ELSEVIER

European Journal of Operational Research 110 (1998) 585–596

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

Theory and Methodology

# Asymptotic loss of priority scheduling policies in closed re-entrant lines: A computational study

Y. Narahari *, L.M. Khan [1]

*Department of Computer Science and Automation, Indian Institute of Science, Bangalore - 560012, India*

Received 1 March 1996; accepted 1 April 1997

**Abstract**

In this paper we present an approximate but efficient analytical method to compute the asymptotic loss of buffer priority scheduling policies in closed re-entrant lines. For simple two-station closed re-entrant lines, this enables the verification of Harrison–Wein conjectures and Jin–Ou–Kumar results. For multi-station re-entrant lines, this provides an efficient way of comparing different buffer priority scheduling policies. We also use the method to evaluate the effect of high priority jobs in re-entrant lines. © 1998 Elsevier Science B.V. All rights reserved.

*Keywords:* Re-entrant lines; Asymptotic loss; Buffer priority policies; Mean value analysis

## 1. Introduction

In closed multi-class queueing networks, it is of much interest to evaluate the performance of scheduling policies. This is especially true in view of the fact that there exist scheduling policies that cannot attain the maximum achievable throughput even under infinite population in a closed queueing network [1,2]. Since priorities in scheduling policies render the queueing network non-product form, there are no exact methods available to compute performance metrics, such as, say, steady-state throughput rate and mean steady-state response time. Recently several researchers have presented methods to compute bounds on the steady-state performance of stationary, non-idling, buffer priority-based scheduling policies, by solving linear programs [3–5,2]. Simulation has been used by some researchers [6–8] to compute the mean and variance of performance measures of various scheduling policies in closed multi-class networks. A mean value analysis-based method for computing the performance of closed re-entrant lines under buffer priority policies has also been proposed [8,9].

The notion of asymptotic loss of a scheduling policy in a closed queueing network was introduced by Harrison and Wein [10] and more recently has been studied in great detail by Jin et al. [2]. The asymptotic loss of a scheduling policy captures the rate at which the throughput of the network attains the maximum possible throughput

* Corresponding author. Address: National Institute of Standards and Technology, Building 304/12, Gaithersburg, MD 20899, USA. E-mail: hari@csa.iisc.ernet.in.

[1] E-mail: mohd@chanakya.csa.iisc.ernet.in.

0377-2217/98/$19.00 © 1998 Elsevier Science B.V. All rights reserved.
PII S0377-2217(97)00302-0

and therefore can be used to compare different scheduling policies. In this paper, we focus on closed re-entrant lines [11] and present an approximate but efficient computational method for computing the asymptotic loss of any buffer priority-based scheduling policy.

### 1.1. Re-entrant lines

Re-entrant lines [11] are a class of non-traditional queueing network models that are appropriate for modeling manufacturing systems with distinct multiple job visits to work centers. Examples of such manufacturing systems include semiconductor wafer fabrication facilities, thin film lines, and systems with rework tasks.

In a re-entrant line, the parts visit the same machine several times, at different stages of processing, before exiting the system, thus making the flow *non-acyclic.* A re-entrant line can be described as follows. There is a set of *service centers* or *stations* $\{1,2,\ldots,m\}$. Service center $i \in \{1,2,\ldots,m\}$ has $n_i$ logical or physical buffers, $b_{i1}, b_{i2}, \ldots, b_{in_i}$. For $j \in \{1,2,\ldots,n_i\}$, the buffer $b_{ij}$ contains parts visiting service center $i$ for the $j$th time. A part visits these buffers in a given sequence and any service center is typically visited several times in the route of a part.

Fig. 1 shows a typical re-entrant line with three stations and 11 buffers. Parts enter the system at buffer $b_{11}$ and visit the centers according to a deterministic route as shown. Finished parts emerge from center 3 after undergoing processing following a wait in $b_{33}$. Note that every part in this example line visits center 1 three times, center 2 five times, and center 3 three times.

A scheduling policy in re-entrant line decides which job to process next when a machine becomes available. Scheduling at a station is necessitated because several parts in different stages of processing may be in contention with one another for service at the same machine. A prominent class of scheduling policies discussed in the literature [11,2] is the class of buffer scheduling policies. When a processing center $i$ finishes processing a part, a buffer priority policy selects the next part for processing from among the buffers in a fixed priority order, which is independent of the state of the system. We shall assume that the priorities accorded are preemptive. Two prominent buffer priority-based policies are the Last Buffer First Serve (LBFS) and the First Buffer First Serve (FBFS) policies. For example, in the case of LBFS applied to the re-entrant line in Fig. 1, we order the buffers at, say, station 2 in the order $b_{25}$, $b_{24}$, $b_{23}$, $b_{22}$, and $b_{21}$ (decreasing order of priority). A part in $b_{24}$ will be taken up for processing if there are no parts waiting in $b_{25}$; a part in $b_{23}$ will be taken up only if there are no parts in $b_{24}$ and no parts in $b_{25}$; and so on.

In this paper we only consider closed re-entrant lines. Such lines are appropriate for modeling fixed-work-process input release policies [12,6]. Also, we consider only re-entrant lines with deterministic route for parts. The results provided here are applicable with straightforward extension to re-entrant lines with probabilistic routing. See for example, Narahari and Khan [13] where the computational methodology for classical re-entrant lines is extended to account for probabilistic routing.

### 1.2. Asymptotic loss

Except in product form queueing networks, very little is known about the throughput of a scheduling policy as the population of customers in a closed queueing network is increased. An interesting issue is whether for a given scheduling policy, the throughput of a closed network will converge to the maximum possible value (which is the throughput capacity of the bottleneck station) when the population is increased to infinity. In particular, the rate of increase of throughput with population is of interest. The notion of asymptotic loss of a scheduling policy in a closed queueing network serves this purpose. This notion has been considered by Harrison and Wein [10] and further enunciated by Jin et al. [2]. Let $u$ be a scheduling policy. Then we define (see [2]) *upper and lower asymptotic losses* respectively, of the policy $u$, by

$$\bar{J}(u) = \limsup_{n\to\infty} n\frac{\alpha^* - \alpha^u(n)}{\alpha^*}, \tag{1}$$

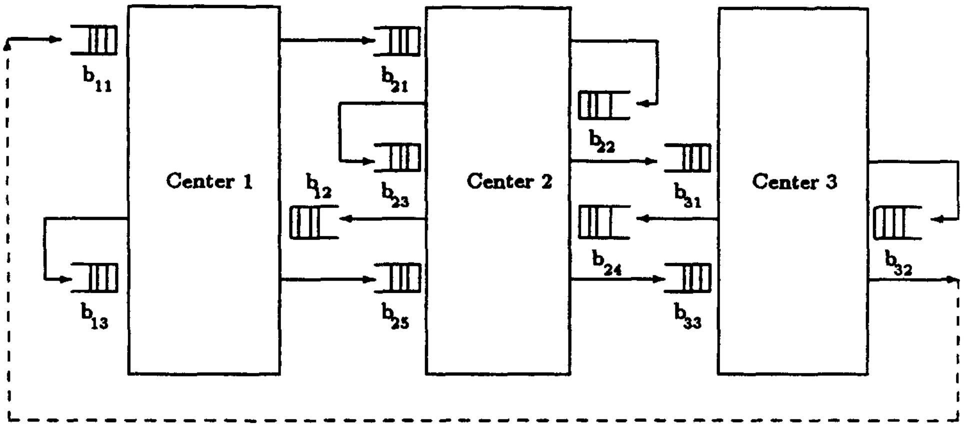

Fig. 1. A re-entrant line with three stations and 11 buffers.

$$\underline{J}(u) = \liminf_{n\to\infty} n\frac{\alpha^* - \alpha^u(n)}{\alpha^*}, \tag{2}$$

where $\alpha^u(n)$ is the steady-state throughput rate of the given network for a population of $n$ under policy $u$ and $\alpha^*$ is the maximum steady-state throughput rate attainable in the given network, which is equal to the bottleneck capacity. If the defining limit exists, we also define the asymptotic loss $J(u)$ of scheduling policy $u$ by

$$J(u) = \lim_{n\to\infty} n\frac{\alpha^* - \alpha^u(n)}{\alpha^*}. \tag{3}$$

A policy $u$ is said to be *asymptotically optimal* if the asymptotic loss is as small as possible. The notion of asymptotic loss is a convenient one for describing the rate of approach to the maximum throughput under a given scheduling policy. For this reason, it can be used for comparing the performance levels of scheduling policies.

In [10], Harrison and Wein looked at two station networks from a dynamic scheduling viewpoint. They used heavy traffic theory of queueing networks to synthesize a buffer priority policy, for two station networks, which was conjectured to be asymptotically optimal. For two station networks, they also conjectured an expression for the asymptotic loss of any buffer priority scheduling policy. These two conjectures are essentially based on a careful study of the reflected Brownian motion arising from a "workload imbalance process" defined for the network under heavy traffic conditions. For more details on these conjectures, see Harrison and Wein [10] and Jin et al. [2]. Chevalier and Wein [14] conducted a similar study, based on heavy traffic theory, on multi-station networks, but their study does not address the computation of asymptotic loss for any general multi-class queueing network.

More recently, Jin et al. [2] have looked at several issues related to the performance of buffer priority scheduling policies in closed queueing networks. Their first result is in deriving two linear programs that yield bounds on asymptotic loss for closed queueing networks. For two station networks, they have shown that the Harrison–Wein policy is efficient (i.e. under this policy, the system throughput converges to the maximum possible value). They have also determined an upper bound on the asymptotic loss of the Harrison–Wein policy for two station re-entrant lines and shown that the asymptotic loss of any buffer priority scheduling policy is no less than that of Harrison–Wein policy.

The results of Harrison and Wein [10] are applicable only to two station networks, whereas the results of Jin et al. [2] only enable computation of bounds on asymptotic loss. Also, a common method like simulation simply cannot provide credible estimates of asymptotic loss unless we can afford to spend huge computing resources. This is because of the need to compute the difference of two very nearly equal quantities and the need to experiment with very large populations. Motivated by this, our aim in this paper is to provide an analytical method to compute accurately the asymptotic loss of a given buffer priority policy in closed re-entrant lines.

### 1.3. Contributions of the paper

In this paper, we focus on asymptotic loss of buffer priority scheduling policies in closed re-entrant lines. We first show, in Section 2, that the asymptotic loss of a simple, two station, closed product form network is zero if the network is unbalanced and unity if it is balanced. In Section 3, we present an approximate but efficient computational method, based on mean value analysis, for computing asymptotic loss. By considering several two station re-entrant lines, we verify, in Section 4, the validity of Harrison–Wein expressions for asymptotic loss. In Section 5, we consider re-entrant lines with internal buffers and bring out the difference in performance between scheduling policies. Finally in Section 6, we consider re-entrant lines with high priority jobs and show by computing asymptotic loss, the degradation in the performance of low priority jobs as more and more high priority jobs are introduced in the network. In all relevant cases, we have also carried out simulations to show the accuracy of the proposed methodology.

## 2. A closed product form re-entrant line

Fig. 2 shows a simple two-station, two-buffer re-entrant line. Each station has only one machine with processing time exponentially distributed with rate $\mu_i$. Since there is only one buffer at each station, the above network is product form under the FCFS scheduling policy. It can be shown [15] that the throughput rate of this network is given by

$$\alpha(n) = \frac{n}{n+1}\mu_1 \quad (\mu_1 = \mu_2)$$
$$= \frac{1-\rho^n}{1-\rho^{n+1}}\mu_1 \quad (\mu_1 < \mu_2),$$

where $\rho = \mu_1/\mu_2$.

When $\mu_1 = \mu_2$, the network is balanced and we get the asymptotic loss as

$$J = \lim_{n\to\infty} n\frac{\alpha^* - \alpha^u(n)}{\alpha^*} = \lim_{n\to\infty}\frac{n}{n+1} = 1,$$

since $\alpha^* = \mu_1$.

When $\mu_1 < \mu_2$ (unbalanced case), $\alpha^* = \mu_1$, and we get the asymptotic loss as

$$J = \lim_{n\to\infty} n\frac{\rho^n - \rho^{n+1}}{1-\rho^{n+1}} = 0.$$

Thus the throughput of the network converges rapidly to the maximum possible in the unbalanced case, whereas in the balanced case, the convergence is slower.

## 3. A computational method

Computation of asymptotic loss by directly computing the limits is feasible only for a very lim-

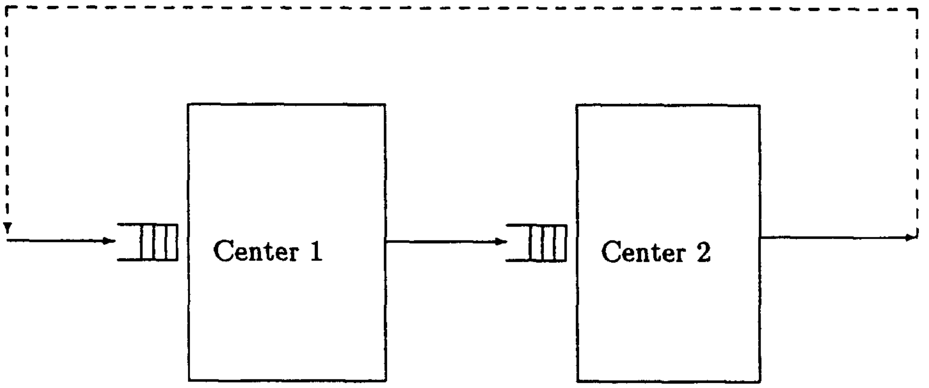

Fig. 2. A re-entrant line with two stations and two buffers.

ited class of product form closed re-entrant lines. Since priorities in scheduling and multiple visits render the network non-product form, such direct computation is not feasible for re-entrant lines of interest. We now present a computational method for this purpose, based on mean value analysis.

Mean value analysis (MVA) [16,17] is a well known computational technique for predicting the performance of product form queueing networks. In [8], Narahari and Khan have presented an MVA-based technique for predicting the performance of buffer priority policies in closed re-entrant lines. We propose a similar method here. The proposed method does not give exact values since the underlying network is not product form.

Since asymptotic loss of a policy $u$ is the limiting value (see Eq. (3)), we seek to compute the values of

$$n\frac{\alpha^* - \alpha^u(n)}{\alpha^*}$$

for very large values of $n$.

MVA yields expressions for mean values of performance measures such as steady-state queue lengths, delays, and throughputs. Two versions of MVA exist, namely, the *exact MVA* for product form queueing networks [16] and *approximate MVA* for non-product form networks [18]. Exact MVA is based on the *Arrival Theorem*, which states that, in the steady state of a closed product form network with population $k$, the distribution of the network state seen by a job arriving at any node in the network is the same as the distribution of the network state a random observer would see with $(k-1)$ jobs circulating in the network. In the literature, several extensions have been proposed to MVA to account for non-product form features and more specifically priorities. See [17,19] for a review. For closed re-entrant lines with buffer priority policies, none of the existing methods is applicable and therefore we present our own MVA-based approximate method here. This method has earlier been used by the authors in several contexts [8,9].

We shall illustrate the formulation of MVA equations by assuming the LBFS scheduling policy. We assume that each processing center has exactly one machine and that the processing time of a job visiting center $i$ on its $j$th visit is an independent exponentially distributed random variable with rate $\mu_{ij}$. Let us denote the $j$th visit of a job to the center $i$ by stage $(i,j)$ of processing.

Let the performance measures of the network be denoted as follows: $L_{ij}(k)$ is the mean steady-state number of jobs in stage $(i,j)$ when the network has $k$ jobs, $W_{ij}(k)$ the mean steady-state delay for jobs in stage $(i,j)$ (mean waiting time in buffer $b_{ij}$ + mean processing time), and $\alpha(k)$ is the mean steady-state throughput rate of jobs when the network has $k$ jobs.

If $W(k)$ denotes the mean total delay (mean cycle time) in the entire network, we immediately have

$$W(k) = \sum_{i=1}^{m}\sum_{j=1}^{n_i} W_{ij}(k). \tag{4}$$

Using MVA, we compute $W(N)$, and $\alpha(N)$ in a recursive way.

We also distinguish between *external* and *internal* buffers. We call a buffer $b_{ij}$ external if the buffer feeding $b_{ij}$ is connected to a center different from center $i$, and buffer $b_{ij}$ is called internal if the buffer feeding $b_{ij}$ is connected to center $i$ itself. For example, in the re-entrant line in Fig. 1, consider center 2. The buffers $b_{21}, b_{24}$, and $b_{25}$ are external, since arrivals into these buffers come from center 1, center 3, and center 1, respectively. The buffers $b_{22}$ and $b_{23}$ are internal because they are directly fed by outputs from center 2 itself.

### 3.1. Computation of performance measures

We consider the calculation of $W(N)$ and $\alpha(N)$. It would be helpful to consider the scenario a job would see upon its arrival at a certain buffer of a machine, and the sequence of events that occur while it is waiting there.

When a job (we shall call it a distinguished job) arrives at a buffer, say $b_{ij}$, it sees a certain number of jobs in various buffers in the system, the ordered set of these integers forms the *state of the system* at the arrival instant of the job. Let $S$ be the set of jobs, currently at center $i$ and having higher priority than the distinguished job. Note that $S$ will include all jobs that are ahead of the distinguished job in $b_{ij}$ and all jobs in all buffers having higher priority than $b_{ij}$. The distinguished job must first wait until all jobs in $S$ are serviced and leave the center $i$. Also, it must wait for the service completion of those jobs which arrive in higher priority buffers, during its wait in buffer $b_{ij}$. And finally it has to get processed before it enters the next buffer.

Hence, the mean total waiting time of a job at any buffer $b_{ij}$ is seen as the sum of three components, let us call them Term 1, Term 2, and Term 3, defined as follows.

- *Term* 1: Mean total time until all jobs in the set $S$ are serviced and leave center $i$.
- *Term* 2: Mean total time required to process all higher priority jobs which arrive during the stay of the distinguished job in the queue at $b_{ij}$.
- *Term* 3: Mean processing time of the distinguished part itself.

We now describe how Terms 1–3 may be computed. We shall describe the case of lines without any internal buffers. Lines with internal buffers require a somewhat different treatment, the details of which can be seen in [8].

### 3.1.1. Computation of term 1

Consider the buffer $b_{ij}$. In this case, an arriving job, according to the Arrival theorem, would see $L_{it}(k-1)$ jobs in the buffers $b_{it}$, where $t = 1,2,\ldots,n_i$. Since LBFS scheduling policy is being used, the arriving job needs only to wait for the processing of jobs waiting ahead of it in buffers $b_{it}$, where $t = j, j+1,\ldots,n_i$. Thus

$$\text{Term } 1 = \sum_{t=j}^{n_i} \frac{L_{it}(k-1)}{\mu_{it}}. \tag{5}$$

### 3.1.2. Computation of term 2

The mean waiting time of a job in buffer $b_{ij}$ is $W_{ij}(k)$. During this waiting, parts may arrive into higher priority buffers at center $i$. Term 2 is the mean total time required to process all such parts. Since all the buffers in the model are external, then during the waiting, parts may arrive into any of the higher priority buffers (from other machines). In fact, the mean number of parts that arrive into any of the higher priority buffers is the same since every part flows through all the buffers according to a deterministic route. Consequently, the mean throughput rate into all the buffers in the network is the same. By assuming the arrival theorem, $\alpha(k-1)$ can be taken as the rate at which the jobs are flowing in the network. The mean number of parts arriving into each higher priority buffer during the waiting of a job in buffer $b_{ij}$ is therefore given by $W_{ij}(k)\alpha(k-1)$. Since $1/\mu_{it}$ is the mean service time in a higher priority buffer $b_{it}$, where $t = j+1,\ldots,n_i$, we have

$$\text{Term } 2 = W_{ij}(k)\alpha(k-1)\left(\sum_{t=j+1}^{n_i} \frac{1}{\mu_{it}}\right). \tag{6}$$

### 3.1.3. Computation of term 3

The mean processing time required for the service of distinguished part itself is of course, $1/\mu_{ij}$. Thus Term 3 $= 1/\mu_{ij}$. The total waiting time $W_{ij}(k)$ is now given by

$$W_{ij}(k) = \text{Term } 1 + \text{ Term } 2 + \text{Term } 3. \tag{7}$$

Now using (4), $W(k)$ can be computed. Applying Little's Law [15] for the job population in the network, we obtain

$$\alpha(k) = \frac{k}{W(k)}. \tag{8}$$

We can again use Little's Law to obtain

$$L_{ij}(k) = \alpha(k)W_{ij}(k). \tag{9}$$

Consider the following initial conditions

$$L_{ij}(0) = 0, \quad i = 1,\ldots,m, \ j = 1,\ldots,n_i, \tag{10}$$

$$\alpha(0) = 0. \tag{11}$$

Using the initial conditions above and the recurrence relations defined by (7)–(9), and the initial values (10) and (11), we can compute $W_{ij}(k)$, $L_{ij}(k)$, and $\alpha(k)$ for $k = 1,2,\ldots,N$. Thus $W(N)$ and $\alpha(N)$ can be computed.

### 3.2. A fixed point method

The method just described is recursive, and builds up a solution for a given network population by starting from the empty system, when the various performance measures are trivially known, and then increasing the population in steps of 1 and computing performance measures at each step until the desired population is reached. However, for the computation of asymptotic loss we have to compute the performance metrics of the re-entrant lines at very high populations (ideally infinite), and this would mean that we have to start from a population equal to zero and approach a desired high population in steps of 1, which can be very time consuming.

To overcome this difficulty, an approximate version of MVA, due to Schweitzer [17] and Bard [18] can be used. This method breaks the recursion relations of the MVA-based technique just described and replaces them with a set of non-linear fixed point equations.

The recursive nature of the MVA-based method above arises due to dependence of $W_{ij}(k)$ upon $L_{ij}(k-1)$, which in turn, depends upon $W_{ij}(k-1)$ and so on. Schweitzer [17] and Bard [18] observed that the fraction of the total population of jobs at each buffer does not change much if there is one less job in the network. In other words they argued that

$$L_{ij}(k-1) = \frac{k-1}{k} L_{ij}(k).$$

When this expression for $L_{ij}(k-1)$ is substituted in earlier expressions, the recursion relations turn into a set of non-linear fixed point equations, which can be solved iteratively by successive substitution. This method eliminates the need for computing performance measures for all populations ranging from $1, \ldots, k$ by directly solving the network for any population equal to $k$. This was the method which we implemented to obtain the numerical results presented in the sequel.

## 4. A two station re-entrant line with eight buffers

Consider the re-entrant line shown in Fig. 3. For this system, we computed the asymptotic loss of various scheduling policies under two cases: Balanced case and the unbalanced case. In the balanced case, we assumed

$$\mu_{ij} = 1 \quad \forall i = 1, 2, \; j = 1, \ldots, 4.$$

Since there are 4! ways of choosing a priority order among the buffers at each station, there are a total of 576 possible buffer scheduling policies here. We considered four representative ones:

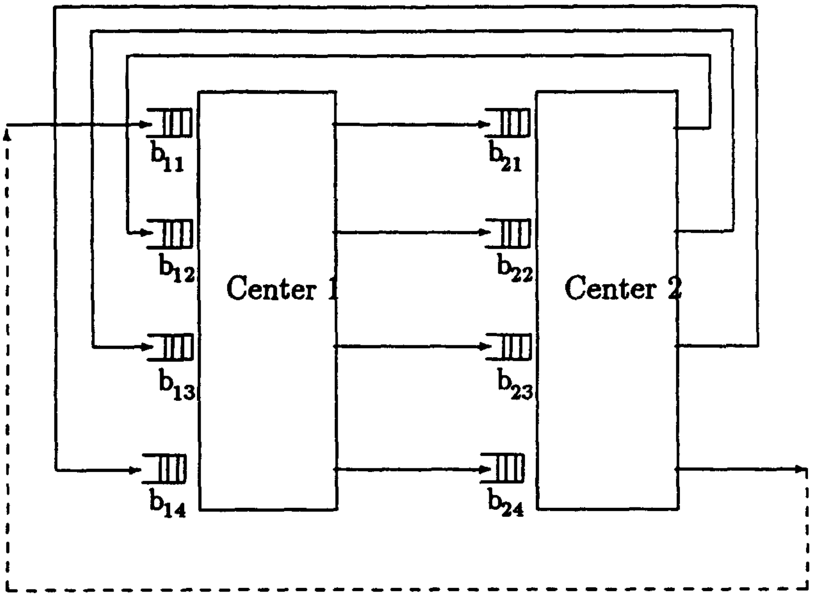

Fig. 3. A re-entrant line with two stations and eight buffers.

1. LBFS at station 1, LBFS at station 2.
2. LBFS at station 1, FBFS at station 2.
3. FBFS at station 1, LBFS at station 2.
4. FBFS at station 1, FBFS at station 2.

It was found in each case that the asymptotic loss is equal to 1. This is confirmed by the Harrison–Wein expressions [10] and also the bounds obtained by Jin et al. [2]. There is good reason to believe that all the 576 policies will have the same asymptotic loss, equal to 1, because of identical processing requirement in all the buffers and symmetric routing.

In the unbalanced case, we assumed

$$\mu_{11} = \mu_{12} = \mu_{13} = \mu_{14} = 2,$$

$$\mu_{21} = \mu_{22} = \mu_{23} = \mu_{24} = 1.$$

In this case, station 2 is the bottleneck and the maximum throughput achievable is decided by its capacity. We computed the asymptotic loss for the four policies above and found it to be zero in all the cases. Here again, all the 576 policies are expected to yield zero or very small values of asymptotic loss. In the unbalanced case, the system throughput rapidly tries to reach the maximum possible value since station 1 keeps pushing out jobs for processing by station 2, whatever the scheduling policy. Consequently station 2 is kept busy almost all the time, leading to a throughput very close to the maximum achievable one.

## 5. Re-entrant lines with internal buffers

Here, we first consider the re-entrant line shown in Fig. 4. This line has been studied in various contexts in the literature [11,10,2]. Assuming $\mu_{11} = \mu_{12} = \mu_{21} = \mu_{22} = 1$, we shall compute the asymptotic loss under all the four possible policies, namely: Policy 1 (LBFS at station 1, LBFS at station 2); Policy 2 (LBFS at station 1, FBFS at station 2); Policy 3 (FBFS at station 1, LBFS at station 2); Policy 4 (FBFS at station 1, FBFS at station 2). Table 1 shows the asymptotic loss values computed by the method proposed in Section 3. The table also shows the values obtained from detailed simulations. Each such value is obtained as the average of the values obtained by running several independent simulations at a level of significance of 0.05, for a very large population of 200. The values obtained were found to saturate for populations higher than 200. There is a close agreement between the computed values and those obtained using simulation, thus validating the accuracy of the proposed computational methodology for such re-entrant lines.

For this re-entrant line, using the expressions given in [2], the Harrison–Wein policy can be shown to be the same as Policy 3 whereas the anti-Harrison–Wein policy (buffer priorities exactly opposite to those in the Harrison–Wein policy) can be computed to be the same as Policy 2. From Table 1, Policy 3 has the least asymptotic loss whereas Policy 2 has the highest. This is consistent with the Harrison–Wein conjecture that their policy is asymptotically optimal and with the result of Jin et al. [2] about the bounds provided by the Harrison–Wein and the anti-Harrison–Wein policies.

Also note that in Policy 3, a part from $b_{12}$ is taken up for processing only when $b_{11}$ is empty

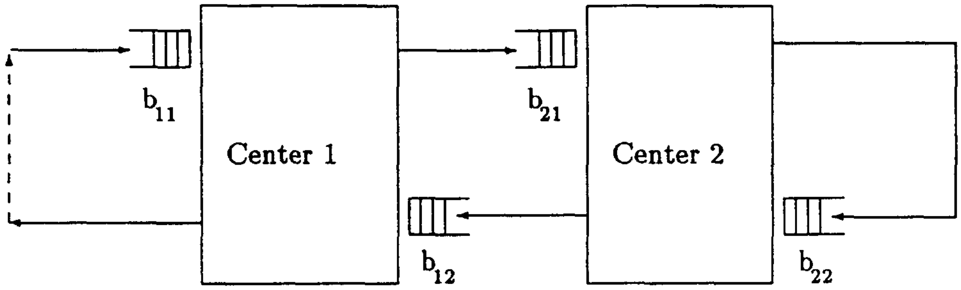

Fig. 4. A two station re-entrant line with internal buffers.

Table 1
Asymptotic losses for the re-entrant line in Fig. 4

| Scheduling policy $u$ | Asymptotic loss $J(u)$ using MVA | Asymptotic loss $J(u)$ using simulation |
|---|---|---|
| Policy 1 | 0.750 | 0.7562 |
| Policy 2 | 1.000 | 1.0147 |
| Policy 3 | 0.500 | 0.4977 |
| Policy 4 | 0.750 | 0.7521 |

whereas a part from $b_{21}$ is taken up for processing only when $b_{22}$ is empty. Also as soon as a part from $b_{12}$ finishes processing, another part enters $b_{11}$ and is immediately taken up for processing. Similarly at station 2, a part from $b_{21}$ after processing will again be processed immediately in $b_{22}$. As a consequence, the processing time of each job at station 1 and station 2 can be considered to be a 2-stage Erlangian distribution, which has the least variance among all two stage distributions. This leads to the minimum cycle time (a direct consequence of Pollaczek–Khintchine formula for M/G/1 queues [15]).

Fig. 5 shows the throughput rate attained by the four policies at different populations. Note that Policy 1 and Policy 4 exhibit identical behavior. Policy 3 shows the best performance while Policy 2 exhibits the worst performance. This graph again exemplifies the foregoing conclusions.

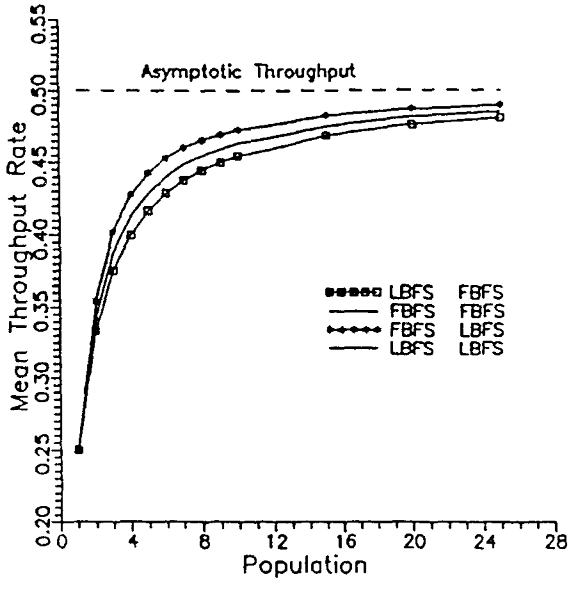

Fig. 5. Throughput rates for different scheduling policies.

Now we consider the 3-station, 11-buffer re-entrant line of Fig. 1. For this re-entrant line, the Harrison–Wein conjectures can no longer be applied (since the number of stations is more than 2). Table 2 shows the asymptotic losses for eight different policies, assuming a balanced line with

$$\mu_{11} = \mu_{12} = \mu_{13} = 3,$$
$$\mu_{21} = \mu_{22} = \mu_{23} = \mu_{24} = \mu_{25} = 5.$$
$$\mu_{31} = \mu_{32} = \mu_{33} = 3.$$

The first entry in Table 2 corresponds to the LBFS policy followed at stations 1, 2, and 3; the second entry to LBFS at station 1 and 2 and FBFS at station 3; and so on. The table also shows the values obtained as the averages of the values obtained by running several simulations at a level of significance of 0.05, for a very large population of 750. The values obtained were found to saturate for populations higher than 750. The close agreement between the asymptotic loss values in the second and third columns provides a kind of validation for our computational method.

Note that among the eight policies considered, LBFS at all stations shows the best behavior while FBFS at all stations has the worst behavior. Thus we are able to rank-order the scheduling policies in terms of their ability to attain the maximum throughput.

Table 2
Asymptotic losses for the re-entrant line in Fig. 1

| Scheduling policy $u$ | Asymptotic loss $J(u)$ using MVA | Asymptotic loss $J(u)$ using simulation |
|---|---|---|
| LBFS-LBFS-LBFS | 0.802 | 0.8156 |
| LBFS-LBFS-FBFS | 1.192 | 1.1935 |
| LBFS-FBFS-LBFS | 1.338 | 1.34162 |
| LBFS-FBFS-FBFS | 1.778 | 1.7962 |
| FBFS-LBFS-LBFS | 1.022 | 1.0635 |
| FBFS-LBFS-LBFS | 1.414 | 1.40056 |
| FBFS-FBFS-LBFS | 1.611 | 1.62012 |
| FBFS-FBFS-FBFS | 2.000 | 1.98741 |

## 6. Re-entrant lines with high-priority jobs

This example is motivated by semiconductor manufacturing systems in which high priority jobs, called hot lots, are often introduced into the system, out of marketing and business considerations. Hot lots are exactly like the regular jobs with identical route and processing requirements but they get priority over regular jobs at all processing stages. Ehteshami et al. [20] have carried out a simulation study to understand the effects of hot lots on the cycle time and throughput of regular lots. More recently, Narahari and Khan [9] have presented an analytical method for explicitly evaluating the effects of hot lots. In both these works, the findings show that as the proportion of hot lots in the work-in-process increases, both the mean cycle time and mean throughput rate of regular lots are drastically affected.

Here, we will study the effect of hot lots by computing the asymptotic loss of the regular lots as we increase the number of hot lots in the network. Consider the 2-station, 4-buffer re-entrant line in Fig. 6. Assuming the LBFS policy we have computed the asymptotic loss for regular lots at various hot lot populations ranging from 0 to 10. For this, we use an extended computational methodology, as outlined in [9]. Given that the regular lot population is $N$ and that the hot lot population is $H$, the methodology above computes performance measures for regular lots in exactly $N$ iterations. The performance measures for hot lots are obtained in exactly $H$ iterations. Table 3 shows these asymptotic losses for LBFS policy, assuming

$$\mu_{11} = \mu_{12} = \mu_{21} = \mu_{22} = 1.$$

The table also shows the values obtained as the averages of the values obtained by running several simulations at a level of significance of 0.05, for a very large regular lot population of 400. The values obtained were found to saturate for populations of regular lots higher than 400. The values obtained using simulation agree closely with the computed values, thus validating our computational methodology for this case.

The loss of throughput experienced by regular lots with increased hot lot population is clearly captured by the asymptotic loss values in Table 3. The implication is that, in the presence of hot lots, we need much more Work-In-Process (WIP) of regular lots to attain a desired throughput of regular lots.

## 7. Conclusions and future work

The main aim of this paper was to provide an efficient computational method to compute the asymptotic loss of scheduling policies in closed re-entrant lines. The proposed method is approximate but is validated by simulation results. Also, the results obtained using the proposed method are consistent with the findings of Harrison and Wein [10] for two station re-entrant lines. Through several experiments, we have shown how asymptotic loss

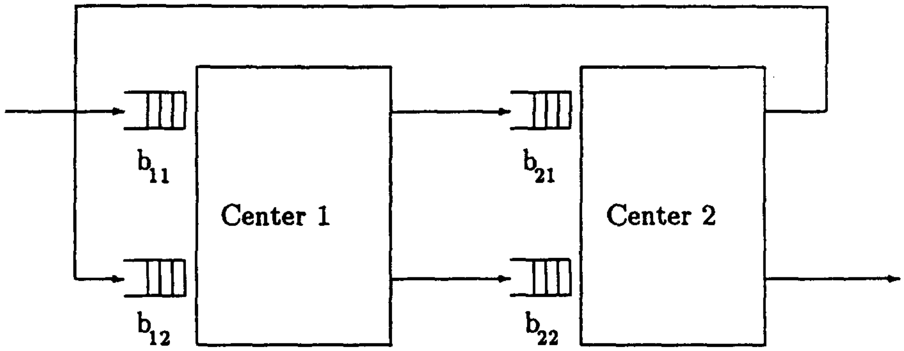

Fig. 6. A re-entrant line with two stations and four buffers.

Table 3
Asymptotic losses of regular lots for the re-entrant line in Fig. 6

| Hot lot population | Asymptotic loss of regular lots using MVA | Asymptotic loss of regular lots using simulation |
|---|---|---|
| 0 | 1.49000 | 1.5231 |
| 1 | 3.45380 | 3.63125 |
| 2 | 5.39210 | 5.46225 |
| 3 | 7.30520 | 8.05621 |
| 4 | 9.19360 | 9.26632 |
| 5 | 11.0577 | 11.15738 |
| 6 | 12.8981 | 12.57761 |
| 7 | 14.7153 | 14.41236 |
| 8 | 16.5096 | 15.9665 |
| 9 | 18.2816 | 18.9522 |
| 10 | 20.0315 | 21.13421 |

can be used to compare different buffer priority scheduling policies and evaluate the effect of high priority jobs.

The limitation of this work is the lack of an adequate proof for the correctness of the asymptotic loss values computed here. This is because not much is known about asymptotic loss in re-entrant lines with more than two stations, except for bounds [2]. Also simulation is not a feasible way of validating our results because of the nature of computation involved and the resulting intractability. In fact, even for the simple re-entrant lines studied here, several hours of CPU time were required to obtain credible results using simulation. This is the most important topic for future work.

## Acknowledgements

The authors have benefited immensely from discussions with Professor P.R. Kumar, University of Illinois, Urbana-Champaign. In fact, it was he who introduced the authors to this topic and gave deep insight into the problems and issues here. This research was supported in part by the Office of Naval Research and the Department of Science and Technology grant N00014-93-1017. We would also like to acknowledge the excellent facilities at the Intelligent Systems Laboratory, Department of Computer Science and Automation, Indian Institute of Science.

## References

[1] J.M. Harrison, V. Nguyen, Some Badly Behaved Closed Queueing Networks, Stanford University and Massachusetts Institute of Technology , 1994.

[2] H. Jin, J. Ou, P.R. Kumar, The throughput of closed queueing networks: functional bounds asymptotic loss efficiency and the Harrison–Wein conjectures, Technical Report, The Cordinated Science Laboratory, University of Illinois-Urbana Champaign,1995.

[3] B. Ou, L.M. Wein, Performance bounds for scheduling queueing networks, The Annals of Applied Probability 2 (2) (1992) 460–480.

[4] C. Bertsimas, D. Paschalidis, J.N. Tsitsiklis, Optimization of multiclass queueing networks: polyhedral and nonlinear characterizations of achievable performance, Annals of Applied Probability 4 (1994) 43–75.

[5] S. Kumar, P.R. Kumar, Performance bounds for queueing networks and scheduling policies, IEEE Transactions on Automatic Control 39 (8) (1994) 1600–1611.

[6] L.M. Wein, Scheduling semiconductor wafer fabrication, IEEE Transactions on Semiconductor Manufacturing 1 (3) (1988) 115–130.

[7] S.H. Lu, D. Ramaswamy, P.R. Kumar, Efficient scheduling policies to reduce mean and variance of cycle-time in semiconductor manufacturing plants, IEEE Transactions on Semiconductor Manufacturing 7 (3) (1994) 374–388.

[8] Y. Narahari, L.M. Khan, Performance analysis of scheduling policies in re-entrant manufacturing systems, Computers and Operations Research 23 (1) (1996) 37-51.

[9] Y. Narahari, L.M. Khan, Modeling the effect of hot lots in semiconductor manufacturing Systems, Technical Report, Department of Computer Science and Automation, Indian Institute of Science, 1994.

[10] J.M. Harrison, L.M. Wein, Scheduling networks of queues: heavy traffic analysis of a two-station closed network, Operations Research 38 (1990) 1052–1064.

[11] P.R. Kumar, Re-entrant lines, Queueing Systems: Theory and Applications 13 (1993) 87–110.

[12] C.R. Glassey, M.G.C. Resende, Closed-loop job release control for VLSI circuit manufacturing, IEEE Transactions on Semiconductor Manufacturing 1 (1) (1988) 36–46.

[13] Y. Narahari, L.M. Khan, Modeling re-entrant manufacturing systems with inspections Journal of Manufacturing Systems, Vol. 15 (No. 6) (1996) 367-378.

[14] P.B. Chevalier, L.M. Wein, Scheduling network of queues: heavy traffic analysis of a multistation closed network, Operations Research 41 (4) (1993) 743–758.

[15] N. Viswanadham, Y. Narahari, Performance Modeling of Automated Manufacturing Systems, Prentice-Hall, Englewood Cliffs, NJ, 1992.

[16] M. Reiser, S.S. Lavenberg, Mean value analysis of closed multichain queueing networks, Journal of the ACM 27 (2) (1980) 313–322.

[17] P.J. Schweitzer, A survey of mean value analysis, its generalizations, and applications for networks of queues,

in: Second International Conference on Mathematics for Operations Research, Amsterdam, 1990.

[18] Y. Bard, Some extensions to multiclass queueing network analysis, in: M. Arato, A. Butrimenko, E. Gelenbe (Eds.), Performance of Computer Systems, North-Holland, Amsterdam, 1979, pp. 51–61.

[19] R. Suri, J.L. Sanders, M. Kamath, Performance evaluation of production networks, in: S.C. Graves, A.G. Rinnoy Kan, P. Zipkin (Eds.), Handbooks in OR and MS, vol. 4, Elsevier, Amsterdam, 1993, pp. 199–286.

[20] B. Ehteshami, R.G. Petrakian, P.M. Shabe, Trade-offs in cycle time management: hot lots, IEEE Transactions on Semiconductor Manufacturing 5 (2) (1992) pp. 101–106.