Fluid limits and the efficiency of scheduling policies for stochastic closed reentrant lines in heavy traffic *†

Sunil Kumar and P.R. Kumar‡

Abstract

In this paper, we address the behavior of stochastic closed reentrant lines in heavy traffic, i.e., when the population of the resulting closed queueing network increases without bound. We show that the fluid limits of the network can be used to study the asymptotic throughput in the infinite population limit. As applications of this technique, the Last Buffer First Serve (LBFS) scheduling policy, and the Harrison–Wein Balanced policy for two station networks, are shown to be efficient, i.e., that they attain bottleneck throughput in the infinite population limit. The latter policy was earlier shown to be efficient for closed networks in Jin, Ou and Kumar [1] using a different approach.

Key Words: Queueing networks, closed networks, throughput, optimal scheduling, performance evaluation, fluid limits.

1 Introduction

Consider a closed queueing network with a population size N. Typically, the objective is to schedule such networks to maximize the throughput. In this paper we are interested in the heavy traffic behavior, i.e., as $N \to \infty$, of the throughput.

Let $\lambda^u(x_n)$ denote the throughput when the closed network is started with the initial condition x_n, and scheduling policy u is employed. The population size is $|x_n|$. Let λ^*

*Please address all correspondence to the second author.

†The research reported here has been supported in part by the U. S. Army Research Office under Grant No. DAAH 04-95-1-0090, the National Science Foundation under Grant No. ECS-94-05371, and the Joint Services Electronics Program under Contract No. N00014-90-J-1270.

‡Department of Electrical and Computer Engineering, and the Coordinated Science Laboratory, University of Illinois, 1308 West Main Street, Urbana, IL 61801.

denote the maximum throughput sustainable by the network. We say that the scheduling policy u is *efficient* if $\lim_{|x_n|\to\infty} \lambda^u(x_n) = \lambda^*$. Our goal in this paper is to address the efficiency of scheduling policies.

Our approach is via the "fluid limits" of the closed network. These fluid limits (see Dai [2] and Chen and Mandelbaum [3]) are obtained by suitably scaling the queue lengths in the network and applying a functional strong law of large numbers to this scaled network to obtain a limit process. This limit process, called the fluid limit, is not unique. However, every such limit must necessarily obey a set of integral equations.

It is known that by studying such fluid limits one can study the stability of *open* queueing networks, see Dai [2]. It has been shown by Lu and Kumar [4] that the First Buffer First Serve (FBFS) policy, the Last Buffer First Serve (LBFS) policy and the Least Slack (LS) scheduling policy are all stable for open reentrant lines, under a bursty deterministic model. Using the fluid limit approach, Dai [5] has established that the FBFS policy is stable for stochastic reentrant lines, while Dai and Weiss [6] and Kumar and Kumar [7] have established the stability of the LBFS policy. Finally, the stability of all Fluctuation Smoothing Policies for Mean Cycle Time (FSMCT) policies, a special subset of LS policies has been established in Kumar and Kumar [8].

In this paper we employ the fluid limit approach for *closed* queueing networks. By studying the integral equations, we can determine a superset of the fluid limits. We relate the throughputs of the fluid limits to the asymptotic throughput of the reentrant line. We thereby obtain sufficient conditions for the reentrant line to be efficient, i.e., to achieve the maximal achievable throughput in the infinite population limit.

We utilize this approach to establish the efficiency of certain specific scheduling policies. Even though there is no notion of "first" or "last buffer" in a closed reentrant line, let us designate some arbitrary buffer as the "last buffer" and consider the resulting LBFS policy. We establish that all such LBFS policies are efficient. Lest it be regarded that all policies which are stable for open networks are also efficient for closed networks, we note that the FBFS policy is inefficient for a closed network as shown in Harrison and Nguyen [9], even

though it is stable for open networks [5].

For two station closed networks, by studying the relected Brownian motion approximation of the network, Harrison and Wein [10] have devised a buffer priority policy which they conjecture is asymptotically optimal in heavy traffic. We establish the efficiency of this Harrison–Wein policy for two station closed reentrant lines. Such a result for general two station closed systems has been shown earlier by Jin, Ou and Kumar [1] using the very different method of functional bounding by linear programs.

In the next section we describe the model of the closed reentrant line. In Section 3 we discuss the existence of fluid limits, and specify the set of integral equations that these limits satisfy. In Section 4 we define the notion of efficiency, and obtain conditions on the fluid limits which guarantee it. In Section 5 we refine the limit for buffer priority policies. In Section 6 establish the efficiency of all LBFS policies, regardless of network topology, for closed reentrant lines. In Section 7 we establish the efficiency of the Harrison–Wein policy for two station networks. Finally, we provide some examples of networks with inefficient fluid flows in Section 8.

2 The stochastic closed reentrant line model

The network we consider consists of S servers labeled $\{1, 2, \ldots, S\}$. Customers begin processing at buffer b_1, located at server $\sigma(1) \in \{1, \ldots, S\}$. Upon completing service, they proceed to buffer b_2 located at server $\sigma(2) \in \{1, \ldots, S\}$. Let b_L at server $\sigma(L)$ be the last buffer visited. The sequence $\{\sigma(1), \ldots, \sigma(L)\}$ is the *route* of the customer. Since one may have $\sigma(i) = \sigma(j)$ for some pairs i and j with $i \neq j$, we say that the system is a *re-entrant line.* Let us suppose that customers in buffer b_i require i.i.d. service times with mean $1/\mu_i = m_i < \infty$, from server $\sigma(i)$. At the same time that a customer completes service at buffer b_L, a new customer is released into buffer b_1. This is what is termed as a "closed loop" release policy in manufacturing [11,12], and a "window" based admission policy in communication networks [13,14]. The end result is a closed network with a fixed population size, denoted by N. In a closed network, one cannot really talk of an exit or entry point, but, for conve-

nience, we will continue to call b_L as the "last buffer." The goal is to maximize the rate of departures from buffer b_L, which we will call the "throughput" hereafter. For each server $\sigma \in \{1, 2, \ldots, S\}$, define

$$\rho_\sigma := \lambda^* \left(\sum_{\{1 \le k \le L,\ \sigma(k)=\sigma\}} m_k \right),$$

where

$$\lambda^* := \frac{1}{\max_{\sigma \in \{1,2,\ldots,S\}} \left(\sum_{\{1 \le k \le L,\ \sigma(k)=\sigma\}} m_k \right)}. \tag{1}$$

We call ρ_σ the *nominal load* on server σ, and λ^* the *bottleneck or maximal throughput.* We note that the long term rate of departures from b_L can never exceed λ^*. Any server σ achieving the "max" on the RHS of (1) will be called a *bottleneck.* We assume that the scheduling policy employed is *non–idling*; that is, a server cannot be idle when any of its buffers is non–empty, and *stationary*; that is, the policy depends only on the current buffer lengths. If a priority policy is used, we assume that it is preemptive resume.

3 The fluid limits

In this section, we define the fluid limits and describe the integral equations which they must satisfy. With the sole exception that the network is closed rather than open, all the results in this section are the same as in [5]. First we introduce the notation and some preliminaries. Let $Q_k(t)$ denote the queue length, and $v_k(t)$ the residual service time at buffer b_k at time t. Let $D_k(t)$ denote the number of departures from buffer b_k in $[0, t]$. Then, $A_k(t)$, the total number of arrivals to buffer b_k in $[0, t]$, including the initial condition, is given by:

$$A_1(t) = Q_1(0) + D_L(t),$$

$$A_k(t) = Q_k(0) + D_{k-1}(t),\ k = 2, \ldots, L.$$

Immediately, we have

$$Q_k(t) = A_k(t) - D_k(t).$$

Let $T_k(t)$ be the amount of time in $[0, t]$ that $\sigma(k)$ spends working on b_k, and let $B_\sigma(t) := \sum_{\{j:1 \le j \le L,\ \text{and}\ \sigma(j)=\sigma\}} T_j(t)$ be the amount of time server σ was busy in $[0, t]$. Let $I_\sigma(t) :=$

$t - B_\sigma(t)$ denote the idle time of σ. Because of the non–idling assumption, we have

$$\int_0^\infty \left[\left(\sum_{\{j:\, 1 \le j \le L \text{ and } \sigma(j)=\sigma\}} Q_j(t) \right) \wedge 1 \right] dI_\sigma(t) = 0. \tag{2}$$

The state of the system is given by $X(t) := (Q_1(t), \ldots, Q_L(t), v_1(t), \ldots, v_L(t))$. For $x = (q_1, ..., q_L, v_1, ..., v_L)$ we define $|x| := \sum_{k=1}^L (q_k + v_k)$. In the sequel, we shall denote explicit dependence on an initial condition x by adding a superscript x to the variable of interest. For any function f, let

$$\bar{f}^x(t) := \frac{1}{|x|} f^x(|x|t), \text{ for } t \ge 0 \text{ and } x > 0,$$

denote its scaled version. This is the so–called "fluid scaling."

For simplicity, we make the additional assumption that all the service time distributions are exponential. This allows us to ignore the effects of v_k. The state $x(t)$ is then redefined to consists only of the queue lengths $(Q_1(t), \ldots, Q_L(t))$, and $|Q(t)| := \sum_{i=1}^L Q_i(t) \equiv N$.

For each initial condition x_n, we consider the resulting vector process $(\bar{D}^{x_n}, \bar{A}^{x_n}, \bar{Q}^{x_n}, \bar{T}^{x_n}, \bar{I}^{x_n})$ as an element of $\mathcal{D}^d_{\mathcal{R}}[0, \infty)$, the space of $\mathcal{R}^d$-valued right continuous paths with left limits, endowed with the Skorohod topology. Thus we can define the countable collection of $\mathcal{D}^d_{\mathcal{R}}$-valued variables (one for each x_n) on a common probability space. This is the probability space with respect to which all the results in the sequel are stated.

First, we state a result which is about the same as that of Dai [5]. It is the closed network version of Theorem 4.4 given in [5] for open networks.

Theorem 1 *Almost surely, the following holds. For every sequence of initial conditions with $|x_n| \to \infty$, there exists a further subsequence x_{n_l} such that along this subsequence, as $l \to \infty$,*

$$(\bar{D}^{x_{n_l}}, \bar{A}^{x_{n_l}}, \bar{Q}^{x_{n_l}}, \bar{T}^{x_{n_l}}, \bar{I}^{x_{n_l}}) \Rightarrow (\bar{D}, \bar{A}, \bar{Q}, \bar{T}, \bar{I}), \tag{3}$$

where "$\Rightarrow$" denotes uniform convergence on compacts. Furthermore, the limit processes, called the fluid limits*, satisfy:*

$$\bar{D}_k(t) = \mu_k \bar{T}(t), \tag{4}$$

$$\bar{A}_1(t) = q_1 + \bar{D}_L(t), \tag{5}$$

$$\bar{A}_k(t) = q_k + \bar{D}_{k-1}(t),\ k \geq 2, \tag{6}$$

$$\bar{Q}_k(t) = \bar{A}_k(t) - \bar{D}_k(t), \tag{7}$$

$$\bar{I}_\sigma(t) = t - \sum_{\{j:1\leq j\leq L \text{ and } \sigma(j)=\sigma\}} \bar{T}_j(t), \tag{8}$$

$$0 \leq \bar{T}_k(t_1) \leq \bar{T}_k(t_2) \text{ and } 0 \leq \bar{I}_\sigma(t_1) \leq \bar{I}_\sigma(t_2) \text{ if } t_1 \leq t_2, \tag{9}$$

$$|\bar{I}_\sigma(t) - \bar{I}_\sigma(s)| \leq |t-s|,\ |\bar{T}_k(t) - \bar{T}_k(s)| \leq |t-s|, \text{ and } \bar{Q}_k(t) \text{ is also Lipschitz}, \tag{10}$$

$$\int_0^\infty \left(\sum_{\{j:1\leq j\leq L \text{ and } \sigma(j)=\sigma\}} \bar{Q}_j(t) \wedge 1 \right) d\bar{I}_\sigma(t) = 0, \tag{11}$$

$$\text{and } \sum_{k=1}^{L} q_k = 1. \tag{12}$$

Proof The equality (4) follows from the renewal theorem. Also see Lemma 4.1 of [2] to conclude uniform convergence on compacts. Since

$$|\bar{T}_k^{x_n}(t) - \bar{T}_k^{x_n}(s)| \leq |t-s|, \text{ for all } x_n,$$

$\{\bar{T}_k^{x_n}\}$ is relatively compact in $\mathcal{D}_{\mathcal{R}}[0,\infty)$; see Ethier and Kurtz [15], Theorem 6.3, p.123. Therefore there is a convergent subsequence in the Skorohod topology. Since the limit is continuous, it follows that the convergence is uniform on compacts, see [15]. The other quantities, namely $\bar{A}_k^{x_n}(t)$, $\bar{Q}_k^{x_n}(t)$ and $\bar{I}_\sigma^{x_n}(t)$, depend on $\bar{D}_k^{x_n}(t)$ and $\bar{T}_k^{x_n}(t)$ in an affine fashion, and hence contain convergent subsequences by the continuous mapping theorem.

The equations (5–9) follow from the fact that they hold for every sample path of the scaled processes. The result (10) is also immediate. The result (12) follows from the scaling by $|x_n|$. Only (11) needs justification. For this, note that $\left(\sum_{\{j:1\leq j\leq L \text{ and } \sigma(j)=\sigma\}} \bar{Q}_j^{x_n}(t) \wedge 1\right)$ is continuous in $\bar{Q}_j^{x_n}(t)$, and $\bar{I}_\sigma^{x_n}(t)$ is continuous and nondecreasing. So by Lemma 2.4 of Dai and Williams [16],

$$\int_0^t \left(\sum_{\{j:1\leq j\leq L \text{ and } \sigma(j)=\sigma\}} \bar{Q}_j^{x_{n_l}}(s) \wedge 1 \right) d\bar{I}_\sigma^{x_{n_l}}(s) \to \int_0^t \left(\sum_{\{j:1\leq j\leq L \text{ and } \sigma(j)=\sigma\}} \bar{Q}_j(s) \wedge 1 \right) d\bar{I}_\sigma(s),$$

for all t. This and (2) yield (11). $\square$.

It should be noted that the derivatives of the processes $\bar{A}_i$, $\bar{Q}_i$, $\bar{D}_i$ and $\bar{I}_\sigma$ exist at a.e. t. Such times of differentiablity are called "regular times," as in Dai and Weiss [6].

4 Fluid limit sufficient condition for efficiency

In this section we define the notion of "efficiency" and provide sufficient conditions for it in terms of the throughput of the fluid limit.

Given a stationary scheduling policy u and an initial state x, we define the throughput $\lambda^u(x)$ as

$$\lambda^u(x) := \lim_{T\to\infty} \frac{E\left[D^x(T)\right]}{T}.$$

Definition The stationary scheduling policy u[1] is said to be *efficient* if for every sequence of initial conditions x_n with $|x_n| \uparrow \infty$, we have

$$\lim_{x_n} \lambda^u(x_n) = \lambda^* \tag{13}$$

where λ^* is defined in (1).

The next lemma provides a sufficient condition for efficiency based on calculating the limits in the opposite order to (13).

Lemma 1 *If for every sequence x_n with $|x_n| \uparrow \infty$*

$$\liminf_{t\to\infty} \liminf_{x_n} \frac{D_L^{x_n}(|x_n|t)}{|x_n|t} \geq \lambda^* \ a.s., \tag{14}$$

then (13) also holds for every such sequence.

Proof Consider a sequence x_n with $|x_n| \uparrow \infty$. For every x_n, let x'_n be a state with the same population size, and lying in a closed communicating class with the least throughput, i.e.,

$$|x_n| = |x'_n| \text{ and } \lambda^u(x'_n) = \min_{\{y_n:\, |y_n|=|x_n|\}} \lambda^u(y_n). \tag{15}$$

[1]More precisely, there is a sequence of stationary, non-idling scheduling policies, one for each $|x_n| = N$.

Since x'_n is in a closed communicating class,

$$\lim_{T\to\infty} \frac{D^{x'_n}(T)}{T} = \lambda^u(x'_n) \text{ a.s.} \tag{16}$$

Fix ω a sample point in our probability space, outside of a set of zero measure excluded in (14) above. Let $D_L^{x'_n}$ be the sequence of $\mathcal{D}_{\mathcal{R}}$-valued variables corresponding to ω. Then, from (14) above applied to the sequence x'_n, given any x'_n, T and $\epsilon > 0$, we can find x'_{n_l} with $|x'_{n_l}| \geq |x_n|$ and $T' \geq T$ such that,

$$\frac{D_L^{x'_{n_l}}(|x'_{n_l}|T')}{|x'_{n_l}|T'} \geq \lambda^* - \epsilon.$$

So we have

$$\limsup_{x'_n} \limsup_{T\to\infty} \frac{D_L^{x'_n}(T)}{T} \geq \lambda^* \text{ a.s.}$$

From (16), we thus obtain

$$\limsup_{x'_n} \lambda^u(x'_n) \geq \lambda^* \text{ a.s.}$$

Then from (15) we have, for *every* sequence x_n with $|x_n| \uparrow \infty$,

$$\limsup_{x_n} \lambda(x_n) \geq \lambda^*. \tag{17}$$

Hence

$$\liminf_{x_n} \lambda(x_n) \geq \lambda^*, \tag{18}$$

and the result (13) follows since $\lambda(x) \leq \lambda^*$ for every x. □

We are now ready to prove the main sufficient condition for efficiency, which is that all the fluid limits have maximal throughput.

Theorem 2 *Suppose that for every fluid limit $\bar{D}_L(t)$*

$$\liminf_{t\to\infty} \frac{\bar{D}_L(t)}{t} \geq \lambda^* \text{ a.s.} \tag{19}$$

Then (14) holds, and hence the stationary non-idling scheduling policy u is efficient.

Proof Fix ω a sample point in our probability space, outside of the set of zero measure excluded in Lemma 1. Let $D_L^{x_n}$, $|x_n| \uparrow \infty$, be the sequence of $\mathcal{D}_{\mathcal{R}}$-valued variables corresponding to ω. For brevity we will omit the explicit dependence on ω. Let $\bar{D}_L^{x_{n_k}}$ be any convergent subsequence, converging to $\bar{D}_L$. Fix $\epsilon > 0$ arbitrary. Then from (19), $\exists t_1 < \infty$ such that $\forall t > t_1$,

$$\frac{\bar{D}_L(t)}{t} \geq \lambda^* - \epsilon.$$

Moreover, from Theorem 1, for every t, $\exists K(t,\epsilon)$ such that for all $k > K$,

$$\left| \frac{D_L^{x_{n_k}}(|x_{n_k}|t)}{|x_{n_k}|} - \bar{D}_L(t) \right| < \epsilon.$$

Hence,

$$\frac{D_L^{x_{n_k}}(|x_{n_k}|t)}{|x_{n_k}|t} \geq \lambda^* - \epsilon(1 + \frac{1}{t}) \text{ for all } t > t_1 \text{ and } k > K(t,\epsilon). \tag{20}$$

So,

$$\liminf_{x_{n_k}} \frac{D_L^{x_{n_k}}(|x_{n_k}|t)}{|x_{n_k}|t} \geq \lambda^* - \epsilon(1 + \frac{1}{t}).$$

Since this holds for all $t > t_1$, we have

$$\liminf_{t\to\infty} \liminf_{x_{n_k}} \frac{D_L^{x_{n_k}}(|x_{n_k}|t)}{|x_{n_k}|t} \geq \lambda^* - \epsilon.$$

Since ϵ was arbitrary, we have

$$\liminf_{t\to\infty} \liminf_{x_{n_k}} \frac{D_L^{x_{n_k}}(|x_{n_k}|t)}{|x_{n_k}|t} \geq \lambda^*.$$

The above holds for almost all ω, and hence the result follows. □

5 Buffer priority policies

Consider a buffer priority policy u. Note that it is stationary and non-idling, and also, as shown in Jin, Ou and Kumar [1], there is a single communicating class.

We use the notation "$b_k \succ b_l$" if $\sigma(k) = \sigma(l)$, $k \neq l$, and b_k has higher priority than b_l. Let $H_k := \{1 \leq j \leq L \mid b_j \succ b_k\} \cup \{k\}$ denote the set of buffers having at least as high a priority as b_k, and let

$$U_k(t) := t - \sum_{j \in H_k} T_j(t),$$

denote the time *not* spent working on them respectively. The following lemma identifies additional constraints satisfied by the fluid limits of the buffer priority policy u.

Lemma 2 *For every sequence of initial conditions x_n with $|x_n| \uparrow \infty$, there exists a subsequence x_{n_l} such that $\bar{U}_k^{x_{n_l}}(t) \to \bar{U}_k(t)$ in the sense of Theorem 1, i.e., almost sure convergence uniformly on compacts. Furthermore, the fluid limits satisfy the additional constraints:*

$$\bar{U}_k(t) = t - \sum_{j \in H_k} \bar{T}_j(t), \tag{21}$$

$$0 \le \bar{T}_k(t) \le \bar{U}_j(t) \le t \text{ for all } j \in H_k,\ j \ne k \tag{22}$$

$$0 \le \bar{U}_k(t_1) \le \bar{U}_k(t_2) \text{ if } t_1 \le t_2, \tag{23}$$

$$\int_0^\infty \left(\sum_{j \in H_k} \bar{Q}_j(t) \wedge 1 \right) d\bar{U}_k(t) = 0, \tag{24}$$

Proof Identical to that of Theorem 1. □

Now we can provide a sufficient condition based on Theorem 2 which depends only on the solutions to the constraints (4–12) and (21–24).

Theorem 3 *(i) Suppose every solution $\bar{D}_L(t)$ of (4–12, 21–24) satisfies*

$$\liminf_{t \to \infty} \frac{\bar{D}_L(t)}{t} \ge \lambda^*. \tag{25}$$

Then the buffer priority scheduling policy u is efficient.

(ii) If for every solution $\bar{D}_L(t)$ of (4–12, 21–24), there exists a time $T < \infty$ such that

$$\frac{d}{dt}\bar{D}_L(t) \ge \lambda^* \text{ for almost every } t > T, \tag{26}$$

then the buffer priority scheduling policy u is efficient.

Proof The result (i) is immediate from Theorem 2 using the fact that the fluid limits almost surely satisfy (4–12, 21–24). The result (ii) follows by noting that (26) implies (25). □

6 Efficiency of all LBFS policies

In this section, we utilize Theorem 3(ii) to establish that the Last Buffer First Serve (LBFS) policy is efficient for all closed reentrant lines, regardless of network topology.

The LBFS scheduling policy functions as follows. If buffers b_j and b_k share the same server, i.e., $\sigma(j) = \sigma(k)$, and $j < k$, then priority is given to b_k. Therefore if $k < L$, b_k can be worked on by $\sigma(k)$ only if $b_j = 0$ for all $j = k+1, \ldots, L$, with $\sigma(j) = \sigma(k)$. Note that b_L is never preempted. One should note that if a different buffer had been chosen as the "first" buffer b_1, then one obtains a different policy. The following theorem applies to *all* the resulting LBFS policies.

Theorem 4 *Every LBFS policy is efficient.*

Proof We shall prove this using Theorem 3(ii). Let σ^* be a bottleneck server, i.e.,

$$\sum_{\{j:\, 1 \le j \le L \text{ and } \sigma(j) = \sigma^*\}} m_j = \frac{1}{\lambda^*},$$

First, let us assume that σ^* is unique. We will relax this assumption later. Let b_{k^*} be the lowest indexed buffer at σ^*, i.e.,

$$b_{k^*} = \min\{j :\ 1 \le j \le L \text{ and } \sigma(j) = \sigma^*\}.$$

Then we claim that $\exists\, T < \infty$ such that for all $t > T$, $\bar{Q}_k(t) = 0$ for all $k \neq k^*$, and $\bar{Q}_{k^*}(t) = 1$. By Dai and Weiss [6], Prop. 4.2[2], this in turn implies that for all $k = 1, 2, ..., L$, and almost all $t > T$,

$$\frac{d}{dt}\bar{D}_k(t) = \lambda^*,$$

thus establishing (26).

We now prove the claim that there exists $T < \infty$ such that $\bar{Q}_k(t) = 0$ for all $k \neq k^*$ and $\bar{Q}_{k^*}(t) = 1$, for all $t > T$. The arguments in the proof will be very similar to those in Kumar and Kumar [7] and Dai and Weiss [6].

[2] Though [6] deals with fluid limits of open networks, their proposition can be adapted to closed networks with the modification that $d_0(t) = d_L(t)$.

Suppose $k^* = L$. This implies in particular that b_L is the only buffer served by $\sigma(L)$. Consider $X(t) := \sum_{k=1}^{L-1} \bar{Q}_k(t)$ at a regular time t. Note that b_{L-1} has highest priority at $\sigma(L-1)$ since $\sigma(L-1) \neq \sigma(L)$ by assumption. Then, since $\sigma(L)$ is the bottleneck and the only buffer it serves is b_L, for almost all t such that $\bar{Q}_{L-1}(t) > 0$, we have

$$\frac{d}{dt}X(t) \leq \mu_L - \mu_{L-1} < 0.$$

So by Lemma 2.2 of [6], $\exists\, t_{L-1} \leq \frac{1}{\mu_{L-1}-\mu_L}$ such that $Q_{L-1}(t_{L-1}) = 0$. Arguing as in Lemmas 2 and 3 of [7], define

$$k := \max\{i < L-1 \mid \sum_{\{j:\, i \leq j \leq L \text{ and } \sigma(j)=\sigma(i)\}} m_j \geq m_{L-1}\}$$

if the set on the right hand side is non–empty, and $k := 0$ otherwise. As shown in [7], it follows that

$$\bar{Q}_{k+1}(t_{L-1}+\delta) = \cdots = \bar{Q}_{L-1}(t_{L-1}+\delta) = 0, \text{ for all } \delta \geq 0.$$

This argument can be iterated backwards from b_k. First we observe that there exists a time t_k such that $\bar{Q}_k(t_k) = 0$, noting that the input rate to the section $\{b_1, ..., b_k\}$ is no larger than λ^* while the output rate from b_k is $\frac{1}{\sum_{\{i:\, k \leq i \leq L \text{ and } \sigma(i)=\sigma(k)\}} m_i} > \lambda^*$. Then at time t_k we identify a k' such that

$$k' = \max\{j < k | \sum_{\{i:\, j \leq i \leq L \text{ and } \sigma(i)=\sigma(j)\}} m_i \geq \sum_{\{n:\, k \leq n \leq L \text{ and } \sigma(n)=\sigma(k)\}} m_n\}.$$

We argue that $b_{k'+1}, ..., b_k$ are empty at t_k and remain empty thereafter. The arguments are identical to those in [7] and are omitted. Iterating backwards from $b_{k'}$ completes the proof, when we reach b_1, leading to

$$\bar{Q}_1(t_1+\delta) = \cdots = \bar{Q}_{L-1}(t_1+\delta) = 0 \text{ for all } \delta \geq 0. \tag{27}$$

Suppose $k^* < L$. Then it is enough to show that there is a $t_{k^*+1} < \infty$ such that

$$\bar{Q}_{k^*+1}(t) = \cdots = \bar{Q}_L(t) = 0 \text{ for all } t \geq t_{k^*+1}. \tag{28}$$

All we would then have left to do for $t \geq t_{k^*+1}$ is to apply the case when $k^* = L$ to the section $b_1, b_2, ..., b_{k^*}$ with μ_j for $j = 1, 2, ..., k^*$ being replaced by

$$\tilde{\mu}_j := \mu_j \left(1 - \sum_{\{i:\, i>j \text{ and } \sigma(i)=\sigma(j)\}} m_i \lambda^* \right).$$

See (16) of [7] for the idea behind this argument. Now we prove (28). The key idea is the following. Suppose at a regular time τ, for some $j > k^*$, $\bar{Q}_j(\tau) > 0$ and $\bar{Q}_{j+1}(\tau) = \bar{Q}_{j+2}(\tau) = \cdots = \bar{Q}_L(\tau) = 0$. Then under LBFS,

$$\frac{d}{dt}\bar{D}_L(\tau) \geq \frac{1}{\sum_{\{i:j\leq i\leq L \text{ and } \sigma(i)=\sigma(j)\}} m_i} > \lambda^*.$$

Since this exit rate cannot be sustained for almost all $\tau \geq t$, it follows that at some t_{k^*+1},

$$\bar{Q}_{k^*+1}(t_{k^*+1}) = \cdots = \bar{Q}_L(t_{k^*+1}) = 0.$$

Moreover as shown in Lemma 3 of [7], this imples (28).

Let us now relax the assumption that the bottleneck server is unique. Suppose that there are exactly two bottleneck servers. Suppose k_1^* and k_2^* are the lowest priority buffers at bottleneck servers σ_1^* and σ_2^* respectively, and $k_1^* > k_2^*$. Then we proceed as follows, imitating the arguments above. For brevity, we will not repeat the arguments and we will confine ourselves to indicating the sections which empty. The exact same argument as leading to (28) shows that there exists a $t_{k_1^*+1} < \infty$ such that

$$\bar{Q}_{k_1^*+1}(t_{k_1^*+1} + \delta) = \cdots = \bar{Q}_L(t_{k_1^*+1}\delta) = 0 \text{ for all } \delta \geq 0.$$

Now consider the section $b_{k_2^*+1}, ..., b_{k_1^*-1}$. Using the same argument as that leading to (27), we see that there exists a $T_1 < \infty$ such that

$$\bar{Q}_{k_2^*+1}(T_1 + \delta) = \cdots = \bar{Q}_{k_1^*-1}(T_1 + \delta) = 0 \text{ for all } \delta \geq 0.$$

Finally, we consider the section $b_1, ..., b_{k_2^*-1}$. The input rate to this section cannot exceed λ^* after $t_{k_1^*+1}$. Also note that there is no buffer of higher priority than buffers $b_1, ..., , b_{k_2^*-1}$

which is nonempty after T_1. Once again, using the arguments leading to (27), we see that there exists a $T_2 < \infty$ such that

$$\bar{Q}_1(T_2 + \delta) = \cdots = \bar{Q}_{k_2^*-1}(T_2 + \delta) = 0 \text{ for all } \delta \geq 0.$$

Thus we have shown that there exists a $T' < \infty$ such that

$$\sum_{\{k: k \neq k_1^* \text{ and } k \neq k_2^*\}} \bar{Q}_k(T' + \delta) = 0 \text{ for all } \delta \geq 0,$$

and so for all $\delta \geq 0$,

$$\bar{Q}_{k_1^*}(T' + \delta) + \bar{Q}_{k_2^*}(T' + \delta) = 1.$$

By applying Dai and Weiss [6], Prop. 4.2, we obtain the result. The extension to any number of bottleneck servers is immediate. □

7 Efficiency of the Harrison–Wein policy

For closed queueing networks with *two* servers, Harrison and Wein [10] have examined the reflected Brownian motion approximation associated with the heavy traffic scenario, and conjectured that a particular buffer priority policy provides maximal throughput in the infinite population limit. In this section we prove the efficiency of their buffer priority policy for closed reentrant lines with two servers.

To describe their policy, it is convenient to imagine that when a customer leaves b_L, it exits from the system only to be replaced by a new customer in b_1. (As mentioned in Section 2, closed queueing networks arise from such window based admission control strategies). Consider a customer in buffer b_i. Then

$$M_{\sigma,i} := \sum_{\{j: i \leq j \leq L \text{ and } \sigma(j) = \sigma\}} m_j$$

is the mean amount of work on a customer in b_i still remaining to be done by server σ prior to the customer's exit from b_L. Also let $\rho_\sigma := M_{\sigma,1}$ be the relative utilization of server σ. At the first server, $\sigma = 1$, rank the buffers to give higher priority to buffers with *smaller* values

of the index

$$\eta_j := \rho_2 M_{1,j} - \rho_1 M_{2,j}.$$

At the second server ($\sigma = 2$) rank the buffers to give higher priority to buffers with *larger* values of the index η_j. The resulting buffer priority policy is enforced in a preemptive resume fashion and will be called the *Harrison–Wein policy* hereafter. It is worth noting that the priority policy is invariant with respect to choice of the last buffer, unlike LBFS. The following result has been established in Jin, Ou and Kumar [1] for general stochastic reentrant lines using a different approach based on functional bounding by linear programs.

Theorem 5 *The Harrison–Wein policy for two server networks is efficient.*

Proof This proof uses Theorem 3(i). In the sequel we will use the notation

$$d_k(t) := \frac{d}{dt}\bar{D}_k(t),$$

to denote the derivative at the regular time t when the derivative exists.

First we provide a sufficient condition for (25).

Lemma 3 *If*

$$\bar{I}_\sigma(t) = 0 \text{ for all } t, \text{ for some } \sigma \in \{1, 2, ...S\}, \tag{29}$$

for a solution of (4–12, 21–24), i.e., if one of the servers never idles, then the solution satisfies (25).

Proof Suppose $\bar{I}_1(t) = 0$ for all t. This means that $\frac{d}{dt}\bar{I}_1(t) = 0$ at every regular t. This leads us to conclude that at any regular t, we have

$$\sum_{\{k:\sigma(k)=1\}} m_k d_k(t) = 1.$$

That is,

$$\int_0^T \sum_{\{k:\sigma(k)=1\}} m_k d_k(t)\, dt = T \text{ for all } T \geq 0.$$

Now assume (25) does not hold, that is,

$$\liminf_{t\to\infty} \frac{\bar{D}_L(t)}{t} < \lambda^*.$$

This means that for a fixed $\delta > 0$ sufficiently small, for any T, we can find a $T' \geq T$ such that $\bar{D}_L(T') < (\lambda^* - \delta)T'$. But, $\bar{Q}_L(T') = q_L + \bar{D}_{L-1}(T') - \bar{D}_L(T')$ and $\bar{Q}_L(T') \leq 1$. So $\bar{D}_{L-1}(T') \leq (\lambda^* - \delta)T' + 1$. Continuing this argument, using $\bar{Q}_{L-1}(T') + \bar{Q}_L(T') \leq 1$, we get $\bar{D}_{L-2} \leq (\lambda^* - \delta)T' + 1$ and so on, to yield

$$\bar{D}_k(T') \leq (\lambda^* - \delta)T' + 1, \text{ for all } k = 1, 2, ..., L.$$

Also, we have

$$\int_0^{T'} \sum_{\{k:\sigma(k)=1\}} m_k d_k(t)\, dt = \sum_{\{k:\sigma(k)=1\}} m_k D_k(T').$$

Thus, as argued above,

$$T' = \sum_{\{k:\sigma(k)=1\}} m_k D_k(T') \leq \left(\sum_{\{k:\sigma(k)=1\}} m_k \right) \left[(\lambda^* - \delta)T' + 1 \right].$$

But $\sum_{\{k:\sigma(k)=1\}} m_k \lambda^* \leq 1$, and by choosing T' sufficiently large we obtain $\delta T' > 1$, leading to a contradiction. Thus (25) must hold. □

The next result identifies some properties of the priority ordering used in the Harrison–Wein policy.

Lemma 4 *Under the Harrison–Wein policy, the following are true.*

(i) If $\sigma(k) = \sigma((k+1) \bmod L)$ *then* $b_{(k+1) \bmod L} \succ b_k$.

(ii) If $j \leq i-1$, $\sigma(j) = 1$ *and* $b_j \succ b_i$, *then*

$$\sum_{\{k:j\leq k\leq i-1;\sigma(k)=2\}} m_k \geq \frac{\rho_2}{\rho_1} \sum_{\{k:j\leq k\leq i-1;\sigma(k)=1\}} m_k.$$

(iii) If $j \geq i+1$, $\sigma(j) = 1$ *and* $b_j \succ b_i$, *then*

$$\sum_{\{k:j\leq k\leq L;\sigma(k)=2\}} m_k + \sum_{\{k:1\leq k\leq i-1;\sigma(k)=2\}} m_k \geq \frac{\rho_2}{\rho_1} \left[\sum_{\{k:j\leq k\leq L;\sigma(k)=1\}} m_k + \sum_{\{k:1\leq k\leq i-1;\sigma(k)=1\}} m_k \right].$$

Above, empty sums are taken to be zero.

Proof (i) If $\sigma(k) = \sigma(k+1) = 1$ for some $k \in \{1, 2, ..., L-1\}$, then $\eta_k = \eta_{k+1} + \rho_2 m_k > \eta_{k+1}$ and so b_{k+1} always has higher priority than b_k. Similarly, if $\sigma(k) = \sigma(k+1) = 2$, for some $k \in \{1, 2, ..., L-1\}$, then $\eta_k = \eta_{k+1} - \rho_1 m_k < \eta_{k+1}$ and so b_{k+1} again has higher priority than b_k. If $\sigma(1) = \sigma(L) = 1$, then $0 = \eta_1 < \eta_L = \rho_2 m_L$ and so b_1 has higher priority than b_L. The result for the case when $\sigma(1) = \sigma(L) = 2$ is obtained similarly.

(ii) $\sigma(j) = \sigma(i) = 1$ and $b_j \succ b_i$ imply $\eta_j \leq \eta_i$, and so we have

$$\rho_2 \sum_{\{k:j\leq k\leq L;\sigma(k)=1\}} m_k - \rho_1 \sum_{\{k:j\leq k\leq L;\sigma(k)=2\}} m_k \leq \rho_2 \sum_{\{k:i\leq k\leq L;\sigma(k)=1\}} m_k - \rho_1 \sum_{\{k:i\leq k\leq L;\sigma(k)=2\}} m_k l,$$

from which we get

$$\rho_2 \sum_{\{k:j\leq k\leq i-1;\sigma(k)=1\}} m_k \leq \rho_1 \sum_{\{k:j\leq k\leq i-1;\sigma(k)=2\}} m_k.$$

Then (ii) is immediate.

(iii) As before, $\sigma(j) = \sigma(i) = 1$ and $b_j \succ b_i$ imply $\eta_i \geq \eta_j$, and so

$$\rho_2 \sum_{\{k:i\leq k\leq j-1;\sigma(k)=1\}} m_k \geq \rho_1 \sum_{\{k:i\leq k\leq j-1;\sigma(k)=2\}} m_k.$$

Thus,

$$\rho_1\rho_2 - \rho_2 \sum_{\{k:i\leq k\leq j-1;\sigma(k)=1\}} m_k \leq \rho_1\rho_2 - \rho_1 \sum_{\{k:i\leq k\leq j-1;\sigma(k)=2\}} m_k,$$

from which (iii) follows. □

Now we are ready to prove (29). Let us assume without loss of generality that $\rho_1 \leq \rho_2$, i.e, server 1 is faster. Then we claim that $\bar{I}_2(t) = 0$ for all t, for all solutions to (4–12, 21–24).

Suppose not. Then for some regular t, we must have

$$\frac{d}{dt}\bar{I}_2(t) > 0.$$

For this to happen, from (21–24) and Proposition 4.2 of Dai and Weiss [6], we must have $\sum_{\{k:1\leq k\leq L;\sigma(k)=2\}} \bar{Q}_k(t) = 0$ and $d_k(t) = d_{(k-1) \bmod L}(t)$ for all $1 \leq k \leq L$ such that $\sigma(k) = 2$. Necessarily, it must also be true that

$$\sum_{\{k:1\leq k\leq L;\sigma(k)=2\}} m_k d_k(t) < 1.$$

Now, $\sum_{\{k:1\leq k\leq L;\sigma(k)=1\}} \bar{Q}_k(t) = 1$, and so

$$\sum_{\{k:1\leq k\leq L;\sigma(k)=1\}} m_k d_k(t) = 1.$$

Thus,

$$\sum_{\{k:1\leq k\leq L;\sigma(k)=2\}} m_k d_k(t) - \sum_{\{k:1\leq k\leq L;\sigma(k)=1\}} m_k d_k(t) < 0. \tag{30}$$

At the regular time t, let b_{k^*} be the highest priority nonempty buffer at server 1.

If b_{k^*} is the lowest priority buffer at server 1, then we must have $d_j(t) = d_{(j-1) \bmod L}(t)$ for all $j \neq k^*$, since every other buffer in the network must be empty (recall that server 2 is empty). Thus, $d_j(t) = d_{k^*}(t)$ for all j. Hence (30) yields

$$\left(\sum_{\{k:1\leq k\leq L;\sigma(k)=2\}} m_j - \sum_{\{k:1\leq k\leq L;\sigma(k)=1\}} m_j \right) d_{k^*}(t) = (\rho_2 - \rho_1) d_{k^*}(t) < 0,$$

which however cannot hold because $\rho_1 \leq \rho_2$.

Thus b_{k^*} is *not* the lowest priority buffer at server 1. Let b_l be the first buffer at server 1 which is of lower priority than b_{k^*}, encountered as we traverse the network in the order b_{k^*}, b_{k^*+1}, ... wrapping around at b_L if need be. Now there are only two possibilities because of Lemma 4(i), either $l > k^* + 1$ or $l \leq k^* - 1$. In the first case, $l > k^* + 1$, we must have $d_{k^*}(t) = d_{k^*+1}(t) = \cdots = d_{l-1}(t)$; $d_l(t) = d_{l+1}(t) = \cdots = d_L(t) = 0$ and $d_1(t) = d_2(t) = \cdots = d_{k^*-1}(t) = 0$. Thus (30) becomes

$$\left(\sum_{\{j:k^*\leq j\leq l-1;\sigma(j)=2\}} m_j - \sum_{\{j:k^*\leq j\leq l-1;\sigma(j)=1\}} m_j \right) d_{k^*}(t) < 0.$$

But from Lemma 4(ii), since $l > k^* + 1$,

$$\sum_{\{j:k^*\leq j\leq l-1;\sigma(j)=2\}} m_j \geq \frac{\rho_2}{\rho_1} \sum_{\{j:k^*\leq j\leq l-1;\sigma(j)=1\}} m_j \geq \sum_{\{j:k^*\leq j\leq l-1;\sigma(j)=1\}} m_j,$$

and therefore (30) cannot hold.

Last, we consider the case when $l \leq k^* - 1$. Then $d_{k^*}(t) = d_{k^*+1}(t) = \cdots = d_L(t)$; $d_1(t) = d_2(t) = \cdots = d_{l-1}(t) = d_{k^*}(t)$ (this second set of equations is, of course, irrelevant

when $l = 1$), and $d_l(t) = d_{l+1}(t) = \cdots = d_{k^*-1}(t) = 0$. Thus (30) becomes

$$\left(\sum_{\{j:k^* \le j \le L; \sigma(j)=2\}} m_j + \sum_{\{j:1 \le j \le l-1; \sigma(j)=2\}} m_j - \sum_{\{j:k^* \le j \le L; \sigma(j)=1\}} m_j - \sum_{\{j:1 \le j \le l-1; \sigma(j)=1\}} m_j \right) d_{k^*}(t) < 0. \tag{31}$$

But from Lemma 4(iii) we have

$$\begin{aligned} \sum_{\{j:k^* \le j \le L; \sigma(j)=2\}} m_j + \sum_{\{j:1 \le j \le l-1; \sigma(j)=2\}} m_j &\ge \frac{\rho_2}{\rho_1} \left[\sum_{\{j:k^* \le j \le L; \sigma(j)=1\}} m_j + \sum_{\{j:1 \le j \le l-1; \sigma(j)=1\}} m_j \right] \\ &\ge \sum_{\{j:k^* \le j \le L; \sigma(j)=1\}} m_j + \sum_{\{j:1 \le j \le l-1; \sigma(j)=1\}} m_j, \end{aligned}$$

and so (31) cannot hold.

Thus, in all cases (30) cannot hold, and so

$$\frac{d}{dt} \bar{I}_2(t) = 0 \text{ for all regular } t,$$

thus completing the proof. □

8 Inefficient reentrant lines

In this section, we give two examples of reentrant lines for which the integral equations (4–12) and (21–24) admit solutions which do not satisfy (25). This does not rule out the possibility that the actual fluid limits still satisfy (25).

The first example is similar to that of Harrison and Nguyen [9], which is the closed version of the examples in Kumar and Seidman [17] and Lu and Kumar [4], and is a counterexample to the conjecture that if all fluid limits of an open reentrant line under a buffer priority policy with Poisson arrivals empty in finite time, i.e, are stable, then all fluid limits of the corresponding closed reentrant line operated under the same buffer priority policy satisfy (25). The second example shows that the topology constraint that there are no self-loops, i.e., $\sigma(k) \neq \sigma(k+1)$, is not enough to guarantee (25) in two server lines. This demonstrates that even in the two server case when self loops are not the problem, alternate blocking and starvation leading to inefficient utilization could well be a problem. The second example

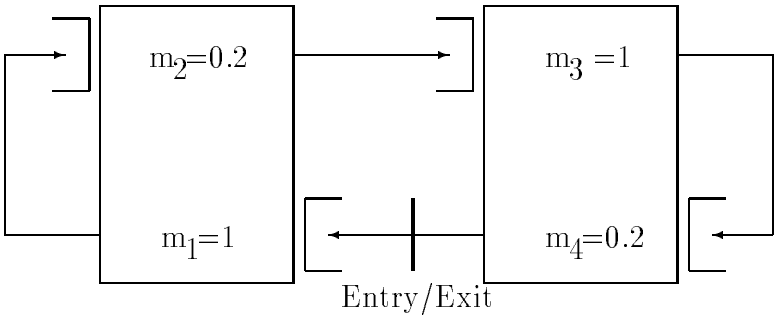

Figure 1: System of Example 1 under FBFS

Time	State
0	$(0, 1, 0, 0)$
0.2	$(0, 0, 0.8, 0.2)$
1.0	$(0, 0, 0, 1)$
1.2	$(0.8, 0.2, 0, 0)$
2.0	$(0, 1, 0, 0)$

Table 1: Inefficient trajectory for Example 1

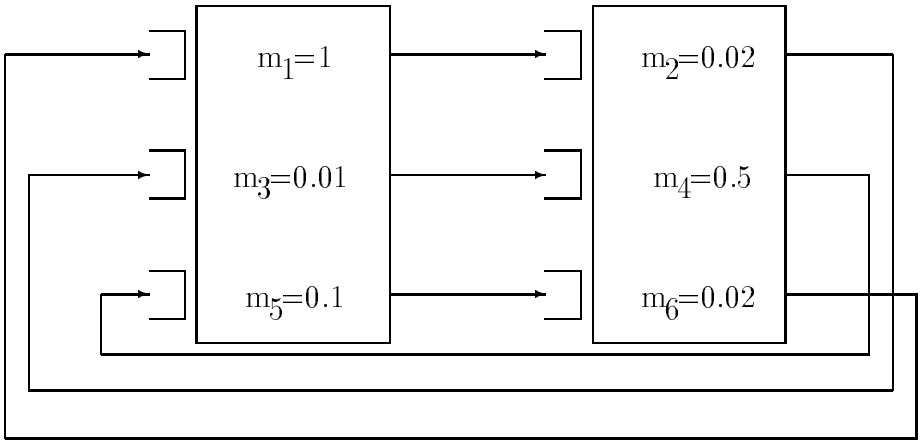

Figure 2: System of Example 2 without self-loops

also shows that even if there is only one bottleneck in a system, the fluid flows still need not satisfy (25).

Example 1 *The FBFS buffer priority policy can admit an inefficient solution to the fluid limit integral equations.*

Consider the closed reentrant line shown in Figure 1. Here $m_1 = m_3 = 1$ and $m_2 = m_4 = 0.2$. So $\lambda^* = 1/1.2$. The scheduling policy used is the FBFS policy which was shown to be stable in the open stochastic case by Dai [5]. So $b_1 \succ b_2$ and $b_3 \succ b_4$. Consider the initial condition $q = (0, 1, 0, 0)$. Then we can construct a piecewise linear trajectory which evolves as shown in Table 1.

Thus we see that $\frac{\bar{D}_L(t)}{t} \to 1/2 < \lambda^* = 1/1.2$, proving that (25) does not hold for this trajectory. Thus, in the closed case, with suitable choice of the "first" buffer, the FBFS policy can lead to inefficient fluid flows.

Example 2 *Two server reentrant lines without self loops can admit an inefficient solution to the fluid limit integral equations.*

Consider the system shown in Figure 2, with the buffer priority policy $b_1 \succ b_5 \succ b_3$ and $b_4 \succ b_2 \succ b_6$. The processing times are $m_1 = 1$, $m_2 = 0.02$, $m_3 = 0.01$, $m_4 = 0.5$, $m_5 = 0.1$,

Time	State
0	$(0,0,1,0,0,0)$
0.0125	$(0,0,0,0.975,0,0.025)$
0.5	$(0,0,0,0,0,1)$
$0.5+1/49$	$(1-1/49,0,1/49,0,0,0)$
1.5	$(0,0,1,0,0,0)$

Table 2: Inefficient trajectory for Example 2

and $m_6 = 0.02$. So $\lambda^* = 1/1.11$. An inefficient piecewise linear trajectory which satisfies the integral equations is shown in Table 2 (with its breakpoints).

Thus we see that $\frac{\bar{D}_L(t)}{t} \to 1/1.5 < 1/1.11$ and hence (25) cannot be satisfied.

9 Concluding remarks

We have investigated the fluid limit approach for analyzing the heavy traffic behavior of closed queueing networks. We have shown that the efficiency of such networks can be established by studying the throughput of the fluid limits. Using the fluid limit approach, we have proved the efficiency of two buffer priority schemes, the Last Buffer First Serve policy for general closed reentrant lines, and the Harrison–Wein buffer priority policy for two station closed reentrant lines. We have also presented examples which illustrate how inefficiency can arise in closed reentrant lines and shown that buffer priority policies which lead to stable fluid flows in an open network may still inefficiently utilize servers in the corresponding closed fluid flow network.

References

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

[2] 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*.

[3] H. Chen and A. Mandelbaum, "Discrete flow networks: Bottleneck analysis and fluid approximations," *Math. Oper. Res.*, vol. 16, pp. 408–446, 1991.

[4] 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.

[5] J. G. Dai, "Stability of open multiclass queueing networks via fluid models," tech. rep., Georgia Institute of Technology, 1994.

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

[7] 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.

[8] S. Kumar and P. R. Kumar, "Fluctuation smoothing policies are stable for stochastic reentrant lines," technical report, Coordinated Science Laboratory, University of Illinois, Urbana, IL, 1995.

[9] J. M. Harrison and V. Nguyen, "Some badly behaved closed queueing networks," Technical Report 3691-94-MSA, Stanford and MIT, May 1994.

[10] 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.

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

[12] D. J. Miller, "Simulation of a semiconductor manufacturing line," *Communications of the ACM*, vol. 33, no. 10, pp. 98–108, 1990.

[13] J. Walrand, *Communication Networks–A first course*. Boston: Aksen Associates, 1991.

[14] D. Bertsekas and R. Gallager, *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[15] S. N. Ethier and T. G. Kurtz, *Markov Processes: Characterization and Convergence*. New York, NY: Wiley, 1985.

[16] J. G. Dai and R. J. Williams, "Existence and uniqueness of semimartingale reflecting Brownian motions in convex polyhedrons." Submitted to *Theory of Probability and its Applications*, 1994.

[17] 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.