} - \mu_1 z_{12} - \lambda - \mu_2 z_{21} = 0$, $\lambda(z_{13} + z_{33} + z_{43}) - \mu_1 z_{13} - \mu_3 z_{31} + \mu_2 z_{21} = 0$, $\lambda(z_{14}+z_{34}+z_{44})-\mu_1 z_{14}-\mu_4 z_{41}+\mu_0 z_{01} = 0$, $\lambda z_{00}-\mu_1 z_{10}-\mu_0 z_{01}+\mu_4 z_{41} = 0$, $\mu_1 z_{13}+\mu_2 z_{22}-\mu_2 z_{23}-\lambda-\mu_3 z_{32} = 0$, $\mu_1 z_{14} - \mu_2 z_{24} + \mu_0 z_{02} - \mu_4 z_{42} = 0$, $\mu_1 z_{10} - \mu_2 z_{20} + \mu_4 z_{42} - \mu_0 z_{02} = 0$, $\mu_2 z_{24} - \mu_3 z_{34} + \mu_0 z_{03} - \mu_4 z_{43} = 0$, $\mu_2 z_{20} - \mu_3 z_{30} + \mu_4 z_{43} - \mu_0 z_{03} = 0$ and $\mu_0 z_{00} - \mu_0 z_{04} - 2\mu_0\mu_4/(\mu_0+\mu_4) + \mu_4 z_{44} - \mu_4 z_{40} = 0$. The non–idling inequality constraints are: $z_{11} + z_{31} + z_{41} \geq z_{21}$, $z_{11} + z_{31} + z_{41} \geq z_{01}$, $z_{22} \geq z_{12} + z_{32} + z_{42}$, $z_{22} \geq z_{02}$, $z_{44} \geq z_{24}$, $z_{44} \geq z_{04}$, $z_{33} + z_{43} \geq z_{23}$, $z_{33} + z_{43} \geq z_{03}$, $z_{00} \geq z_{20}$ and $z_{00} \geq z_{10} + z_{30} + z_{40}$.

Consider $\mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu$, and $\lambda = \mu_4$, subject to the normalization $\lambda+\mu_0+\mu_1+\mu_2+\mu_3+\mu_4 = 1$. As the ratio μ_0/μ_4 = Mean Time to Repair/Mean Time to Failure is varied, one obtains the lower and upper bounds on the mean number in system shown in Figure 13. □

11.4 Modelling general interarrival and service distributions by the method of stages

In many networks, interarrival and service times are not exponentially distributed. When the distributions have rational Laplace transforms, one can model them by the "method of stages;" see Kleinrock [24]. We illustrate how to analyze such systems in the following example.

Example 7: The $E_2/M/1$ queue with Erlang arrivals

Consider the system shown in Figure 14. Customers arrive randomly to buffer b_3, with an interarrival distribution that is the sum of two independent exponentially distributed random variables with means $\frac{1}{\lambda_1}$, and $\frac{1}{\lambda_2}$. This is modelled by the subnetwork on the left in Figure 14. After leaving b_2, a customer splits into two customers, one travels to b_3 while one returns to b_1. There is one trapped customer in the closed subnetwork $\{b_1, b_2\}$. Since there is one customer in either b_1 or b_2, we have

$$z_{12} = 0, z_{21} = 0.$$

Since Server 0 is always working on either b_1 or b_2, we have $w_1(\tau_1) + w_2(\tau_n) = 1$. Hence

$$z_{13} + z_{23} = z_{33}.$$

The equality constraints are: $\lambda_2 z_{21} - \lambda_1 z_{11} + \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = 0$, $-\lambda_1 z_{12} + \lambda_1 z_{11} - \lambda_2 z_{21} + \lambda_2 z_{22} - 2\frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = 0$, $\lambda_1 z_{12} - \lambda_2 z_{22} + \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = 0$, $\lambda_2 z_{23} - \mu z_{33} + \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = 0$, $\lambda_2 z_{22} - \lambda_2 z_{23} + \lambda_1 z_{13} - \mu z_{32} - \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = 0$, and $-\lambda_1 z_{13} + \lambda_2 z_{23} + \lambda_2 z_{21} - \mu z_{31} + \frac{\lambda_1\lambda_2}{\lambda_1+\lambda_2} = 0$. The non–idling constraints are, $z_{13} + z_{23} \leq z_{33}$, $z_{31} \leq z_{11}$ and $z_{32} \leq z_{22}$. The non-negativity constraints are $z_{ij} \geq 0$ for all i, j.

The nominal load on Server 1 is $\rho := \frac{\lambda_1\lambda_2}{\mu(\lambda_1+\lambda_2)}$. We take $\lambda_2 \leq \lambda_1$. The linear programs for obtaining upper and lower bounds on the number of customers at b_3 can be *explicitly* solved

Theorem 5: Bounds for $E_2/M/1$ queue.

(i) The mean number in the above $E_2/M/1$ queue is bounded below by $z_{33,min}$ and above by $z_{33,max}$ where:

$$\begin{aligned} z_{33,min} &:= \text{Max } \{\rho, \frac{\mu\rho^2}{\lambda_2(1-\rho)}, \frac{\rho}{1-\rho} - \frac{\mu^2\rho^2}{\lambda_1\lambda_2(1-\rho)}\} \\ &= \rho \quad \textit{if } \lambda_1 \leq \mu, \\ &= \frac{\mu\rho^2}{\lambda_2(1-\rho)} \quad \textit{if } \lambda_2 \leq \mu \leq \lambda_1, \\ &= \frac{\rho}{1-\rho} - \frac{\mu^2\rho^2}{\lambda_1\lambda_2(1-\rho)} \quad \textit{if } \mu \leq \lambda_2, \end{aligned}$$

$$\begin{aligned} z_{33,max} &:= \text{Min } \{\frac{\rho}{1-\rho}, \frac{\rho}{1-\rho} - \frac{\mu^2\rho^2}{\lambda_1\lambda_2(1-\rho)} + \frac{\lambda_2}{\lambda_1+\lambda_2}\} \\ &= \frac{\rho}{1-\rho} \quad \textit{if } \lambda_1 \leq \mu, \\ &= \frac{\rho}{1-\rho} - \frac{\mu^2\rho^2}{\lambda_1\lambda_2(1-\rho)} + \frac{\lambda_2}{\lambda_1+\lambda_2} \quad \textit{if } \mu \leq \lambda_1. \end{aligned}$$

(ii) The upper and lower bounds differ by no more than $\frac{\lambda_2}{\lambda_1+\lambda_2} \leq 1/2$.

(iii) The upper bound is always tighter by at least $\frac{\rho}{2}$ than Kingman's upper bound for the mean sojourn time in a $GI/GI/1$ queue, $\frac{\sigma_a^2+\sigma_b^2}{2\bar{t}(1-\rho)} + \bar{x}$ (see [25]), where σ_a^2, σ_b^2, $\bar{t}$, and $\bar{x}$ are the variance of the inter–arrival time, the variance of the service time, the mean inter–arrival time, and the mean service time, respectively.

The performance bounds in this section are drawn from Kumar and Kumar [23], to which we refer the reader for additional results and details. For closed queueing networks with two servers, explicit analytical

bounds can be obtained. They are detailed in Jin, Ou and Kumar [26]. These bounds partially validate some of the conjectures on optimal scheduling of such networks in heavy traffic, developed by Harrison and Wein [27]. Bertsimas, Paschalidis and Tsitsiklis [22] have studied higher order constraints. The approaches to stability and performance bounds are duals in the sense of linear programming. This provides additional information on the two problems. These results are developed in Kumar and Meyn [14]. One can also obtain bounds on the transient behavior of a network. These are also provided in [14].

12 Concluding remarks

We have developed procedures for performance evaluation of previously intractable queueing networks. These procedures can be routinely utilized since they only call for linear programming, a well developed computational tool. They should be useful in many technological important fields, including communication networks, computer systems, and manufacturing systems.

References

[1] J. R. Jackson, "Networks of waiting lines," *Mathematics of Operations Research*, vol. 5, pp. 518–521, 1957.

[2] J. R. Jackson, "Jobshop-like queueing systems," *Management Science*, vol. 10, pp. 131–142, 1963.

[3] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios, "Open, closed and mixed networks of queues with different classes of customers," *J. Assoc. Comput. Mach.*, vol. 22, pp. 248–260, April 1975.

[4] F. P. Kelly, *Reversibility and Stochastic Networks*. New York, NY: John Wiley and Sons, 1979.

[5] L. M. Wein, "Scheduling semiconductor wafer fabrication," *IEEE Transactions on Semiconductor Manufacturing*, vol. 1, pp. 115–130, August 1988.

[6] S. H. Lu and P. R. Kumar, "Distributed scheduling based on due dates and buffer priorities," *IEEE Transactions on Automatic Control*, vol. AC-36, pp. 1406–1416, December 1991.

[7] P. R. Kumar and T. I. Seidman, "Dynamic instabilities and stabilization methods in distributed real-time scheduling of manufacturing systems," *IEEE Transactions on Automatic Control*, vol. AC-35, pp. 289–298, March 1990.

[8] M. Bramson, "Instability of FIFO queueing networks." To appear in *Annals of Applied Probability*, 1994.

[9] T. I. Seidman, "First come first serve can be unstable," *IEEE Transactions on Automatic Control*, vol. 39, pp. 2166–2170, October 1994.

[10] P. R. Kumar, "Re–entrant lines," *Queueing Systems: Theory and Applications: Special Issue on Queueing Networks*, vol. 13, pp. 87–110, May 1993.

[11] P. R. Kumar and S. P. Meyn, "Stability of queueing networks and scheduling policies," *IEEE Transactions on Automatic Control*, vol. 40, pp. 251–260, February 1995.

[12] S. Lippman, "Applying a new device in the optimization of exponential queueing systems," *Operations Research*, vol. 23, pp. 687–710, 1975.

[13] S. Karlin and H. M. Taylor, *A First Course in Stochastic Processes*. New York, NY: Academic Press, 1975.

[14] P. R. Kumar and S. Meyn, "Duality and linear programs for stability and performance analysis of queueing networks and scheduling policies," tech. rep., C. S. L., University of Illinois, 1993.

[15] D. G. Down and S. P. Meyn, "Piecewise linear test functions for stability of queueing networks," in *Proceedings of the IEEE 33th Conference on Decision and Control*, (Buena Vista, FL), December 1994. to appear.

[16] J. G. Dai, "On positive Harris recurrence of multiclass queueing networks: A unified approach via fluid limit models," tech. rep., Georgia Institute of Technology, 1993. To appear in *Annals of Applied Probability*.

[17] H. Chen, "Fluid approximations and stability of multiclass queueing networks I: Work conserving disciplines," tech. rep., University of British Columbia, 1993.

[18] J. Dai and G. Weiss, "Stability and instability of fluid models for certain re-entrant lines." Preprint, February 1994.

[19] S. Kumar and P. R. Kumar, "The last buffer first policy is stable for stochastic re–entrant lines," technical report, Coordinated Science Laboratory, University of Illinois, Urbana, IL, 1994.

[20] S. C. H. Lu, D. Ramaswamy, and P. R. Kumar, "Efficient scheduling policies to reduce mean and variance of cycle–time in semiconductor manufacturing plants," *IEEE Transactions on Semiconductor Manufacturing*, vol. 7, pp. 374–385, August 1994.

[21] S. Kumar and P. R. Kumar, "Fluctuation smoothing policies are stable for stochastic re-entrant lines," in *33rd IEEE Proceedings Conference on Decision and Control*, pp. 1476–1480, December 1994.

[22] D. Bertsimas, I. C. Paschalidis, and J. N. Tsitsiklis, "Optimization of multiclass queueing networks: Polyhedral and nonlinear characterizations of achievable performance," *Annals of Applied Probability*, vol. 4, pp. 43–75, 1994.

[23] S. Kumar and P. R. Kumar, "Performance bounds for queueing networks and scheduling policies," *IEEE Transactions on Automatic Control*, vol. AC-39, pp. 1600–1611, August 1994.

[24] L. Kleinrock, *Queueing Systems, Volume 1: Theory*. New York, NY: Wiley–Interscience, 1975.

[25] J. F. C. Kingman, "Inequalities in the theory of queues," *Journal of the Royal Statistical Society, Series B*, vol. 32, pp. 102–110, 1970.

[26] H. Jin, J. Ou, and P. R. Kumar, "The throughput of closed queueing networks–functional bounds, asymptotic loss, efficiency, and the Harrison-Wein conjectures." October 1994.

[27] J. M. Harrison and L. M. Wein, "Scheduling networks of queues: Heavy traffic analysis of a two-station closed network," *Operations Research*, vol. 38, no. 6, pp. 1052–1064, 1990.