Srinivasan, M.M. (Mandyam M.)

ON EXTENDING THE SCOPE OF A BOUNDING TECHNIQUE FOR CLOSED QUEUEING NETWORKS

M. M. SRINIVASAN

Department of Industrial & Operations Engineering
The University of Michigan
Ann Arbor, Michigan 48109-2117

July 1985
Revised April 1986

Technical Report 85-25
(To appear in Large Scale
Systems Journal)

On extending the scope of a bounding technique for closed queueing networks

Abstract

Queueing network models are widely used to analyze the performance of automated manufacturing facilities and computer systems. These models aid the designers/operators of the system in evaluating the impact of various alternate configurations on the performance of these systems. In many instances, however, the exact analysis of these networks for the measures of performance may not be necessary; some bounds on the performance measures may be adequate. This has motivated research on obtaining bounds on performance measures, and a number of techniques have recently been developed. These techniques have generally obtained bounds on the throughput for networks with a single class of customers, where each node is either a single server fixed-rate type or is a delay (infinite server) type.

In this paper we extend the scope of these techniques to networks where non-delay nodes are allowed to have some service rates which depend on the number of jobs present at the nodes. In addition, efficient means are developed for calculating bounds on the mean queue lengths forming at the nodes in these networks.

1. Introduction:

The performance of complex systems such as flexible manufacturing systems (FMS) are often evaluated using queueing networks. Several studies using such analytical models of FMS have been made in the past [12,15,16]. These models aid the FMS designers in predicting the behaviour of these systems under different configurations. In the FMS context, closed queueing network models are usually preferred over open queueing network models as being more realistic [16], although open models are easier to analyze. The exact solution of these queueing networks is, however, infeasible in general unless certain assumptions are made. These assumptions give rise to a certain class of networks known as Product Form (PF) networks, or separable networks. For these PF networks, it is possible to obtain equilibrium performance measures with relatively less computational effort.

Quite often, exact solutions of these queueing networks may not be needed. One such situation arises in the design phase of a system, where the workload parameters are themselves not usually known with reasonable accuracy. Similarly, for the case when many alternate configurations are to be evaluated, obtaining exact solutions may be unnecessary. In such cases, one would like to obtain approximate solutions fairly quickly.

As an example, consider a flexible assembly system where components to be assembled move from one workstation to another, undergoing some operations at each point. Suppose that each workstation would operate ~~proportionately~~ faster depending on the number of assembly operators present at a station. Assuming, for this simple example, that all the available operators are equally capable of working at any station, the task of the supervisor here could be the assignment of additional operators to each workstation at the start of each shift, so as to maximize the throughput of assembled parts. In such a situation, the presence of a technique which can obtain quick, albeit approximate, estimates of throughputs for several operator assignments would be very helpful to the supervisor. Needless to say these approximate solutions should take relatively little time to compute and the resulting errors in approximation should preferably be bounded.

Recent years have witnessed a substantial amount of work on developing techniques for bounding the performance measures of closed queueing networks [3,5,7,11,13,14,17,19]. These bounding techniques require the PF assumption and they usually consider networks with a single class of customers where each node in the network is either single server fixed-rate type or is a delay (infinite server) type. The bounds obtained are for the cycle time (the mean system residence time) of a job, or alternately, the throughput of the system (number of job completions by the system in unit time).

In this paper, it is shown how the scope of these bounding techniques can be extended in two ways: First, bounds are obtained on the cycle time/throughput of PF networks where nodes exhibit a certain kind of load dependent behaviour. Specifically it is required here that the service rate at these nodes is a non-decreasing function of the number of customers present at the nodes. Examples of such nodes, apart from delay nodes, are multiple server nodes, and flow-equivalent nodes [2].

Secondly, an effective means of obtaining bounds on the mean queue lengths at the nodes in a PF network is also demonstrated. This method uses the Convolution algorithm [1] and a bounding technique [13] which obtains successively improving bounds (SIB) on the cycle time/throughput.

Previous work:

As noted before, work on obtaining bounds has usually been restricted to networks with fixed rate and delay nodes. The Balanced Job Bounds [19] (BJB) technique obtains a set of simple bounds on the cycle time. However, the bounds obtained here can be quite loose. A bounding scheme based on the Mean Value Analysis (MVA) algorithm [10], is the technique due to Kriz [7]. However, these bounds are effective only when the network population becomes large, at which point the network begins to behave like an open network. The SIB technique, The Performance Bound Hierarchies [5] (PBH), and the Generalized Quick Bounds [17] (GQB) are all also based on the MVA algorithm. These techniques obtain a sequence of increasingly tighter bounds at the expense of increased computational effort. The GQB technique cannot handle delay nodes and does not appear to perform significantly better than the Balanced Job Bounds. The PBH technique produces a sequence of improving bounds which converge to the exact solution, but it requires considerable computational effort to produce tighter bounds, as it uses an iterative algorithm. The SIB technique, on the other hand, obtains closed form expressions for the bounds and is thus much easier to evaluate than the PBH bounds. Further, the SIB technique produces bounds which are usually tight for all population values.

Recently, some results have been presented in [3,11,14] which obtain a lower bound on the throughput of networks with multiserver nodes. These techniques depend, in turn, on bounds obtained for a network consisting only of fixed rate and delay nodes. These approaches are reviewed in section 3.

The outline of the rest of this paper is as follows: In section 2 the SIB technique is extended to include networks where non-delay nodes are allowed to exhibit a limited form of load dependent behaviour. Section 3 then illustrates how this technique can then be applied to networks with some more general load dependent behaviour, such as networks with multi-server nodes. Section 4 presents methods for obtaining effective bounds on the mean queue lengths forming at the nodes in networks consisting only of fixed rate and delay nodes.

2. Bounds for networks with a class of load dependent nodes:

2.1 Preliminaries:

Consider a closed PF queueing network with M nodes and a customer population N. We adopt the following notation:

τ_m : mean value of service demand by a customer per visit to node m

v_m : mean number of visits to node m to service a typical request

$\mu_m(n)$: service rate at node m when n customers are present at the node

$p_m(i|N)$: marginal queue size probability of i customers at node m with network population N

$Q_m(N)$: mean queue length at node m with network population N

$w_m^*(N)$: mean residence time at node m for a request at network population N

$W(N)$: the cycle time for a typical request (the sum of mean residence times at all nodes) at network population N

λ_N : throughput of the network with network population N

Each node in the network is allowed here to display some limited type of load dependent behaviour. Specifically, for $n \geq 0$, this has the form

$$\mu_m(n) = n/(a_m + b_m \cdot n); \quad a_m, b_m \geq 0. \tag{2.1}$$

It is implicitly assumed here that $a_m+b_m > 0$. We choose to term these nodes as nodes with a Parametrized Rate function or PR nodes. In general, for the networks considered here, each node m is thus associated with a set of four data parameters: a_m, b_m, τ_m, and v_m. Note that if $a_m = 0$, equation (2.1) characterizes the behaviour of a fixed rate node, and if $b_m = 0$, this equation characterizes the behaviour of a delay node. For different values of a_m and b_m, a variety of service rate functions can be generated as indicated in Figure 2.1.

In the following discussion, unless otherwise specified, the index for any summation, is over the range 1,..,M. We also implicitly assume that we are considering network populations of $N \geq 2$.

Figure 2.1

The mean residence time at any node in the network can be written as [10]:

$$w_m^*(N) = v_m \cdot \tau_m \sum_{n=1}^{N} \left(n/\mu_m(n)\right) \cdot p_m(n-1|N-1). \tag{2.2}$$

From equations (2.1) and (2.2), we can write

$$w_m^*(N) = v_m \tau_m \sum_{n=1}^{N} (a_m + b_m \cdot n) \cdot p_m(n-1|N-1)$$

$$= I_m + L_m + L_m \cdot Q_m(N-1), \tag{2.3}$$

where

$$L_m = b_m \cdot v_m \cdot \tau_m, \tag{2.4a}$$

and

$$I_m = a_m \cdot v_m \cdot \tau_m. \tag{2.4b}$$

The values L_m and I_m are, for convenience, referred to as the 'fixed rate' load, and the 'delay' load at node m. The cycle time, W(N), is then

$$W(N) = \sum_m L_m + \sum_m I_m + \sum_m L_m Q_m(N-1). \tag{2.5}$$

Let

$$\rho_m = L_m / \left(\sum_m (L_m + I_m)\right), \tag{2.6a}$$

$$\sigma_m = I_m / \left(\sum_m (L_m + I_m)\right), \tag{2.6b}$$

$$\theta_m = \rho_m + \sigma_m, \tag{2.6c}$$

and define a sequence of terms $\{S_i\}$, i=1,2,.. by

$$S_i = \sum_m \rho_m^{i-1} \cdot \theta_m. \qquad (2.7)$$

The expression for the cycle time given by equation (2.5) is then rewritten as

$$W(N) = \Big(\sum_m (L_m+I_m)\Big) \cdot \big(1 + \phi(N-1)\big), \qquad (2.8)$$

where

$$\phi(K) = \sum_m \rho_m \cdot Q_m(K), \quad K \geq 0. \qquad (2.9)$$

For notational convenience, set

$$D_K = 1 + \phi(K). \qquad (2.10)$$

In the expression for the cycle time given by equation (2.8), the unknown term is $\phi(N-1)$. A bound on $\phi(N-1)$ then directly gives a bound on the cycle time.

2.2 The bounding technique:

The mean queue length at node m at population $K \geq 0$, is given by Little's rule [9] as:

$$Q_m(K) = \lambda_K w_m^*(K) = K w_m^*(K)/W(K). \qquad (2.11)$$

Hence, from equations (2.3),(2.8),(2.9) and (2.11), we express $\phi(N-1)$ as

$$\phi(N-1) = \sum_m \rho_m \cdot (N-1) \cdot w_m^*(N-1)/W(N-1)$$

$$= (N-1) \sum_m \frac{\rho_m^2 + \rho_m \sigma_m + \rho_m^2 Q_m(N-2)}{1 + \phi(N-2)},$$

and, using equations (2.6c) and (2.10), this expression is rewritten as

$$\phi(N-1) = (N-1)S_2 + \frac{(N-1)}{D_{N-2}} \cdot Y^1(N-2), \qquad (2.12)$$

where

$$Y^1(N-2) = \sum_m \rho_m^2 \cdot Q_m(N-2) - S_2 \phi(N-2)\big). \qquad (2.13)$$

Let

$$\Lambda_{K,i} = \prod_{j=0}^{i} \lambda_{K-j}. \qquad (2.14)$$

Theorem 2.1 first establishes an expression for the mean queue length forming at a node with network population N in terms of the throughputs at populations 1 through N.

Theorem 2.1:

The mean queue length at a node m is given by

$$Q_m(N) = \tilde{Q}_m(N) \cdot \theta_m/\rho_m, \tag{2.15}$$

where

$$\tilde{Q}_m(N) = \sum_{i=1}^{N} L_m^i \cdot \Lambda_{N,i}, \tag{2.16}$$

Proof:

From equations (2.11) and (2.3), for K > 1,

$$Q_m(K) = \lambda_K \cdot (L_m + I_m) + \lambda_K \cdot L_m Q_m(K-1),$$

and by a repeated application of the above expression for $Q_m(K)$, for $N \geq K > 1$,

$$Q_m(N) = (L_m + I_m) \cdot \lambda_N + \dots + (L_m + I_m) \cdot L_m^{N-2} \cdot \Lambda_{N,N-2} + L_m^{N-1} \cdot \Lambda_{N,N-2} \cdot Q_m(1).$$

Noting that $Q_m(1) = \lambda_1 w_m^*(1) = \lambda_1 (L_m + I_m)$, we can write

$$Q_m(N) = (L_m + I_m) \sum_{i=1}^{N} L_m^{i-1} \cdot \Lambda_{N,i}.$$

The result follows by noting that

$$(L_m + I_m)/L_m = \theta_m/\rho_m. \qquad \square$$

A set of simple upper and lower bounds on the cycle time is now established. These are termed level 1 bounds analogous to [13]. These bounds are obtained by noting that in equation (2.12), $Y^1(N-2) \geq 0$. To obtain these bounds, we need the following:

Definition 2.1:

Given a sequence $X = \{x_m\}$, m = 1, ..., M, such that $x_M \geq x_{M-1} \geq \dots \geq x_1 \geq 0$, then a sequence $Y = \{y_m\}$, m = 1, ..., M, is said to have a Positive Correspondence with X, denoted as Y(PC)X, if for some k, $1 \leq k \leq M$, $y_M \geq \dots \geq y_k \geq 0$; $y_{k-1}, \dots, y_1 \leq 0$.

Lemma 2.1:

Given sequences $U = \{u_m\}$, $X = \{x_m\}$, $Y = \{y_m\}$, $m = 1, \ldots, M$, such that

(a) $u_m \geq 0$, $m = 1, \ldots, M$,

(b) $x_M \geq \ldots \geq x_1 \geq 0$,

(c) $Y(PC)X$,

(d) $\sum_m u_m \leq 1$,

(e) $\sum_m u_m y_m \geq 0$,

then

$$\sum_m x_m u_m y_m \geq \left(\sum_m x_m u_m\right)\left(\sum_n u_n y_n\right). \tag{2.17}$$

A proof of Lemma 2.1 is given in Appendix A.

Note that if we let $\tilde{Q} = \{\tilde{Q}_m(\cdot)\}$, and $\rho = \{\rho_m\}$, $m = 1,..,M$, then $\tilde{Q}(PC)\rho$.

In equation (2.17), setting $x_m = \rho_m$, $u_m = \theta_m$, and $y_m = \tilde{Q}_m(\cdot)$, and noting that $\theta_m \cdot \tilde{Q}_m(K) = \rho_m \cdot Q_m(K)$ we get the following result which we state as

Theorem 2.2:

$$\sum_m \rho_m^2 Q_m(K) \geq \left(\sum_m \rho_m \theta_m\right) \sum_n \rho_n Q_n(K). \tag{2.18}$$

From Theorem 2.2, we can obtain a set of upper and lower bounds on D_{N-1} (and hence on the cycle time) which we term as the level 1 SI bounds. The bounds are expressed as Thèorem 2.3. A proof of Theorem 2.3 is given in Appendix B.

Theorem 2.3:

Let u be the node such that $L_u = \max_m \{L_m\}$. Then the level 1 SI bounds on D_{N-1} are given by

$$\underline{D}_{N-1} \leq D_{N-1} \leq \bar{D}_{N-1}, \tag{2.19}$$

where

$$\underline{D}_{N-1} = 1 + \xi, \tag{2.19a}$$

and

$$\bar{D}_{N-1} = 0.5[1 + \Psi + \mathrm{SQRT}((\Psi-1)^2 + 4\xi)], \qquad (2.19b)$$

with

$$\Psi = (N-1)\rho_u \qquad (2.19c)$$

$$\xi = (N-1)S_2. \qquad (2.19d)$$

Remark: If we set $I_m = 0$ for m=1,..,M, we have a network of fixed rate nodes. The Balanced Job (BJ) Bounds technique can obtain bounds on the term D_{N-1} in this case. These bounds require computation of the terms L_u and L_a, where $L_a = \sum L_m/M$, and these computations involve a total of about 2M operations. Computing the upper and lower BJ bounds on D_{N-1} then involves 2 more arithmetic operations for a total of 2M + 2 operations. Now consider the level 1 SI bounds on D_{N-1} obtained using Theorem 2.3: These bounds require computation of the terms S_2 and ρ_u, which involves a total of 3M arithmetic operations; once these terms are obtained, computing $\underline{D}_{N-1}$ requires 2 more operations, and computing $\bar{D}_{N-1}$ requires about 9 more operations for a total of 3M+11 operations. Hence the level 1 SI bounds on D_{N-1} here require about M+9 additional operations over that needed by the BJ bounds. However, for this case it is easily shown [13] that the level 1 upper and lower SI bounds are both tighter than the corresponding BJ bounds. Further, the BJ Bounds technique do not handle delay nodes efficiently.

□

Define

$$\alpha_0 = S_2, \qquad (2.20a)$$

and

$$\alpha_i = S_{i+2} - \sum_{j=0}^{i-1} S_{i+1-j}\,\alpha_j, \ i>0. \qquad (2.20b)$$

The term $Y^1(N-2)$ can now be expressed as the sum of a sequence of non-negative terms involving the throughputs at populations N-1,..,1, and the terms α_i, $i \geq 0$. This gives an expression for the term D_{N-1} which is expressed as

Theorem 2.4. This theorem makes a statement analogous to a similar statement proved in [13] and its proof is omitted here.

Theorem 2.4:

The term D_{N-1} can be written as:

$$D_{N-1} = 1 + (N-1)\cdot\left(S_2 + \frac{N-2}{D_{N-2}}\alpha_1 + \frac{(N-2)(N-3)}{D_{N-2}\cdot D_{N-3}}\alpha_2 + .. + \frac{(N-2)..1}{D_{N-2}..D_1}\alpha_{N-2}\right). \qquad (2.21)$$

Equation (2.21) for D_{N-1} is used to obtain a sequence of increasingly tighter lower bounds which we term as Successively Improving bounds of higher levels. This is achieved by considering more and more terms from the expression for D_{N-1} in the bound. A level 1 lower SI bound uses the first two terms in the expression. For example, a level 2 lower SI bound, $\underline{D}^2_{N-1}$, makes use of the first three terms from the above expression for D_{N-1} and is obtained from

$$D_{N-1} \geq 1 + (N-1)\cdot(S_2 + (N-2)\alpha_1/D_{N-1}.$$

Solving this quadratic in D_{N-1} gives:

$$\underline{D}^2_{N-1} = 0.5\cdot\left(1 + (N-1)\alpha_0 + \mathrm{SQRT}(T1)\right), \qquad (2.22a)$$

where

$$T1 = \left((N-1)\alpha_0 + 1\right)^2 + 4\cdot(N-1)(N-2)\alpha_1. \qquad (2.22b)$$

Now, a sequence of upper SI bounds is obtained by noting that from equations (2.15) and (2.16) for the mean queue lengths, we can write

$$\sum_m \rho_m^2 Q_m(k) = \rho_u\phi(k) - \sum_m (\rho_u - \rho_m)\rho_m Q_m(k)$$

$$= \rho_u\phi(k) - (k/D_{k-1})(\rho_u S_2 - S_3) - (k/D_{k-1})\sum_m (\rho_u - \rho_m)\rho_m^2 Q_m(k-1), \qquad (2.23)$$

and so on for an expression with up to k terms.

Hence, including the first two terms from the above expression for $\sum\rho_m^2 Q_m(k)$ in the expression for $Y^1(N-2)$ as given by equation (2.12), we get

$$D_{N-1} \leq 1 + (N-1)S_2 + (N-1)(\rho_u - S_2)\frac{\phi(N-1)}{D_{N-1}} - \frac{(N-1)(N-2)}{D_{N-2}\,\bar{D}_{N-3}}(\rho_u S_2 - S_3).$$

$$\leq 1 + (N-1)S_2 + (N-1)(\rho_u - S_2)\frac{\phi(N-1)}{D_{N-1}} - \frac{(N-1)(N-2)}{D_{N-1}\,\bar{D}_{N-3}}(\rho_u S_2 - S_3).$$

Here, $\bar{D}_{N-3}$ is given by the level one upper bound on D_{N-3} (equation 2.19b). Solving this resulting quadratic in D_{N-1} gives the level two upper SI bound.

The level two bounds require a little more than 3M extra operations over that required by the level one SI bounds. The increase in computations here is mainly due to the calculation of the term S_3 which requires about 3M operations.

Thus a smooth tradeoff between accuracy and computational effort is achieved by these bounds. The use of these bounds is illustrated by a few examples in this and the following sections.

Computational Remark: Suppose the network has a number of delay nodes. Then, in order to compute the exact cycle time/throughput, these nodes can be replaced by a single delay node whose load is the sum of the load at each delay node.

Example 2.1:

This is a network with 15 nodes. Table 2.1 gives the mean service time demand per visit, τ_m, and the parameters a_m, and b_m for each node. Note that nodes 1 through 5 are fixed rate nodes, and node 6 is a delay node. The mean number of visits to each node, v_m, is assumed as 1.

The throughputs were computed exactly, and also evaluated by the bounding technique using the level 5 SI bounds, for populations ranging from 1 to 50. Table 2.2 gives the results of these evaluations for some population values.

Node	1	2	3	4	5	6	7	8	9	10
Mean Time Demand	40	35	30	30	20	40	60	70	70	70
a_m	0	0	0	0	0	1	.50	.50	.40	.35
b_m	1	1	1	1	1	0	.50	.50	.60	.65

Table 2.1 Data parameters for example 2.1

Population	Throughputs		
Values	Exact	Lower Bound	Upper Bound
10	0.0128	0.0127	0.0128
20	0.0173	0.0169	0.0174
30	0.0193	0.0187	0.0196
40	0.0203	0.0197	0.0208
50	0.0209	0.0202	0.0217

Table 2.2: Throughput bounds for example 2.1.uj 0

3. Bounds for networks: nodes with non-decreasing service rates:

In this section, we consider networks where load dependent nodes (other than the PR nodes we considered in section 2) have service rates which are non-decreasing functions of their queue lengths. Examples of such nodes are multiserver nodes and flow equivalent service centers. When these nodes are present in a queueing network, exact solution algorithms require $O(MN^2)$ computations to obtain the performance measures. In this section we develop bounds on the throughput/cycle time for these networks which take O(M) computations.

Suppose it is desired to obtain a throughput bound for a given queueing network T. We now consider a new queueing network T' which is constructed as follows: for each PR node, m, in T we create a PR node in T' which has the same values for the fixed rate load, L_m, and the delay load, I_m (refer equations (2.4a) and (2.4b). However, each load dependent node which has a non-PR behaviour in T is replaced by one or more PR nodes in T'. Suppose the replacement is made such that the throughput of T' can be guaranteed to bound the throughput of T. for the range of populations being considered. Then, using the techniques developed in section 2, a lower bound on the throughput T' can be obtained with O(M) computations and this gives a lower bound on the throughput of network T. In the same manner, an alternate network, T'', can be constructed to obtain an upper bound on the throughput of T.

In section 3.1 we develop the technique for obtaining throughput/cycle time bounds for these networks. In the special case of a network with multiserver nodes, an alternate scheme to obtain throughput/cycle time bounds is developed in section 3.2.

3.1 A bounding scheme:

Consider a network T with M nodes where, for ease of discussion, we assume that exactly one of the nodes has a non-PR behaviour. The extension to cases where more than one node has a non-PR behaviour is straightforward. Without loss of generality, let the node with the non-PR behaviour be node M. We construct a new network T' that also has M nodes. The first M-1 nodes in T' have identical characteristics as the first M-1 nodes in T. The Mth node in T' has the same mean service time and the same mean number of visits as that of the Mth node in T. However, the rate functions for these two nodes can be different. Let $\mu_M(n)$ be the rate function for the Mth node in T and let $\underline{\mu}_M(n)$ be the rate function for the Mth node in T'. Now, if we choose a function $\underline{\mu}_M(n)$ that is non-decreasing in n and such that for $n \geq 0$, $\underline{\mu}_M(n) \leq \mu_M(n)$, then it is possible to show that the throughput of network T' is at most that of network T. This follows from a result obtained in [18] which in effect shows that in a network of nodes with non-decreasing rate functions decreasing the service rate at any node decreases the throughput. (Hence, from Little's rule, the cycle time of network T' is at least that of network T). Similarly, we can construct a network T'' which is identical to T' except that the non-decreasing service rate $\bar{\mu}_M(n)$ for node M is such that $\bar{\mu}_M(n) \geq \mu_M(n)$, $n \geq 0$. Then the throughput of network T can similarly be shown to be at most equal to that of network T''.

Given the service rate $\mu_M(n)$, these functions $\underline{\mu}_M(n)$ and $\bar{\mu}_M(n)$ are chosen from the parametrized class of functions given by equation (2.1), with suitable choices for a_M and b_M for the two cases. For example, suppose that node M is a multiserver node with 6 servers. However, suppose that for some reason, the effective service rate realized at the node is as follows:

$$\mu_M(n) = \begin{cases} 1+0.6(n-1); & n = 1,..,6, \\ 4; & n > 6. \end{cases} \qquad (3.1)$$

One possible way to obtain $\bar{\mu}_M(n)$ using equation (2.1) would be to solve two equations for the two unknowns $\bar{a}_M$ and $\bar{b}_M$ obtained by setting $\bar{\mu}_M(1) = 1$, and $\bar{\mu}_M(6) = 4$. This gives $\bar{a}_M$=0.9, and $\bar{b}_M$=0.1. This choice ensures that $\bar{\mu}_M(n) \geq \mu_M(n)$ for all n. It is possible to choose values of the parameters $\underline{a}_M$ and $\underline{b}_M$ for the function $\underline{\mu}_M(n)$ in a similar manner. Thus, a possible set of choices for the functions $\underline{\mu}_M(n)$ and $\bar{\mu}_M(n)$ which, for $n \leq 20$ ensures that $\underline{\mu}_M(n) \leq \mu_M(n) \leq \bar{\mu}_M(n)$ is given by

$$\underline{\mu}_M(n) = n \,/\, (0.8333 + 0.2083n), \qquad (3.2a)$$

$$\bar{\mu}_M(n) = n \,/\, (0.9000 + 0.1000n). \qquad (3.2b)$$

Figure 3.1 plots the functions $\mu_M(n)$, $\underline{\mu}_M(n)$, and $\bar{\mu}_M(n)$ for this case.

Figure 3.1

This is the approach taken here to bound the throughput/cycle time of networks where one or more nodes are allowed to have non-PR behaviour. The method is best illustrated with an example.

Example 3.1:

It is desired to estimate the throughput of a network T with 5 nodes; the first three nodes in this network are fixed rate nodes with mean service time demands of 20, 24 and 28 units respectively. Nodes 4 and 5 have load dependent service rates of the form given by equation (3.1) and the mean service time demand at each of these nodes is 120 units. The mean number of visits to each node is equal to 1. The throughput is to be estimated for a network population of 20.

To evaluate the throughput bounds two alternate networks, T' and T'', are constructed: In these networks, nodes 1, 2 and 3 have the same mean service time demands as in the original network. However, in network $T_{'}$, nodes 4 and 5 are PR nodes with rate function as given by equation (3.2a); in network T'', nodes 4 and 5 are PR nodes with rate function as given by equation (3.2b). The mean number of visits at all nodes is equal to 1 for these networks. Using the techniques developed in section 2, a level 5 lower SI bound on throughput is obtained for network T', and a level 5 upper SI bound on the throughput is obtained for network T''. Table 3.1 shows the results for some population values.

N	λ_N	$\underline{\lambda}_N$	$\overline{\lambda}_N$
5	0.013	0.012	0.013
10	0.021	0.018	0.022
15	0.026	0.022	0.027
20	0.029	0.025	0.030

Table 3.1: Throughput bounds for example 3.1.

3.2. Throughput Bounds for Networks with Multiserver nodes:

Here, we consider the special case of a network with multiserver nodes. The service rate of a multiserver node, m, with K_m multiservers is given by

$$\mu_m(n) = n; \quad n \leq K_m, \tag{3.3a}$$

and

$$\mu_m(n) = K_m; \; n > K_m. \tag{3.3b}$$

Some bounding schemes have been proposed recently [3,11,14], which obtain lower bounds on the throughput of such networks with O(M) computations. These are reviewed below:

3.2.1 Previous work:

In a recent paper, Shanthikumar and Yao [11] obtained a lower bound on the throughput for networks with multiserver nodes. This bound is obtained using some results developed by them on likelihood ratio ordering and its preservation under convolution, and essentially works as follows: Suppose each node in the network has K_m multiservers (for fixed rate servers, $K_m = 1$). Let $c_m = K_m-1$, $m = 1,..,M$, and set $c = \sum c_m$. Now, consider an alternate network which consists only of fixed rate servers. Each node in this network has the same mean service time demand as that of the corresponding node in the original network, but has a constant service rate of K_m. A lower throughput bound on this network at population N-c then gives a lower throughput bound on the original network at population N. Needless to say, we require that N-c > 0. This method is effective when N-c is not very small.

The technique proposed by Srinivasan [14] operates as follows: Consider a network T with M nodes and suppose that node M is a multiserver node with K_M multiservers. Let the throughput of the multiserver node, when considered in isolation, and with a network population N be given by $\lambda_N(M)$. Now consider a network T' where we replace this multiserver node by a set of J_M fixed rate nodes with suitably chosen fixed rate loads, where J_M is a large number. Exactly one of these J_M nodes has a fixed rate load equal to $v_M \cdot \tau_M/K_M$, and the remaining nodes each has a fixed rate load that is much smaller than $v_M \cdot \tau_M/K_M$, and such that the sum of the loads at these J_M nodes is equal to $v_M \cdot \tau_M$. Let the throughput, in isolation, of a Flow Equivalent Service Center (FESC) [2], M1, which is obtained by aggregating these J_M nodes be $\lambda_N(M1)$. It can be easily shown that $\lambda_N(M1) \leq \lambda_N(M)$. Then, using the results obtained in [18], it can be shown that the throughput of T' is bounded from above by the throughput of T. Hence, a lower bound on the throughput of T' gives a lower bound on the throughput of T. In a similar manner, it is possible to construct a network T'' where node M is replaced by a FESC, M2, with fixed rate loads chosen such that $\lambda_N(M2) \geq \lambda_N(M)$. An upper bound on the throughput of T'' gives an upper bound on the throughput of T. It is straightforward to generalize this approach to obtain throughput bounds for a network with more than one multiserver node.

Working independently, a technique similar to the above has been proposed recently by Dallery [3]. Here, each multiserver node with K_m multiservers in the original network is replaced by a set of K_m fixed rate nodes each having a mean service time demand of τ_m/K_m. The Balanced Job lower bound on the throughput of the resulting network gives the desired lower bound on the throughput of the original network.

It may be observed that if the Balanced Job Bound is used, the techniques proposed by [14] and [3] would give the same results, and would involve essentially involve the same computational effort. One major problem, however, with the BJB technique is that it does not handle networks with delay nodes efficiently. Also, the BJ bounds are often quite loose. (This was the motivation for the development of alternate schemes which obtain tighter bounds at the expense of increased computational effort). The technique developed in [3] is intended, though, to be used only in conjunction with the BJ bounds. When alternate bounding methods such as PBH or SIB are used, the lower throughput bound obtained using the technique given in [14] can be shown to be tighter than the bound obtained using the technique given in [3].

3.2.2 The scheme for networks with multiserver nodes:

The technique outlined here is essentially a modification of the technique proposed in [14]. For ease of discussion, assume as before that the network T consists of M nodes where nodes 1 through M-1 are PR nodes and node M is a multiserver node with K_M multiservers, mean service time demand τ_M, and visit ratio v_M. The extension to networks with more than one multiserver node is, again, straightforward.

Obtaining the lower throughput bound:

To obtain a lower bound on the throughput of network T, we use Theorem 3.1.

Theorem 3.1:

Consider two closed systems operating with n customers in each system. The first system consists of one node, M, having mean service time demand τ_M, and with K_M servers. The second system consists of one FESC, M1, which represents the aggregation of a network consisting of one fixed rate node with mean service time demand τ_M/K_M, and a delay node with mean service time demand $\tau_M \cdot (K_M-1)/K_M$. Let $\lambda_n(M)$ and $\lambda_n(M1)$ respectively denote the throughput of these two systems. Then

$$\lambda_n(M) \geq \lambda_n(M1).$$

Proof:

Consider the second system M1. From the Asymptotic Bound Analysis for networks with delay nodes [8], an upper bound on the throughput of this network at population n is given by $\overline{\lambda}_n(M1)$, where

$$\bar{\lambda}_n(M1) = \min(K_M/\tau_M,\ n/(\tau_M/K_M+\tau_M\cdot(K_M-1)/K_M))$$
$$= \min(K_M/\tau_M,\ n/\tau_M). \tag{3.4}$$

Now consider the throughput of the system with the multiserver node. This is given by

$$\lambda_n(M) = n/\tau_M;\ \ n \le K_M, \tag{3.5a}$$
$$\lambda_n(M) = K_M/\tau_M;\ n \ge K_M. \tag{3.5b}$$

Comparing equation (3.4) with (3.5a) and (3.5b) the result follows.

□

Now, if the multiserver node M is replaced by a FESC, M1, obtained as indicated above, then using the results obtained in [18], it can be shown that a lower bound on the throughput of the resulting network gives a lower bound on the throughput of the original network. In effect each multiserver node, m, with mean number of visits v_m is replaced here by two PR nodes both with mean number of visits v_m, and mean service time demands chosen as indicated by Theorem 3.1. Since the resulting network consists only of PR nodes, the bounds developed in section 2 can be used.

Obtaining the upper throughput bound:

A simple upper bound on the throughput is obtained using similar arguments as those used to obtain the lower throughput bound. Each multiserver node m, with K_m multiservers is replaced by a fixed rate node working at rate τ_m/K_m. [3,11,14]. In some cases, alternate replacements can do better [14].

The effectiveness of the technique developed here to obtain lower throughput bounds for these networks is illustrated with a couple of examples.

Example 3.2:

There are 11 nodes in this network. The mean service time demands at these nodes and the number of servers at these nodes are given in Table 3.2. The mean number of visits at each node is assumed to be 1. The exact throughput was calculated at various population levels. Table 3.3 gives the results of the evaluation for some population values.

Node	1	2	3	4	5	6	7	8	9	10	11
Mean Time Demand	120	160	140	100	100	90	72	30	20	16	15
No of servers	6	5	5	4	4	3	3	2	1	1	1

Table 3.2: Data parameters for Example 3.2

Following the technique developed here, each multiserver node is replaced by a fixed rate node and a delay node with mean service time demands chosen as indicated by Theorem 3.2. A simple level 1 SI bound on the resulting network gives the desired throughput bound. (As noted earlier, this requires about M more arithmetic operations than the BJ bound does). This bound is termed the SI bound in Table 3.3.

For comparison, the bound obtained using the technique outlined ~~in~~ [11] is also given (termed as SY bound in Table 3.3). The resulting network of fixed rate nodes obtained by the technique is also evaluated using the level 1 SI bound. Although this technique cannot obtain bounds at the initial population values this technique produces bounds comparable to the SI bound for large N.

Finally, the bound obtained using the technique outlined in [3] (termed as the DAL bound) is also presented in Table 3.3. The resulting network of fixed rate servers obtained here was evaluated for the throughput using the level 1 SI bound in this case too. It must be noted, however, that this technique is only intended to be used in conjunction with the BJ bound (which in any case gives a poorer estimate of throughput than the level 1 SI bound).

N	λ (actual)	SI bound	SY bound	DS ~~DAL~~ bound
10	0.011	0.011	-	0.009
20	0.021	0.018	-	0.014
30	0.026	0.023	0.016	0.018
40	0.029	0.025	0.023	0.020
50	0.030	0.027	0.026	0.022

Table 3.3: Throughput bounds for Example 3.2

Example 3.3:

As an example of an application of the bounding technique in a manufacturing environment, consider again, the problem of the assembly line supervisor who has to man 5 stations with additional operators so as to maximize throughput. Each job completed in this shop requires an exponentially distributed amount of time from each station. These stations are already manned by one or more operators. Assume that each station would operate proportionately faster depending on the number of operators assigned to it. Due to certain limitations, 30 jobs can be in the system at any time, and completion of one job immediately triggers another into the system. The five stations have mean service time demands of 40, 32, 9, 8, and 6, respectively, and the mean number of visits to each station is 1. In addition, there is some time taken for material movement between stations which is modelled by a delay station, and for a typical cycle of operations, this takes 65 units of time. The number of operators assigned at present to each station (Configuration 0) are, respectively, 4, 3, 1, 1, and 1. The supervisor would like to obtain quick estimates of the improvement in throughput resulting from adding one extra operator either at station 1 (Configuration 1) or at station 2 (Configuration 2). Using the bounding technique developed above, the level 2 lower and upper bounds on the throughput for the three resulting configurations are calculated and presented in Table 3.4.

Configuration	Lower bound	Upper bound	Exact throughput
0	0.0822	0.1092	0.0889
1	0.0845	0.1145	0.0913
2	0.0885	0.1174	0.0952

Table 3.4: Throughput bounds for example 3.3

For comparison, the exact throughput values are also presented in Table 3.4 To compare the computational effort involved, calculating the exact throughput for the three configurations involved a total of 62 milliseconds. The bounding technique took less than 2 milliseconds to obtain all the bounds. However, it is important to note that the bounding technique here would probably only require a hand held calculator.

4. Bounds on mean queue lengths:

Techniques now exist for obtaining reasonably tight bounds on the throughput for networks with PR nodes. These bounds can be obtained with considerably less computational effort than that required in order to compute the throughput/cycle time exactly. This suggests the possibility of achieving good bounds on the mean queue lengths which form at the nodes. Needless to say, these bounds should require little computational effort compared to that required to obtain the exact values.

In section 4.1, some bounds are presented which are very easy to evaluate once bounds on the throughput have been obtained. These bounds appear adequate for nodes which do not experience very high utilizations (e.g., over 80%). For nodes with high utilizations, an alternate bounding method is developed in section 4.2. This latter method is based on an application of the convolution algorithm [1] and requires a relatively larger number of arithmetic operations (O(M) operations for each such node) to obtain the mean queue length bounds. The discussion below is restricted, for simplicity, to networks with fixed rate nodes. It is easily extended to networks with delay nodes and nodes with parametrized service rate functions of the type covered in section 2.

4.1 Obtaining simple bounds on mean queue length:

For networks with only single server fixed-rate nodes nodes, the mean queue length at a fixed-rate node, m, is given from equations (2.15) and (2.16) as:

$$Q_m(N) = \sum_{i=1}^{N} L_m^i \Lambda_{N,i-1}. \qquad (4.1)$$

For the networks considered here, it can be shown that for K > 0, we must have $\lambda_K \geq \lambda_{K-1}$. Hence

$$Q_m(N) \leq \sum_{i=1}^{N} L_m^i \lambda_N{}^i$$

$$= U_m(N) \frac{1 - (U_m(N))^{N+1}}{1 - U_m(N)}, \qquad (4.2)$$

where

$$U_m(N) = L_m \lambda_N, \qquad (4.3)$$

is the utilization of node m. Let $\overline{\lambda}_N$ denote an upper bound on λ_N and let $\overline{U}_m(N) = L_m \overline{\lambda}_N$ denote the corresponding upper bound on the utilization at node m. Then it is easily seen that

$$Q_m(N) \leq \overline{U}_m(N) \frac{1 - (\overline{U}_m(N))^{N+1}}{1 - \overline{U}_m(N)}. \qquad (4.4)$$

Let L_u denote the load at the node(s) with the highest load. As the network population increases, the network begins to behave like an open network with mean arrival rate $1/L_u$. For this corresponding open network, any node m with load $L_m < L_u$ will experience a mean queue length $Q_m = U_m/(1-U_m)$ where $U_m = L_m/L_u$ would be the utilization of this node in the open network. Hence, for large N the upper bound on mean queue lengths, as given by equation (4.4) would begin to get increasingly closer to the exact value for these nodes. Let there be k nodes with load L_u. For these nodes, at large N, an upper bound is:

$$\overline{Q}_u(N) = 1/k \; (N - \sum_{m \neq u} \underline{Q}_m(N)) \qquad (4.5)$$

where $\underline{Q}_m(N)$ is a lower bound on $Q_m(N)$.

A lower bound on $Q_m(N)$ is given by Theorem 4.1. A proof of the theorem is given in Appendix C.

Theorem 4.1:

In a closed PF network with only single server fixed rate nodes and delay nodes, the mean queue length, $Q_m(N)$, at a fixed rate node m is bounded from below by the term $f(\underline{U}_m(N))$, where $\underline{U}_m(N)$ is a lower bound on $U_m(N)$ and

$$f(\underline{U}_m(N)) = \underline{U}_m(N) \cdot \frac{(N-1)(1 - \underline{U}_m(N)) + 1}{(N-1)(1 - \underline{U}_m(N))^2 + 1}. \qquad (4.6)$$

4.2 Obtaining tighter bounds on mean queue lengths:

Although the bounds presented in Section 4.1 are easy to evaluate, they can be quite loose for nodes which have high utilizations. This is apparent since the bounds in Section 4.1 were obtained as follows: (i) Bounds were developed to express the mean queue lengths at the nodes in terms of their utilizations and (ii) Some bounds on these utilizations themselves were used in the expressions developed in (i) above.

It is, however, possible to obtain an exact closed form expression for the mean queue lengths in terms of a throughput with some additional computation. This is achieved by means of the Convolution algorithm [1]. For ease of presentation, only networks with fixed rate servers are considered here. The extension to networks with delay servers is straightforward.

The Convolution algorithm for a network T with M fixed rate single server nodes gives an expression for the normalizing constant g(N,M) as

$$g(N,M) = g(N,M-1) + L_m g(N-1,M). \tag{4.7}$$

The throughput for this network, λ_N, is then given as

$$\lambda_N = g(N-1,M)/g(N,M). \tag{4.8}$$

Suppose it is desired to obtain bounds for the mean queue length at some designated node. Without loss of generality, let this be node M. The mean queue length at this node can be shown to be [4]

$$Q_m(N) = \sum_{i=1}^{N} L_m^i \, g(N-i,M)/g(N,M) \quad . \tag{4.9}$$

Now consider an augmented network $T^{(M)}$ with M+1 nodes where nodes 1 through M have the same loads as in T and node M+1 had a load L_M. Let $\lambda_N^{(M)}$ be the throughput of this augmented network. Theorem 4.1 then obtains an expression for $Q_M(N)$ in terms of $\lambda_N^{(M)}$.

Theorem 4.2

Given a network, T, of M single server fixed rate nodes, the mean queue length at the M^{th} node, $Q_m(N)$ is given by

$$Q_M(N) = \frac{\lambda_N^{(M)} \cdot L_M}{1 - \lambda_N^{(M)} \cdot L_M} , \tag{4.10}$$

where $\lambda_N^{(M)}$ is the throughput of the network T augmented by one additional node, M+1, with load L_M.

Proof:

The normalization constant of the augmented network, $T^{(M)}$, is given by

$$g(N,M+1) = g(N,M) + L_M \, g(N-1,M+1), \tag{4.11}$$

and the throughput of $T^{(M)}$ is

$$\lambda_N^{(M)} = g(N-1,M+1)/g(N,M+1). \tag{4.12}$$

By repeated application of (4.11) in equation (4.12), we get

$$\begin{aligned}\lambda_N^{(M)} &= \frac{g(N-1,M) + L_M\, g(N-2,M) + \ldots + L_M^{N-1} g(0,M)}{g(N,M) \quad + L_M\, g(N-1,M) + \ldots + L_M^N g(0,M)} \\ &= \lambda_N \frac{1 + Q_M(N-1)}{1 + Q_M(N)}, \end{aligned} \tag{4.13}$$

where the last equality is obtained using equations (4.8) and (4.9).

From the MVA theorem an expression relating $Q_M(N)$ with $Q_M(N-1)$ can be obtained as:

$$Q_M(N) = \lambda_N L_M + \lambda_N L_M Q_M(N-1). \tag{4.14}$$

Substituting the expression for $Q_M(N-1)$ resulting from (4.14) into equation (4.13), and simplifying, the desired result in obtained.

□

Corollary 4.1:

Let $W^{(M)}(N)$ be the cycle time of the network, $T^{(M)}$. Then

$$Q_M(N) = \frac{N L_M}{W^{(M)}(N) - NL_M}. \tag{4.15}$$

Proof:

The proof is immediate from an application of Little's rule [9].

□

To illustrate how this method extends directly to include, for example, delay nodes, consider a network with M single server fixed rate nodes and one delay node. Let this delay node be labelled node 0, and have a load L_0. Then the throughput for this network is given by

$$\lambda_N = (N/L_0) \cdot h(N-1,M)/h(N,M), \tag{4.16}$$

where $h(\cdot, \cdot)$ is obtained recursively from [4] as

$$h(n,m) = h(n,m-1) + (nL_m/L_0) \cdot h(n-1,m); \quad n \geq 1,\ m \geq 1; \tag{4.17}$$

with

$h(0,m) = h(n,0) = 1; \quad n,m > 0.$

It is easily seen from equations (4.16) and (4.17), that the mean queue length at a fixed rate node in this network is given in terms of an augmented network, as before, by equation (4.10) or (4.15). The mean queue length, $Q_0(N)$, at the delay node is directly obtained from

$$Q_0(N) = \lambda_N \cdot L_0.$$

In general, in order to use the approach outlined above, a new augmented network is to be constructed for each fixed rate node with a distinct load, for which mean queue length bounds are desired. This augmented network should then be analyzed, using a bounding technique, to obtain bounds on its cycle time. Equation (4.15) can then be used to obtain the bounds of the mean queue length.

Now suppose the SIB technique is used. Assume that the level two bounds on the throughput have been obtained. This means that the terms S_i, $i = 2,3$ must have been calculated, where S_i is defined by equation (2.7). Now, to obtain mean queue length bounds for a node, n, $n \leq M$, with load L_n, the augmented network with $M+1$ nodes is constructed. The values of the relative utilizations $\rho_m^{(n)}$, $m = 1, \ldots, M+1$, for this augmented network, are given by

$$\rho_m^{(n)} = \frac{\rho_m}{L + L_n} \cdot L_m, \quad m = 1, \ldots, M+1.$$

Let

$$S_i^{(n)} = \sum_{i=1}^{M+1} (\rho_m^{(n)})^i, \quad i = 2,3. \tag{4.18}$$

$$= (S_i + \rho_n{}^i)/(1 + \rho_n)^i. \tag{4.19}$$

Hence the terms $S_i^{(n)}$ are easy to evaluate, given the values for S_i and the relative utilizations. The lower bound for the augmented network is then given by equation (2.22a) where the terms α_0 and α_1 are replaced by $\alpha_0^{(n)}$ and $\alpha_1^{(n)}$. The terms $\alpha_i^{(n)}$ are defined as

$$\alpha_0^{(n)} = S_2^{(n)},$$

$$\alpha_i^{(n)} = S_{i+2}^{(n)} - \sum_{j=0}^{i-1} S_{i+1-j}^{(n)}\alpha_j, \quad i > 0.$$

Using this level 2 bound on cycle time in equation (4.15), an upper bound on the mean queue length at node n is obtained.

The use of the bounding techniques developed in this section is illustrated below with a few examples. In each case, the approach taken is to use bounds developed in Section 4.1 for nodes with lower utilizations, and use the bounds from Section 4.2 for nodes with higher utilizations. The basis for determining which to use was, somewhat arbitrarily, set as follows: from the upper bound on throughput, the upper bound on the utilizations at the nodes was determined using equation (4.3). Nodes with this upper bound value ≤ 0.80 were evaluated for their mean queue length bounds by the method of Section 4.1, while the other nodes used the method of Section 4.2. The SIB technique was used in all test cases, and the level 5 SI bounds were used to calculate the throughput bounds.

For comparison, exact values for mean queue lengths were also evaluated. The test cases were run on an AMDAHL 5860 running the MTS. The time taken by the exact analysis and the bounding technique were recorded in each case.

Example 4.1

This is the example given in [19]. There are 49 single server fixed rate nodes, with loadings as follows: 1 node at 21, 10 nodes at 20, 4 nodes at 10, 4 nodes at 5, 3 nodes at 4, 11 nodes at 3, 1 node at 2, and 15 nodes at 1 for a total load of 343. The bounds were evaluated at a population of 100. The mean queue length bounds and the exact values for mean queue lengths for each node with a distinct load are given in Table 4.1. Even though many nodes had identical loads, bounds were evaluated on all 49 nodes. The exact analysis took 17 milliseconds, while the bounding technique took 2 milliseconds.

	Mean Queue Lengths		
Load at node	Exact	Lower bound	Upper bound
21	12.865	10.131	15.754
20	7.957	6.605	8.985
10	0.811	0.768	0.835
5	0.289	0.279	0.295
4	0.218	0.212	0.223
3	0.155	0.151	0.158
2	0.098	0.096	0.100
1	0.047	0.046	0.048

Table 4.1: Mean queue length bounds for example 4.1

Example 4.2:

This example is taken from [5]. There are 50 nodes with loads as follows: 1 node at 20, 2 nodes at 19, 5 nodes at 18, 5 nodes at 15, 5 nodes at 10, 8 nodes at 7, 8 nodes at 5, 8 nodes at 4, 8 nodes at 2, for a total load of 417. The bounds were evaluated at a network population of 50. The exact analysis required 9 milliseconds while the bounding technique required 2 milliseconds. Table 4.2 compares results for some of these nodes, Again, for comparison purposes, bounds were evaluated for all 50 nodes.

Example 4.3:

The final example here is a very unbalanced network with 20 nodes, and loads as follows: 2 nodes at 30, 1 node at 17, 3 nodes at 14, 1 node at 12, 2 nodes at 11, 1 node at 10, 2 nodes at 8, 1 node at 7, 2 nodes at 5, 3 nodes at 4, 1 node at 3, and 1 node at 1 for a total load of 212. Table 4.3 compares results for some of the nodes. The comparisons were made for a population of 45. The exact analysis required about 3.3 milliseconds while the bounding technique required about 0.7 millisecond of CP time.

	Mean queue lengths		
Load at node	Exact	Lower bound	Upper bound
20	5.181	4.484	5.691
19	4.027	3.531	4.329
18	3.207	2.464	3.749
15	1.781	1.544	1.923
10	0.753	0.701	0.781
7	0.431	0.409	0.443
5	0.274	0.262	0.281
4	0.208	0.200	0.213

Table 4.2: Mean queue length bounds for example 4.2

	Mean queue lengths		
Load at node	Exact	Lower bound	Upper bound
30	18.540	16.650	18.681
17	1.228	1.145	1.308
14	0.831	0.796	0.875
12	0.637	0.617	0.667
10	0.480	0.468	0.500
8	0.350	0.344	0.364
7	0.294	0.289	0.304
4	0.149	0.147	0.154

Table 4.3: Mean queue length bounds for example 4.3

Remark:

In many instances, it may not be necessary to evaluate mean queue lengths at all nodes. (This is certainly true if many nodes have the same loads.) It is especially in such cases that a bounding technique would have an edge over the exact analysis methods such as the MVA algorithm which would have to compute these measures for all nodes, in any case, and at all intermediate population values. In the above examples, however, to compare the computational effort involved, the mean queue length bounds were calculated for all the nodes.

□

5. Conclusions:

In many situations, the use of bounding techniques for analyzing queueing networks is justified. The scope of bounding techniques for closed PF queueing networks has been extended to enable throughput bounds to be obtained for networks in which the nodes are allowed to display some limited forms of load dependent behaviour. The application of these bounding techniques in analyzing flexible manufacturing systems has been illustrated by some examples. A simple, but effective, means of obtaining bounds on the mean queue lengths that form at the nodes in such networks has also been presented.

Appendix A:

Proof of Lemma 2.1:

Lemma 2.1:

Given sequences $U = \{u_m\}$, $X = \{x_m\}$, $Y = \{y_m\}$, $m = 1, \ldots, M$, such that

(a) $u_m \geq 0$, $m = 1, \ldots, M$,

(b) $x_M \geq \ldots \geq x_1 \geq 0$,

(c) $Y(PC)X$,

(d) $\sum_m u_m \leq 1$,

(e) $\sum_m u_m y_m \geq 0$,

then

$$\sum_m x_m u_m y_m \geq (\sum_m x_m u_m)\,(\sum_n u_n y_n). \tag{A1}$$

Proof:

Let $\gamma = \sum_m u_m y_m$. We need to show that $\sum_m x_m u_m (y_m - \gamma) \geq 0$.

Since $\sum_m u_m \leq 1$, it must be true that

$$\sum_m u_m y_m \geq (\sum_m u_m) \cdot (\sum_n u_n y_n). \tag{A2}$$

Further, since $\gamma \geq 0$, and $u_m \geq 0$, there exists some $i \leq M$, such that $y_m \geq \gamma$ for all $m \geq i$; and $y_m < \gamma$, $m < i$. Since $x_i \geq 0$, multiplying both sides of (A2) by x_i,

$$\sum_m x_i u_m y_m \geq (\sum_m x_i u_m)\gamma.$$

Hence

$$0 \leq \sum_m x_i u_m (y_m - \gamma) = \sum_{m=1}^{i-1} x_i u_m (y_m - \gamma) + \sum_{m=i}^{M} x_i u_m (y_m - \gamma)$$

$$\leq \sum_{m=1}^{i-1} x_m u_m (y_m - \gamma) + \sum_{m=i}^{M} x_i u_m (y_m - \gamma) \leq \sum_{m=1}^{i-1} x_m u_m (y_m - \gamma) + \sum_{m=i}^{M} x_m u_m (y_m - \gamma),$$

where the first (respectively second) inequality follows from the fact that $y_m - \gamma < 0$ (respectively ≥ 0) for all $m < i$ (respectively $\geq i$). The result follows. □

Appendix B:

Proof of Theorem 2.3:

Theorem 2.3:

Let u be the node with maximum load. Then the level 1 bounds on D_{N-1} are:

$$\underline{D}_{N-1} \leqq D_{N-1} \leqq \bar{D}_{N-1}, \qquad \text{(B1)}$$

where

$$\underline{D}_{N-1} = 1 + \xi, \qquad \text{(B1a)}$$

$$\bar{D}_{N-1} = 0.5[1 + \Psi + \text{SQRT}((\Psi-1)^2 + 4\xi)], \qquad \text{(B1b)}$$

with

$$\Psi = (N-1)\rho_u, \qquad \text{(B1c)}$$

$$\xi = (N-1)S_2. \qquad \text{(B1d)}$$

Proof:

Equation (2.11) gives

$$D_{N-1} = 1 + \phi(N-1) = 1 + (N-1)S_2 + (N-1) \cdot Y^1(N-2)/D_{N-2}.$$

By construction, $S_2 \geq 0$. Theorem 2.2 gives $Y^1(N-2) \geq 0$. So a lower bound is:

$$\underline{D}_{N-1} = 1 + (N-1)S_2.$$

To obtain $\bar{D}_{N-1}$, replace $Y^1(N-2)$ by a larger factor, i.e., from equation (2.13),

$$Y^1(N-2) = \rho_u \sum \rho_m Q_m(N-2) - \rho_m \cdot \theta_m \, \phi(N-2))$$
$$= (\rho_u - S_2)\phi(N-2).$$

Hence

$$D_{N-1} \leq 1 + \xi + (\Psi - \xi) \, \phi(N-2)/D_{N-2}.$$

Since $S_2 = \sum_m \rho_m\theta_m \leq \rho_u \sum_m \theta_m \leq \rho_u$, it is clear that $(\Psi - \xi) \geq 0$.

From known results on the 'monotonic' behavior [18] of networks, it is possible to show $D_{N-1} \geq D_{N-2}$. Hence, noting that $D_k = 1 + \phi(k)$, it is seen that $\phi(N-2)/D_{N-2} \leq \phi(N-1)/D_{N-1}$. So, noting that $D_{N-1} = 1 + \phi(N-1)$,

$$D_{N-1} \leq 1 + \xi + (\Psi - \xi)\phi(N-1)/D_{N-1} = 1 + \xi + (\Psi - \xi)(D_{N-1} - 1)/D_{N-1}. \qquad \text{(B2)}$$

Solving this resulting quadratic in D_{N-1}, we have the desired upper bound.

□

APPENDIX C

Proof of Theorem 4.1

Theorem 4.1:

In a closed PF network with only single server fixed rate nodes and delay nodes, the mean queue length, $Q_m(N)$, at a fixed rate node m is bounded from below by the term $f(\underline{U}_m(N))$, where $\underline{U}_m(N)$ is a lower bound on $U_m(N)$ and

$$f(\underline{U}_m(N)) = \underline{U}_m(N) \cdot \frac{(N-1)(1 - \underline{U}_m(N)) + 1}{(N-1)(1 - \underline{U}_m(N))^2 + 1}. \qquad (C1)$$

Proof:

From Little's rule [9] we can write:

$$\lambda_N \cdot W(N)/N = \lambda_{N-i}W(N-i)/N-i, \quad 0 \leq i \leq N.$$

A direct consequence of the results in 'monotonicity' in PF queueing networks [18] gives the relation

$$W(N-i) \leq W(N), \quad 0 \leq i \leq N.$$

Hence

$$\lambda_{N-i} \geq \lambda_N(N-i)/N, \quad 0 \leq i \leq N. \qquad (C2)$$

From equations (C2), (4.1) and (4.3), for a fixed rate node m, we can write

$$Q_m(N) > \sum_{i=1}^{N} \prod_{j=1}^{i} U_m(N) \frac{N-j+1}{N} \geq \sum_{i=1}^{N} \prod_{j=1}^{i} \underline{U}_m(N) \frac{N-j+1}{N}.$$

For notational ease, set $U = \underline{U}_m(N)$. Then we can write

$$Q_m(N) > g(U,N) = \sum_{i=1}^{N} U^i \prod_{j=0}^{i-1} \frac{N-j}{N}. \qquad (C3)$$

Noting that $\prod_{j=0}^{N} (1-j/N) = 0$, the function g(U,N) can be rewritten as

$$g(U,N) = \sum_{i=1}^{N} U^i \prod_{j=0}^{i-1} (1 - j/N) + U^{N+1} \prod_{j=0}^{N} (1 - j/N)$$

$$= U + U \sum_{i=2}^{N+1} U^{i-1} \prod_{j=0}^{i-1} (1 - j/N),$$

and, setting $k = i - 1$,

$$g(U,N) = U + U \sum_{k=1}^{N} U^k \prod_{j=0}^{k}(1 - j/N) = U + U \sum_{k=1}^{N} U^k(1 - k/N) \prod_{j=0}^{k-1}(1 - j/N)$$

$$= U + U\,g(U,N) - U/N\,h(U,N), \qquad \text{(C4)}$$

where

$$h(U,N) = \sum_{k=1}^{N} k\,U^k \prod_{j=0}^{k-1}(1 - j/N)$$

$$= \sum_{k=1}^{N} U^k \prod_{j=0}^{k-1}(1 - j/N) + \sum_{k=1}^{N} (k-1)U^k \prod_{j=0}^{k-1}(1 - j/N)$$

and so, using equation (C3), and after some elementary algebra,

$$h(U,N) = g(U,N) + U \sum_{i=1}^{N-1} i\,U^i \prod_{j=0}^{i}(1 - j/N)$$

$$= g(U,N) + U^2(1 - 1/N) + U(1 - 1/N) \sum_{i=2}^{N-1} iU^i \prod_{j=0}^{i}(1 - j/N)$$

$$< g(U,N) + U(1 - 1/N)\left(U + \sum_{i=2}^{N-1} iU^i \prod_{j=1}^{i-1}(1 - j/N)\right)$$

$$< g(U,N) + U(1 - 1/N)\left(U + \sum_{i=2}^{N} iU^i \prod_{j=1}^{i-1}(1 - j/N)\right)$$

$$= g(U,N) + U(1 - 1/N)\,h(U,N).$$

Hence

$$h(U,N) < g(U,N)/(1 - U + U/N). \qquad \text{(C5)}$$

Substituting the inequality for $h(U,N)$ given by (C5) into equation (C4),

$$g(U,N) > U + U\,g(U,N) - \frac{U\,g(U,N)}{N\,(1 - U + U/N)}\,.$$

Collecting terms and simplifying the above expression gives

$$g(U,N) > U\,\frac{(N-1)(1-U) + 1}{(N-1)(1-U)^2 + 1}. \qquad \text{(C6)}$$

Equations (C3) and (C6) give the desired result.

□

REFERENCES

1. BUZEN, J.P., "Computational algorithms for closed queueing networks with exponential servers", C. ACM, v 16, no 9, Sept 1973, pp. 527-531.

2. CHANDY, K.M., HERZOG, U., and WOO, L., "Approximate analysis of general queueing networks", IBM J. Res. Dev., v 19, pp. 43-49.

3. DALLERY, Y., and SURI, R., "Approximate disaggregation and performance bounds for queueing networks with multiple-server stations", Joint Performance '86 and ACM Sigmetrics 1986 conference, Raleigh, 1986.

4. DENNING, P.J., and BUZEN, J.P., "The operational analysis of queueing network models", Computing Surveys, v 10, 3, Sept 1978, pp. 225-262.

5. EAGER, D.L., and SEVCIK, K.L., "Performance bound hierarchies for queueing networks", ACM TOCS, vol 1, No 2, May 1983, pp. 99-115.

6. GORDON, W.J., and NEWELL, G.F., "Closed queueing systems with exponential servers", Operations Research, v 15, no 2, pp. 254-265.

7. KRIZ, J., "Throughput bounds for closed queueing networks", Performance evaluation, v 4, 1984, pp. 1-10.

8. LAZOWSKA, E.D., ZAHORJAN, J., GRAHAM, G.S., and SEVCIK, K.C., "Computer system analysis using queueing network models", Prentice Hall, 1984.

9. LITTLE, J.D.C., "A proof of the queueing formula $L = \lambda W$". Operations Research, 9, 1961, pp. 383-387.

10. REISER, M., and LAVENBERG, S.S., "Mean value analysis of closed multichain queueing networks", J. ACM, 27, 2, April 1980, pp. 313-322.

11. SHANTHIKUMAR, J.G., and YAO, D.D., "Throughput bounds for closed queueing networks with queue-dependent service rates", July 1985.

12. SOLBERG, J.J., "Quantitative design tools for computerized manufacturing systems", Proc. Sixth North American Metalworking Committee, Florida, April 1978.

13. SRINIVASAN, M.M., "Successively improving bounds on performance measures for product form queueing networks", Tech. Report 85-2, Department of I & OE, University of Michigan, 1985.

14. SRINIVASAN, M.M., "Bounds on performance measures for closed queueing networks: networks with multiserver nodes", Tech. Report 85-37, Department of I & OE, University of Michigan, 1985

15. STECKE, K.E., "Production planning problems for flexible manufacturing systems", Ph.D. dissertation, School of Industrial Engineering, Purdue University, W. Lafayette, Indiana, 1981.

16. SURI, R., and Hildebrant, R.R., "Modelling flexible manufacturing systems using mean value analysis", Tech. Report, August 1983.

17. SURI, R., "Generalized quick bounds for performance of queueing networks", Computer Performance, vol 5, No 2, June 1984, pp. 116-120.

18. SURI, R., "A concept of monotonicity and its characterization for closed queueing networks", Operations Research, May/June 1985, v 33, no 3, pp. 606-624.

19. ZAHORJAN, J., SEVCIK, K.C., EAGER, D.L., and GALLER, B., "Balanced job bound analysis of queueing networks", C. ACM, 25, 2, Feb 1982, pp. 134-141.

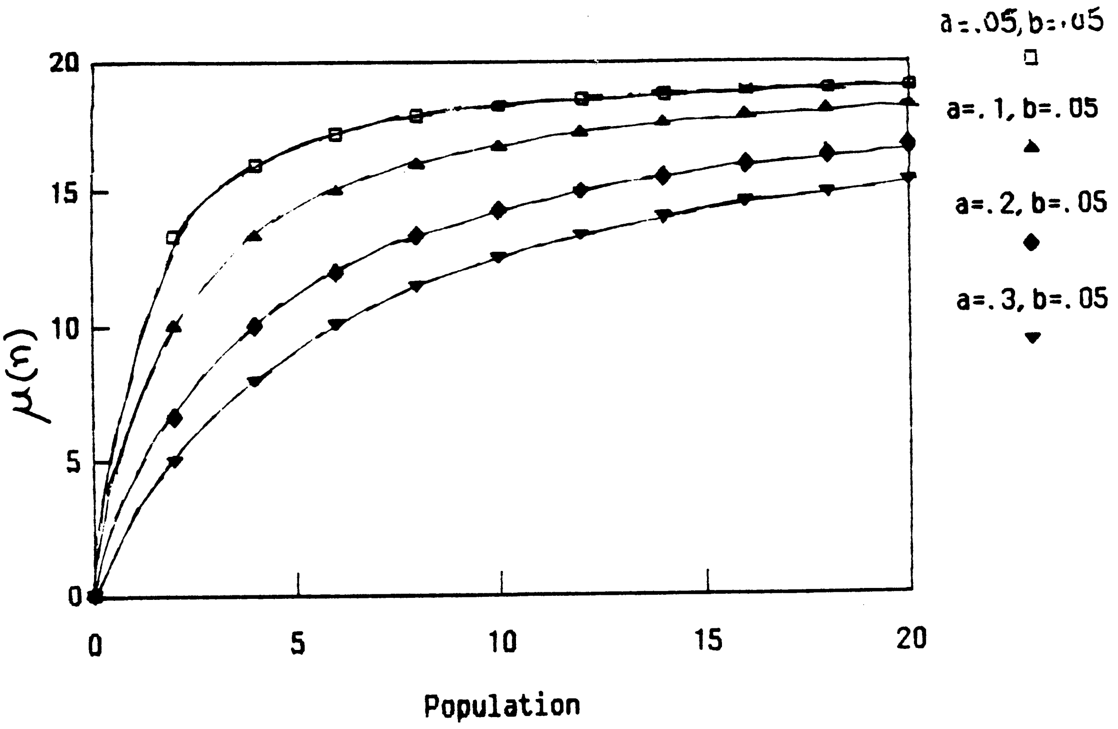

Figure 2.1: Service rate functions for some PR nodes

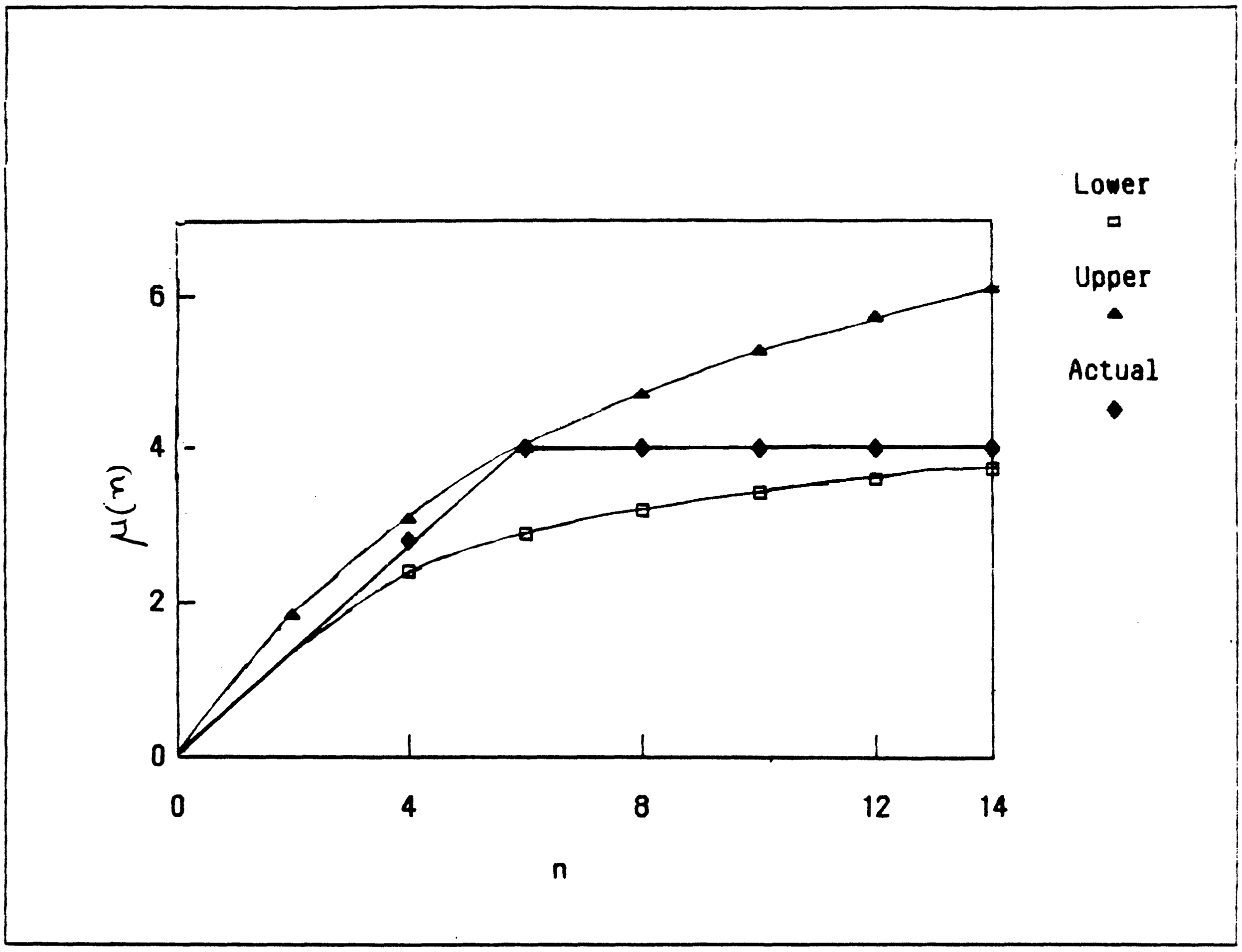

Figure 3.1: Bounding service rate for a node with rate given by equation 3.1