Approximate Analysis of Open Network of GE/GE/m/N Queues with Transfer Blocking

Hema Tahilramani
ECSE Deptt
Rensselaer Polytech Inst
Troy NY 12180
tahilh@rpi.edu

D. Manjunath
Deptt of Elecl Engg
IIT, Bombay
Mumbai INDIA
dmanju@ee.iitb.ernet.in

Sanjay K. Bose
School of EEE
Nanyang Tech Univ
Singapore
skb@ieee.org

Abstract

In this paper we describe an approximate method for the analysis of an open network of finite capacity queues. Finite buffer capacities at the nodes introduce blocking of jobs that finish service at a node and find that the destination node is full at that time. When this happens, we assume that the blocked job is held at the server of the queue where it just completed service, blocking that server until the destination can accommodate it. This is called transfer blocking or blocking after service. We also assume that an external arrival that finds a full queue is lost.

We consider an open queuing network with interarrival times of external arrivals and service times at each queue having generalized exponential (GE) distribution. Queues can have finite buffers. To solve this system we augment the network by adding a "holding node" for every stream that can be blocked to hold the blocked jobs during the period corresponding to them blocking the server. The mean time spent in the holding node will be equal to that spent while being blocked. Also, to account for the increased service time of a blocked job as seen by customers behind it in the queue, the service times of these customers need to be increased. We thus use an iterative procedure to converge on to the parameters of the GE distributions of the inter-arrival and service time distributions. Results from our analysis are compared against simulations and they compare very well.

1. Introduction

Queueing networks have been widely used for performance modeling of computer systems, telecommunication networks and manufacturing systems. Queues with finite buffer space are a more realistic model for most of the above systems. Queueing networks with finite buffer queues can cause blocking and we can define many models for the behavior of a blocked job. We are specifically interested in type-1 or transfer blocking in open queuing networks.

Queuing networks with transfer blocking can be used in modeling communication, computer and manufacturing systems. For example, a disk-to-tape backup system or mass storage system can be modelled by a closed queuing network with transfer blocking [9]. The following is a good example for an open network with transfer blocking. Consider a FDMA cellular communication system with a set of channels allocated to each base station. Adjoining cells use different sets of channels to avoid adjacent channel interference. Typically, channel allocation is static. A call on a mobile phone leaving one cell and entering a neighboring cell must be transferred to the base station of this neighboring cell. This is called handoff. We assume that the minimum acceptable signal level is low enough that the calls are not forced to terminate due to unacceptable signal levels. In the situation when there is no channel available at the neighboring cell, the call continues to use the channel of the original cell though at a poor quality (handoff blocking). This system may be modeled as an open queueing network with the time spent in a cell corresponding to the service time at the node. New calls arriving when all the channels at the cell are occupied are dropped (new call blocking). The number of voice channels allocated to a cell corresponds to the number of servers at the queue. When more than one calls are handoff blocked, they can be handed off as soon as the channels become available at the destination nodes. Essentially here, the number of waiting spaces is same as the number of servers. This example will be studied in Section 4 when we deal with numerical examples. For another example of the use of an open queuing network with transfer blocking, consider a serial manufacturing system which can be modelled as tandem network of finite buffer queues. Each machine is modeled as a server with the number of buffers equal to the amount of storage space in front of the machine. Raw parts enter the machine facility, move through some set of machines, undergoing some processing at each machine and a finished product leaves the manufacturing system. The queuing network model for such a system has been considered in [11]. The processing times at each machine have been considered to be random owing to variability introduced due to set-up times, down times of the machine,material shortage, maintainenece etc. If a part cannot proceed to the next machine due to its downstream buffer being full, remains in the machine where it has been processed until buffer space becomes available at the downstream machine.

Exact solutions for finite capacity networks are often intractable and computationally complex. Various approximate analytic or numerical solution techniques have been proposed for the analysis of such networks. Some such approximate analysis techniques have been proposed in [6, 3, 7, 1, 2]. Of these, [3, 7, 2] study open networks with transfer blocking. In [2] an analysis technique is presented for open networks under transfer blocking, but the restriction is that all nodes must have single servers and it assumes that all service times and inter-arrival times are exponentially distributed. In [7] analyse acyclic networks with Poisson arrivals and exponentially distributed service times but the technique used here cannot be used fo general topologies. In [3] a framework has been defined to analyze open networks with transfer blocking where each node is a single server queue. In this paper we generalise this technique by allowing the nodes in the network to have multiple servers. In [6], Kouvatsos and Xenios propose a very general approximation technique for open and closed networks with repetitive service blocking with both fixed and random destinations. They introduce the generalised exponential (GE) distributions that can be used to model distributions where mean and the standard deviation are not equal. We will use many of the results from [6] and [5] in our generalised approximation technique for analyzing open network of queues with transfer blocking. In the next section we introduce and motivate the generalised exponential distribution and also define the model and the parameters. In Section 3 we describe the analysis technique and in Section 4 we present numerical results in which we consider example networks that are analysed using our technique and the results compared against simulation results to analyse the goodness of our approximation technique.

2. Preliminaries and Notation

We first discuss the generalized exponential distribution and its properties. We then describe transfer blocking in finite buffer queueing network and introduce the notation.

2.1. Generalized Exponential (GE) Distribution

The generalized exponential (GE) distribution is a two parameter distribution which may be used to approximate distributions by matching two moments (as opposed to just one moment with the exponential distribution). The probability density function, $f(x)$ is given by

$$f(x) = \begin{cases} \alpha & \text{for } x = 0 \\ (1-\alpha)^2 \lambda e^{-\lambda(1-\alpha)x} & \text{for } x > 0 \end{cases} \qquad (1)$$

with $\alpha \in [0,1)$. The mean of this distribution is $1/\lambda$ and the squared coefficient of variation (SqV), the ratio of the variance to the square of the mean, is $(1+\alpha)/(1-\alpha)$. It is easy to see that this distribution is a limiting case of a hyper-exponential distribution when one of the two phases has zero service time. GE distribution is useful for modeling random variables with SqV greater than 1. When modelling interarrival times, the GE distribution can model batch Poisson arrivals with gemotrically distributed batch sizes. Furthermore, in [5], Kouvatsos has shown that the GE distribution is a robust two moment approximation for any service time distribution by showing that the mean queue length distribution of GE/G/1 queue subject to utilization and mean queue length constraints is identical to exact equilibrium solution when service distribution is represented by an moment-matched GE distribution.

The two most useful properties of the GE distribution for use in a queueing network is that it is closed under merging and Bernoulli sampling. This means that if a GE distribution is used to describe the inter-arrival times of jobs into a queue and for their service times, it has properties similar to that of the exponential distribution. If two arrival processes with inter-arrival times drawn from a GE distribution are merged, the inter-arrival times of the merged process is also a GE distribution. Similarly, if a Bernoulli sampling of an arrival process with GE distribution is performed, then the distribution of the inter-arrival times of the sampled process is also a GE distribution. The following results for the merging and splitting of GE arrivals are known.

Consider N streams with λ_i and $C\!I_i$ as the rate and SqV of the inter-arrival time of the i^{th} stream. If these are merged, the rate and SqV of the combined stream, λ_M and $C\!I_M$ respectively, are given by

$$\lambda_M = \sum_{i=1}^{N} \lambda_i \qquad C\!I_M = \frac{\lambda_M}{\sum_{i=1}^{N} \lambda_i/(1 + C\!I_i)} - 1 \qquad (2)$$

Similarly, if a GE arrival process with rate λ and SqV $C\!I$ is sampled with probability p, the rate and SqV of the sampled process, λ_s and $C\!I_s$ respectively, will be

$$\lambda_s = p\lambda \qquad C\!I_s = 1 - p + pC\!I \qquad (3)$$

In addition to being closed under merging and Bernoulli sampling, the output from a GE/GE/m/N queue can also be approximated as a GE process [6, 5]. It has been shown that if λ and CA are the rate and SqV of the inter-arrival times, μ and CA_s that of the service times, then the output process has a rate and SqV, λ_{out} and CA_{out} respectively, given by

$$\lambda_{out} = \lambda(1 - PB)$$
$$CA_{out} = \rho^2 CA_s + (1-\rho)CA + \rho(1-\rho) \quad (4)$$

where $\rho = \lambda/m\mu$ and PB is the probability that an arriving call is blocked. Note that this is not the time average of there being N customers in the system because the arrivals are not Poisson and is given by

$$PB = \sum_{n=0}^{m-1}\left[\frac{\sigma}{\sigma(1-\tau)+\tau}\right]^{m-n}(1-\tau)^{N-n}\pi(n) + \sum_{n=m}^{N}(1-\tau)^{N-n}\pi(n) \quad (5)$$

where $\sigma = \frac{2}{1+CS}$, $\tau = \frac{2}{1+CA}$ and $\pi(n)$ is the probability that there are m jobs in the queue. $\pi(n)$ is obtained using the Maximum Entropy techniques outlined in [6]. In this paper, this last result is used to approximate the steady state solution of an open network of GE/GE/m/N queues by a product form solution similar to Jackson networks.

2.2. Model and Notation

We now describe the model, the notation and the assumptions in our analysis. We consider a network of M GE/GE/m/N queues. The set of these queues is denoted by $\mathcal{M}$. The external arrivals into queue i have a rate λ_i^e and SqV of the inter-arrival time is CA_i^e. The probability that a job finishing service at queue i goes to queue j is denoted by p_{ij}. Queue i will have m_i servers with μ_i as the rate of service and CS_i as the SqV of the service time distribution. The buffer capacity of queue i will be N_i. The steady state probability of there being n_i jobs in queue i will be denoted by $\pi_i(n_i)$ and $\Pi(\mathbf{n})$ is the steady state probability that the system state is $\mathbf{n} = [n_1, n_2, n_3, \cdots, n_M]$. From the results of [6] which approximate the output of a GE/GE/m/N queue by a GE process, we can provide a product form approximation for $\pi(\mathbf{n})$ and write $\pi(\mathbf{n}) = \prod_{i=1}^{M}\pi_i(n_i)$ and $\pi_i(n_i)$ can be obtained using Maximum Entropy techniques as described in [6]. Note that to use this technique, the mean and SqV of the inter-arrival times and the service times for each of the M queues need to be specified. The iterative analysis technique that we describe will obtain the effective arrival and service time descriptions at each of the M queues in the network.

Since we are considering queues with finite capacity, some jobs will experience blocking. External arrivals that see a full queue are assumed lost. Jobs moving from one queue to another seeing a full queue at the destination are blocked. Blocked jobs are held at the server of the queue where they just completed service, blocking the server so that it cannot serve any other job, of the node where they just completed service until such a time that the destination can accommodate them. These blocked jobs will block the server of the queue where they just completed service and the server cannot serve any other jobs. This is called transfer blocking. Let PB_j^i be the probability that a job finishing service at queue i and wanting to go to queue j is blocked, PBa_j the probability that a job wanting to arrive into queue j is blocked, PBf^i the probability that a job finishing service at queue i is blocked and PBe_i the probability that an external arrival to node i is blocked.

3. Approximate Analysis

In the approximate method that we describe here, we make use of the three properties of GE distributions described in Section 2.1 - (i) the departure process from a GE/GE/m/N queue can be approximated by a GE distribution, (ii) Bernoulli sampling of these departures (corresponding to Markovian routing in the network) continues to have a GE distribution and (iii) Combining these sampled processes, corresponding to the combined arrival process at a node in the network, still has a GE distribution. Thus the nodes could be modeled as GE/GE/m/N queues and a product form solution defined for the network. But the fact that there is blocking complicates the process of describing the arrival and service process at the nodes. Further, because the blocking discipline is transfer blocking, the queues are not work conserving and the product approximation cannot be used[12]. Thus we make an equivalent work conserving network by adding "holding nodes" to the original network. The holding nodes will be GE/GE/∞ queues and the parameters of the GE distributions corresponding to the service time and the inter-arrival times will be such as to correspond to the blocking delay experienced by a job finishing service and wanting to go to a finite buffer queue. Thus if we are able to describe the arrival and service processes to the queues in this equivalent network, we can solve for the system parameters. To obtain these descriptions we use an iterative procedure similar to the relaxation method of obtaining a fixed point solution. We stop the iterative process when the parameters of interest converge. To explain our analysis method, we will use the running example of Fig. 1

3.1. Equivalent Work Conserving Network

We construct the equivalent work conserving network by introducing a *holding node* between every node pair (i, j) for which $p_{ij} \neq 0$. The idea is to unblock a blocked server

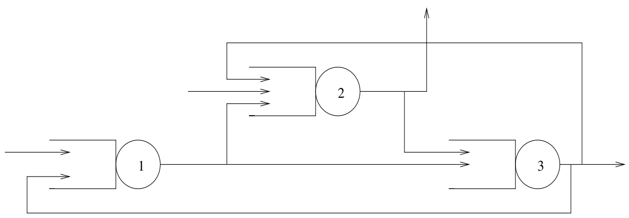

Figure 1. Network for the running example. External arrivals occur to nodes 1 and 2 and departures occur from nodes 2 and 3. Nodes 1 and 3 have a finite number of buffers.

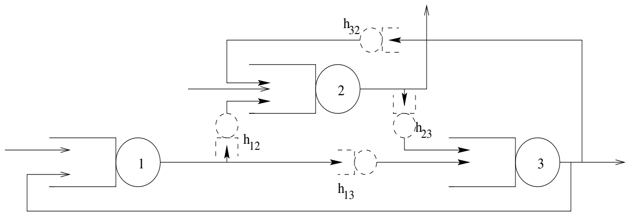

Figure 2. Holding nodes for jobs along (1,3), (1,2), (2,3) and (3,2) that could be blocked due to finite buffers at nodes 2 and 3. The holding nodes are drawn using broken lines.

by removing the job that is blocking it and holding it in the holding node. Thus there will be a holding node corresponding to every off-diagonal, non-zero entry of the routing matrix. The time spent by a blocked job in the holding node should correspond to the time that it would be blocked. Simultaneously, we should also increase the service time of other jobs at the queue that was unblocked to reflect the time lost to the jobs behind the blocked job. This is similar to the method used in [3]. The holding node that holds jobs blocked at node i that want to go to node j is denoted by h_{ij}. We will denote the set of holding nodes by $\mathcal{H}$ and the union of the sets $\mathcal{M}$, the set of all nodes in the network, and $\mathcal{H}$ by $\mathcal{MH}$. The holding nodes just represent the additional delay encountered due to blocking. This is shown in Fig. 2. The holding node is a GE/GE/∞ queue and is used primarily to introduce a delay equivalent to the blocking delay for the blocked jobs.

We now obtain the parameters of the arrival process and the service process at h_{ij}. The arrival process is approximated as a Bernoulli sampling, of the departure process from queue i with probability PB_j^i, the blocking probability for the jobs that are leaving i and going to j. To obtain the service time moments at the holding node $h_{ij}l$ we note the following. The time for node j to accept a new job is the minimum of m_j residual times corresponding to one of the jobs in service finishing service. The moments of the distribution of this time can be calculated from residual time results (see for example [4]). Further, note that there may be jobs at other nodes that want to go to j and are blocked. In the running example, if there are blocked jobs wanting to go to node 3 from both nodes 1 and 2, the holding nodes h_{23} and h_{13} are occupied. When a service completion occurs at node 3 only one of these can move to node 3 and the other one is blocked again. Thus "service completion" (corresponding to the destination finishing a service) at the holding node could result in one more round of blocking before the job becomes eligible to move into queue j. Thus there will be a non zero blocking probability, $PB_j^{h_{ij}}$, for jobs going from h_{ij} to j and these blocked jobs return to h_{ij}. We therefore need to further modify the moments of the service time at the holding node to reflect this immediate feedback and is done by increasing the service times at the holding node.

The mean and SqV of the inter arrival times of the jobs arriving at the holding node h_{ij} is obtained as follows. The job will arrive at the holding node if it selects node j (with probability p_{ij}) and is blocked (with probability PB_j^i). Thus the arrival process is a sampling of the output process of node i with a sampling probability $p_{ij}PB_j^i$. Thus, using Eqns 3 and 4, the arrival rate and the SqV of the inter-arrival times into the holding node h_{ij}, $\lambda_{h_{ij}}$ and $CA_{h_{ij}}$ respectively, are given by

$$\begin{aligned}\lambda_{h_{ij}} &= (PB_j^i p_{ij})\lambda_i \\ CA_{h_{ij}} &= 1 - (PB_j^i p_{ij}) + (PB_j^i p_{ij}) \\ &\qquad (\rho_i^2 CS_i + (1-\rho_i)CA_i + \rho_i(1-\rho_i))\end{aligned} \quad (6)$$

where $\rho_i = \lambda_i/(m_i\mu_i)$.

The mean and SqV of the service times are determined as follows. When blocking occurs, all the servers of the blocking node are busy. The distribution of the time until any one of the m_i servers finishes service is the residual life of the minimum of m_i GE random variables of rate μ_i and SqV CS_i. Let μr_i and CSr_i be the mean and SqV of this residual life and these can be obtained using well known techniques. (See for example [4]. The mean and SqV for the service time can be approximated by the geometric sum of random variables with mean μr_j and CSr_j. From [3], the mean and SqV of the service time distribution at the holding node h_{ij} will be

$$\begin{aligned}\mu_{h_{ij}} &= \mu r_j(1 - PB_j^{h_{ij}}) \\ CS_{h_{ij}} &= PB_j^{h_{ij}} + (1 - PB_j^{h_{ij}})CSr_j\end{aligned} \quad (7)$$

Obviously, we need to use the following modified routing probabilities.

$$\begin{aligned}p_{ih_{ij}} &= PB_j^i \\ p_{ij} &= p_{ij} - PB_j^i\end{aligned}$$

$$p_{h_{ij}k} = \begin{cases} 1 & \text{for } k = j \\ 0 & \text{for } k \neq j, k \in \mathcal{MH} \end{cases} \quad (8)$$

The mean number of jobs in the original node is actually equal to the sum of the jobs in the original node and the expected number of jobs in the holding nodes of all the streams emerging from this original node. Also, the actual service received at a node is the sum of the service time at the node and the service time in the holding node.

Having obtained the parameters to describe the queueing system at the holding nodes, we are now ready for an iterative procedure along the lines of a fixed point solution to obtain the approximate steady state solution for the queueing network.

3.2. Iterative Procedure

The iterative procedure that we use is fairly straightforward. We first obtain the arrival process and service times at each node of the expanded network. We then treat each queue independently and use the Maximum Entropy method of [6] to find $\pi_i(n_i)$, the probability of there being n_i jobs in node i and also obtain the blocking probabilities PB^i_j, PBf^i and PBa_j defined earlier. It is easy to see that the blocking probabilities and the arrival and service process moments are coupled and we will have to solve a set of non linear equations. At the end of the recursion, we know the first two moments of the arrival and service processes at the "holding node". Using these results the statistics of the original network are modified to take into account the blocking in the network and then product form solutions are used to find overall network parameters.

It is shown in [12] that $p_{ii} = 0$ for arrival process at queue i to be a renewal process. We achieve this by assuming that a immediate feedback corresponds to a service time equal to two consecutive service times and then, following [3], modify the service time moments of the nodes with $p_{ii} \neq 0$ as follows

$$\begin{aligned} \mu_i &= \mu_i(1 - p_{ii}) \\ CS_i &= p_{ii} + CS_i(1 - p_{ii}) \end{aligned} \quad (9)$$

We also need to normalize the transition probabilities, p_{ij}, by conditioning on the transition not being to itself

$$p_{ij} = \begin{cases} \frac{p_{ij}}{1-p_{ii}} & \text{for } i \neq j, j \in \mathcal{M} \\ 0 & \text{for } i = j \end{cases} \quad (10)$$

The mean and SqV for nodes $i \in \mathcal{M}$ are obtained as follows. The arrival rate into node j, λ_j, should satisfy the following flow balance equation

$$\lambda_j(1 - PBa_j) = \lambda^e_j(1 - PBe_j) + \sum_{\imath \in \mathcal{MH}} \lambda_i(1 - PBa_i)p_{ij} \quad (11)$$

The LHS is the actual rate of arrivals into node j after accounting for those that are blocked and the RHS is the sum of the successful external arrivals and the arrivals from other nodes that want to come to node j. This is similar to the flow balance in Jackson networks. The SqV of the inter-arrival time at each node j is obtained as follows. The departures from every node $i \in \mathcal{MH}$ are sampled with probability p_{ij} and Eqns 4 and 3 are used to get the individual streams arriving at node j. The SqV of the combined arrival process into node j is then obtained using Eqn 2.

To obtain the mean and SqV of the inter-arrival times to node j, we need to know the following blocking probabilities - blocking probabilities of jobs finishing service at node i and wanting to go to node j (PB^i_j), blocking probability for arrivals to node j (PBa_j), and blocking probability for external arrivals to node j (PBe_j). These are obtained from [6] and are as follows.

$$PBe_j = \sum_{n=0}^{m_j-1} \left[\frac{CA^i_j + 1}{CA^i_j + CS_j}\right]^{m_j-n} \left[\frac{CA^i_j - 1}{CA^i_j + 1}\right]^{N_j-n} \pi_j(n) + \sum_{n=m_j}^{N_j} \left[\frac{CA^i_j - 1}{CA^i_j + 1}\right]^{N_j-n} \pi_j(n) \quad (12)$$

where CA^i_j is the SqV of the inter-arrival times of the jobs from node i to node j obtained using Eqn. 4 with appropriate substitutions. PBa_j is given by

$$PBa_j = \frac{\lambda^e_j PBe_j + \sum_{i \in \mathcal{M}} \lambda_i p_{ij} PB^i_j/(1 - PB^i_j)}{\lambda^e_j + \sum_{i \in \mathcal{M}} \lambda_i p_{ij}/(1 - PB^i_j)} \quad (13)$$

Note that $PBe_j \neq PBa_j$ because arrivals from inside the network into node j are not Poisson. PBe_j is given by [6]

$$PB^i_j = \sum_{n=0}^{m_j-1} \left[\frac{CA^e_j + 1}{CA^e_j + CS_j}\right]^{m_j-n} \left[\frac{CA^e_j - 1}{CA^e_j + 1}\right]^{N_j-n} \pi_j(n) + \sum_{n=m_j}^{N_j} \left[\frac{CA^e_j - 1}{CA^e_j + 1}\right]^{N_j-n} \pi_j(n) \quad (14)$$

As was explained earlier, we need to increase the service time moments of the nodes in $\mathcal{M}$ to account for the fact that a blocked job has been moved to a holding node. The increase should be such as to have the same effect as that when the blocked job is blocking the server. Thus the mean and the SqV of the service times at the nodes $i \in \mathcal{M}$ are as follows. The derivation is similar to Eqns 7 or 9

$$\text{For } i \in \mathcal{M} \qquad \begin{aligned} \mu_i &= (1 - PBf^i)\mu_i \\ CS_i &= PBf^i + CS_i(1 - PBf^i) \end{aligned} \quad (15)$$

with

$$PBf^j = \sum_{i \in \mathcal{MH}} p_{ij} PB^i_j \quad (16)$$

The iterative algorithm is summarized as follows

1. Make $p_{ii} = 0$ by using modifications in Eqns 9-10.
2. Let $\mathit{PB}^i_j = \mathit{PBe}_j = \mathit{PBa}_j = \mathit{PBf}^i = 0 \; \forall \, i, j \in \mathcal{M}$.
3. Expand network by adding holding nodes
4. Obtain mean of interarrival times using Eqn. 11 and its SqV as explained therein. Also obtain mean and SqV of service times of using Eqns. 15-16.
5. Using ME method of [6], solve for $\pi_i(n_i), i \in \mathcal{MH}$
6. Find blocking probabilities PB^i_j, PBa_j and PBe_j $\forall$ $i, j \in \mathcal{MH}$
7. Do as in step 4
8. If flow statistics (mean and SqV of the interarrival times at nodes) have not converged, go to step 5

At the end of the iteration, the arrival and service processes at the nodes in the expanded network are available and also the distribution of the jobs in each of the nodes. Most of the performance parameters of interest can now be found using standard theory. However, it must be noted that because of the use of holding nodes, the performance parameters for the real network should be appropriately corrected. For example, consider the average number of jobs in a queue. In the expanded network model, the average number of jobs in node i is given by $\sum_{n_i=1}^{N_i} n_i \pi_i(n_i)$ for all i. Recall that the jobs in the holding nodes $h_{ij}, j \in \mathcal{M}$ also correspond to those that are waiting at node j. Thus we apply a correction term to obtain the actual mean of the number of jobs at node j, $\overline{K_j}$, as follows

$$\overline{K_i} = \sum_{n_i=1}^{N_i} n_i \pi_i(n_i) + \frac{\sum_{\mathrm{J} \in \mathcal{MH}} p_{ij} \lambda_{h_{ij}}}{\mu_{h_{ij}}} \qquad (17)$$

4. Numerical Results

In this section we discuss the performance of our approximation analysis method. We have not investigated the proofs for the convergence of the iteration procedure or its performance in terms of the rate of convergence. Since much of the algorithm uses the ME technique that is discussed in detail in [6], we argue that this technique will work as well. In the rest of this section we will consider only the goodness of the approximation technique by comparing the analysis results with those from a simulation model. We will consider four examples in this section. In three of the examples we consider small three node networks with one feature like feedforward, feedback and splitting. In the fourth example a larger system with eight nodes is considered. We evaluate the utilization ρ, the mean buffer occupancy $\overline{K}$ and the mean throughput from the node, λ.

4.1. A Feedforward Network

The network configuration and routing probabilities are shown in Fig. 3. External arrivals occur only at node 1 and jobs depart from the network from all the three nodes. The routing probabilities are $p_{12} = 0.4$, $p_{13} = 0.4$ and $p_{23} = 0.5$. The departure probabilities are $p_{1e} = 0.2$, $p_{2e} = 0.5$, $p_{3e} = 1.0$. The other input parameters are

i	λ_i^e	CA_i^e	μ_i	CS_i	m_i	N_i
1	1.5	2.0	2.0	1.0	1	5
2	0.0	0.0	2.0	1.0	3	4
3	0.0	0.0	2.0	1.0	1	3

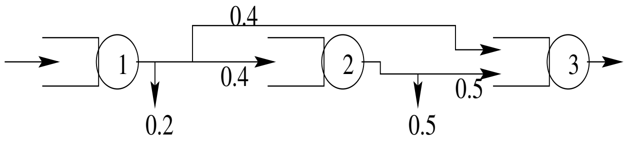

Figure 3. The feedforward network

The performance measures according to the analysis and simulation model are shown in Table 1. Note that the highest relative error between the analytical and simulation models is about 10.6%. We believe this to be tolerable, especially because of the high load.

Parameter	Analysis	Simulation	Rel Error
ρ_1	0.750	0.678	0.106
ρ_2	0.086	0.092	0.068
ρ_3	0.406	0.399	0.017
$\overline{K_1}$	1.702	1.802	0.055
$\overline{K_2}$	0.261	0.276	0.056
$\overline{K_3}$	0.578	0.621	0.069
$\overline{\lambda_1}$	1.284	1.346	0.046
$\overline{\lambda_2}$	0.514	0.538	0.046
$\overline{\lambda_3}$	0.771	0.807	0.045

Table 1. Performance measures for the feedforward network example

4.2. Network with Splitting

In this network, the flow is split out of the first node as shown in Fig. 4. Departures occur from all nodes and external arrivals also occur into all nodes. The routing probabilities are $p_{12} = 0.3$ and $p_{13} = 0.6$. The exit probabilities are $p_{1e} = 0.1$ and $p_{2e} = p_{3e} = 1.0$. The other input parameters are

i	λ_i^e	CA_i^e	μ_i	CS_i	m_i	N_i
1	2.0	1.0	1.0	2.0	2	5
2	0.2	1.0	2.0	0.5	1	4
3	0.2	1.0	2.0	0.5	1	4

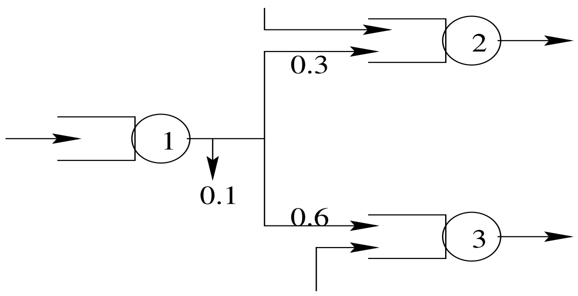

Figure 4. The Network with splitting

The results from the analysis and simulation models are shown in Table 2. Once again, the relative errors are very small and the worst error is about 6.6%.

Parameter	Analysis	Simulation	Rel Error
ρ_1	0.807	0.808	0.002
ρ_2	0.336	0.344	0.012
ρ_3	0.569	0.574	0.009
$\overline{K_1}$	2.548	2.727	0.066
$\overline{K_2}$	0.482	0.466	0.035
$\overline{K_3}$	1.073	1.054	0.018
$\overline{\lambda_1}$	1.579	1.613	0.021
$\overline{\lambda_2}$	1.579	1.613	0.021
$\overline{\lambda_3}$	1.138	1.160	0.019

Table 2. Performance measures for the network with splitting

4.3. Network with Feedback

In this example we consider a feedback from node 3 back to node 1 with probability 0.4. The routing probabilities are $p_{12} = p_{13} = 0.35$ and $p_{31} = 0.4$. The exit probabilities are $p_{1e} = 0.3$, $p_{2e} = 1.0$ and $p_{3e} = 0.6$. The other input parameters that are used are

The performance measures from analysis and simulation are tabulated in Table 3. The worst relative error that we observe is about 4.6%.

i	λ_i^e	CA_i^e	μ_i	CS_i	m_i	N_i
1	1.5	0.5	1.0	2.0	3	6
2	0.0	0.0	1.0	1.0	2	6
3	0.0	0.0	1.0	1.0	2	6

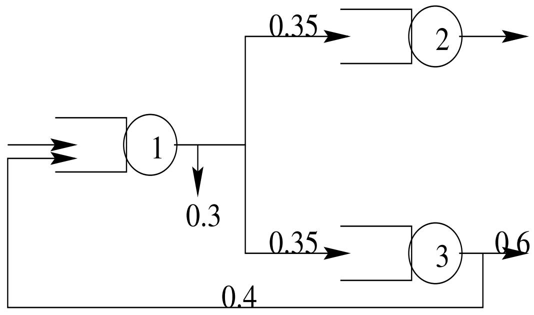

Figure 5. The Network with feedback

4.4. Manufacturing System Example

We now consider a larger example of a manufacturing system. A serial manufacturing system with eight machines $M1$–$M8$ is considered. This system is modeled as finite buffer tandem queue. The job flow is split in the process which to account for possibly different sizes or to provide different quality with the same initial processing. New jobs can enter the system at $M1$ and the completed jobs depart the system from $M6$, $M7$ or $M8$. All the queues in this network are single server queues. The routing probabilities from $M2$ to $M3$ is 0.5, to $M4$ is 0.3 and to $M5$ is 0.2. The arrival rate into $M1$ is $\lambda 1_e = 0.9$ and the SqV is $CA_2^e = 1.0$. The service rates and SqV at various machine are

$\mu_1 = 1.0$ $\quad \mu_2 = 1.0$ $\quad \mu_3 = 2.0$ $\quad \mu_4 = 3.0$
$\mu_5 = 5.0$ $\quad \mu_6 = 2.0$ $\quad \mu_7 = 3.0$ $\quad \mu_8 = 5.0$
$CS_1 = 1.0$ $\quad CS_2 = 1.0$ $\quad CS_3 = 2.0$ $\quad CS_4 = 3.0$
$CS_5 = 5.0$ $\quad CS_6 = 2.0$ $\quad CS_7 = 3.0$ $\quad CS_8 = 5.0$
$N_1 = 9$ $\quad N_2 = 9$ $\quad N_3 = 9$ $\quad N_4 = 9$
$N_5 = 9$ $\quad N_6 = 9$ $\quad N_7 = 9$ $\quad N_8 = 9$

The network is shown in Fig. 6

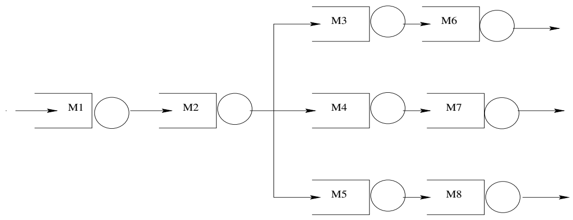

Figure 6. Network for the manufacturing system

The performance measures according to the analysis and simulation model are shown in Table 4. Since the load is

Parameter	Analysis	Simulation	Rel Error
ρ_1	0.567	0.567	0.000
ρ_2	0.298	0.295	0.008
ρ_3	0.298	0.299	0.003
$\overline{K_1}$	1.844	1.933	0.046
$\overline{K_2}$	0.658	0.647	0.031
$\overline{K_3}$	0.659	0.647	0.018
$\overline{\lambda_1}$	1.703	1.722	0.011
$\overline{\lambda_2}$	0.596	0.602	0.010
$\overline{\lambda_3}$	0.596	0.602	0.010

Table 3. Performance measures for the network with feedback

high, the error has increased but it is still tolerable at about 8.7% as the highest error.

Parameter	Analysis	Simulation	Rel Error
ρ_1	0.8465	0.8577	0.0131
ρ_2	0.8465	0.8194	0.0271
ρ_3	0.2116	0.2318	0.0871
ρ_4	0.0846	0.0872	0.0298
ρ_5	0.0338	0.0346	0.0231
ρ_6	0.2116	0.2131	0.0070
ρ_7	0.0846	0.0846	0.0000
ρ_8	0.0338	0.0336	0.0059
$\overline{K_1}$	3.7037	3.6632	0.0111
$\overline{K_2}$	3.6605	3.3594	0.0896
$\overline{K_3}$	0.2859	0.2945	0.0292
$\overline{K_4}$	0.0993	0.1010	0.0168
$\overline{K_5}$	0.0372	0.0381	0.0282
$\overline{K_6}$	0.2671	0.2747	0.0277
$\overline{K_7}$	0.0974	0.0982	0.0081
$\overline{K_8}$	0.0370	0.0368	0.0054

Table 4. Performance measures for manufacturing system

5. Discussion and Conclusion

In this paper we described an approximation technique to analyze a network of GE/GE/m/N with transfer blocking. The method involves approximating a product form solution and obtaining the marginal distributions for the state of the individual queues by the MEM technique. "Holding Nodes" are added to take into account transfer blocking encountered due to finite buffer space at the queues. The queue state distributions and the correction term for "holding nodes" are used to obtain the performance parameters for individual queues and the overall network.

We have implemented this algorithm as part of QNAT[8], a powerful GUI based Mathematica library for the Microsoft Windows environment that can be used to analyze a large class of queueing networks. QNAT is designed to analyze arbitrary configurations of multi-server queues with infinite or finite capacity, fork-join queues with or without synchronization queues that are specified through a GUI. An important feature of QNAT is that it can analytically analyze open or closed blocking networks that employ transfer, repetitive service or rejection blocking. We developed this technique to make QNAT capable of analyzing all kinds of open and closed blocking networks that serve a single class of customers.

References

[1] I. F. Aykildiz. Product form approximations for queueing networks with multiple servers and blocking. *IEEE Trans on Computers*, 38(1):99–113, Jan 1989.

[2] K. P. Jun and H. G. Perros. *Approximate Analysis of Arbitrary Configurations of Queueing Networks with Blocking and Deadlock*. Elsevier Science Publishers B.V. (North Holland), Queueing Networks with Blocking H. G. Perros and T. Atilok (Editors), 1989.

[3] L. Kerbache and J. M. Smith. The generalised expansion method for open finite queueing networks. *European Journal of Oper Res*, 38:448–461, 1990.

[4] L. Kleinrock. *Queueing Systems, Vol I: Theory*. John Wiley, 1975.

[5] D. D. Kouvatsos. MEM for arbitrary queueing networks with multiple general serv ers and repetitive-service blocking. *Performance Evaluation*, 10:169–195, 1989.

[6] D. D. Kouvatsos and N. P. Xenios. MEM for arbitrary queueing networks with multiple general servers and repetitive-service blocking. *Performance Evaluation*, 39(2):183–200, 1988.

[7] H.-S. Lee and S. M. Pollock. Approximation analysis of open acyclic queueing networks with blocking. *Operations Research*, 38:1123–1134, 1990.

[8] D. Manjunath, H. Tahilramani, and S. K. Bose. Queueing network analysis tool (qnat): A graphical tool for the analysis of queueing networks. *Proceedings of IEEE TENCON'98 International Conference, New Delhi*, Dec 1998.

[9] R. Onvural. Survey of closed queueing networks with blocking. *ACM Computing Surveys*, 22(2):83–121, Jun 1980.

[10] H. G. Perros. *Queueing Networks with Blocking - Exact and Approximate Solutions*. Oxford University Press, 1994.

[11] A. Schoemig. On transfer blocking and minimal blocking in serial manufacturing systems - the impact of buffer allocation. *http://www.informatik.uni-wuerzburg.de/reports/tr.html*, (TR121), July 1995.

[12] W. Whitt. The queueing network analyser. *Bell Systems Technical Journal*, 62(9):2279–2815, 1983.