Turkish J. Eng. Env. Sci.
30 (2006) , 269 – 283.

Performance Evaluation of Closed Queueing Networks with Limited Capacities

Mustafa YÜZÜKIRMIZI
Kırıkkale University, Department of Industrial Engineering, Kırıkkale-TURKEY
e-mail: myuzukirmizi@kku.edu.tr

Received 02.09.2005

Abstract

An algorithm was developed for performance evaluation of single class closed queueing networks with configurations likely to occur in real-world manufacturing systems, i.e. limited waiting spaces, split-merge topologies, and stations with multiple servers. The approach was tested by several numerical experiments to evaluate its robustness under different conditions. The algorithm proved to be accurate, efficient and very consistent with balanced and unbalanced service rates, varying number of customers in the system, and the system size.

Key words: Closed queueing networks, Blocking after service, Multiple servers.

Introduction

Motivation and purpose

Queueing networks are said to be finite if some or all of its nodes have limited capacities for entities stationed at them. Many complex service, manufacturing, and computer systems can be modeled using queueing networks and most of these systems have finite buffers. For example, in manufacturing systems, there is usually limited waiting room between workstations in assembly lines, material-handling systems, and cellular manufacturing cells. In telecommunication systems, there are finite capacity telephone lines and capacitated ATM switches.

Studies of finite closed queueing networks date back 4 decades to the pioneering work by Gordon and Newell (1967b). However, a vast majority of the studies concentrate on tandem queueing systems and single servers. Less research can be found on more general systems such as with multiple servers and arbitrary topologies. These types of closed networks have great applicability to real systems and it would be desirable to further develop appropriate analytical methods. As mentioned before, exact solutions are limited to specific cases for finite closed queueing networks; therefore we should consider approximation methods.

The approximation technique proposed in this study uses insights from the Expansion Method (EM) for modeling series, merging and splitting topologies. The EM's algorithmic approach has been used successfully to evaluate performance measures of finite queueing networks. Kerbache and Smith (1988) introduced this method to solve a wide range of finite open queueing networks, with exponential service inter-arrival times with varying traffic intensities. EM was further extended to general open queueing networks (Kerbache and Smith, 1987), i.e. general service times and non-exponential i.i.d. arrival rates. Soon, this approach was adopted for open networks with M/M/c/K state-dependent service and applied to arbitrarily configured series-parallel network topologies (Jain and Smith, 1994).

Moreover, Gonzales (1997) successfully adopted the EM to handle closed queueing networks with finite buffers in conjunction with an Equalization Phase and well-known Mean Value Analysis (MVA). The EM is based on a combination of repeated trials and node-by-node decompositions and characterized

by the addition of artificial nodes associated with each finite queue to reroute blocked customers to the blocking queues. These additional nodes effectively expand the network and enable the transformation of such systems into equivalent Jackson networks in which each node can be treated independently.

One of the main goals of this study was to utilize EM in the analysis of multiple server topologies in closed queueing networks.

Outline of paper

In the following part, we furnish a brief review of the literature. In model description section, we introduce the notation used for network descriptions, ideas used to develop the approximation, and present the algorithm. In the assessment of the method section, the algorithm is assessed numerically against the values obtained from simulation as well as existing algorithms where applicable. Lastly, the conclusions section is presented.

Literature Review

Although queueing theory has its foundations in telecommunication systems with the work by Erlang, it has also been applied to several other areas, such as manufacturing and computer system modeling. The modeling of these systems has been actively addressed by a large number of researchers over the last decades. There are several books published in this area, Bolch et al. (1998), Papadopoulos et al. (1993), Cooper (1981), and Nain (1998), to name a few. In the presentation of the material, we place emphasis on closed queueing networks with limited buffers, production blocking, and first-come-first-served queueing discipline.

Product form

Product form queueing networks have a simple closed form expression for the stationary state distribution that permits efficient algorithms to evaluate average performance measures. The state probability of a closed single class queueing has a product form solution as follows:

$$\pi(n_1, n_2, ..., n_M) = \frac{1}{G(N)} \prod_{i=1}^{M} g_i(n_i) \qquad (1)$$

where $G(N)$ is the normalization constant, N is the network population, and $g_i(n_i)$ is a function of state n_i and depends on the type of service center *i, (i = 1, . . . ,M).*

Jackson (1963) introduced product form queueing network models for open exponential networks, while Gordon and Newell (1967a) did the same for closed exponential servers. The queueing disciplines at all stations were assumed to be FCFS. These results were extended to open, closed, and mixed networks with several customer classes, non-exponentially distributed service times, and different queueing disciplines by Baskett et al. (1975), and known as the BCMP theorem.

The computation of the normalization constant, $G(N)$, was facilitated by the convolution algorithm of Buzen (1973). Since the calculation of the normalization constant is very intensive, a new algorithm, MVA, was developed by Reiser and Lavenberg (1980), which avoids its explicit calculation for closed queueing networks with infinite buffer capacities and allows for multiple customer classes and chains. This approach is based on the following 2 fundamental theorems and computes the mean values of interest such as mean waiting time, mean number of customers at each node, throughput, and utilization.

Another algorithm for analyzing product-form queueing networks is the flow-equivalent-server (FES) method (Chandy et al., 1975). This method applies Norton's theorem from electric circuit theory to closed queueing networks. The queueing network is decomposed into a simpler network of 2 stations, where one represents a single station from the original network and the other represents the complement network. The related theorem states that the queue length distributions at the isolated station in the original network and the 2-station network are equal.

The convolution algorithm, MVA, and FES algorithm have the same running time complexity. However, due to the algorithmic appeal and intuitiveness of MVA, many approximations have been suggested to deal with shortcomings and extensions to non-product-form cases. Even before the publication of the MVA, approximate MVA algorithms were presented by Bard (1979) and Schmidt (1997). In a comparative study, Wang (1997) surveyed existing approximate MVA algorithms for product-form networks and examined the relative merits and trade-offs of different implementations and analyzed the complexity of the approximate MVA algorithms.

A closed queueing network with finite buffers cannot be shown to have a product form solution other

than for a few special cases. Some classes of product-form closed networks with finite capacities can be found in the survey conducted by Balsamo (1993). In his review, finite queueing networks under different blocking mechanisms, and their exact and approximate analytical solution methods were presented. In addition, the properties of product-form finite queueing networks that arise in comparison of different models are discussed.

A closed queueing network with finite buffers under production blocking is shown to have a product-form solution when:

- The network has exactly 2 nodes (Akyildiz, 1987), and
- The number of customers in the network is equal to the capacity of the smallest buffer plus one (Onvural and Perros, 1986).

Another comprehensive survey of closed queueing network compiled by Onvural (1989) noted that, other than these special cases, closed queueing networks under production blocking do not have a product-form.

Non-product form

Although many algorithms are available for solving product form queueing networks, most practical queueing problems lead to non-product form networks. Theoretically, all closed queueing networks with exponential service times can be solved numerically. This is done by identifying the Markov chain underlying the model and solving the system of equations to determine the steady-state probability. For a homogeneous, irreducible, continuous time Markov chain with n states, the set of equations is:

$$\pi\Theta = 0$$

where Θ is an nxn whose elements θ_{ij}denote the rate of transition of the chain from state i to state j and π, $\pi = (\pi_1,..., \pi_n)$ is the stationary probability vector in which π_i is the stationary probability of the Markov chain being in state i.

However, the state space and the number of numerical evaluations grow exponentially with the network size. Instead of the costly alternative of a discrete-event simulation, approximate solutions may be considered.

A comprehensive survey of the literature on closed queueing networks with blocking was compiled by Onvural (1990). Akyildiz (1988) presented an approximation method for the throughput of a finite closed queueing network. His concept is based on that of a non-blocking queueing network with an appropriate total number of customers that could be derived such that the state space is equal to the state space of the blocking network. His transformation of state spaces is exact for 2 station networks and approximate for multiple station cases.

Onvural and Perros (1989) developed an approximation algorithm to calculate the throughput of large closed exponential queueing networks with finite buffers. The algorithm determines approximately the number of customers such that the throughput of the network is at maximum and fits a curve that passes through a number of known points to estimate the unknown throughput values as the number of customer in the network varies.

A decomposition procedure was developed by Suri and Diehl (1986) for cyclic networks in which the first node has infinite capacity. The algorithm is based on the idea of replacing all the downstream nodes of the i^{th} node by a single flow-equivalent finite capacity node with varying buffer size.

Another algorithm for cyclic queues where the first node has infinite capacity was proposed by Dallery and Frein (1986). They decomposed the network into m individual queues and analyzed each of them as a $M/M/1/K$ queue with a state dependent service mechanism.

An approximate MVA of queueing networks with blocking after service was developed by Akyildiz (1988). His approximation is based on the modification of the time spent in the queue and in service by a customer due to the blocking events that occur in the network.

Another approximate MVA algorithm for solving an exponential cyclic network with blocking was developed by Zhuang et al. (1994). In their algorithm, they modify the arrival instant theorem to account for finite queue capacity and use a set of equations involving the average values of performance measures.

Frein and Dallery (1993) developed an analytical method modeling a production line with finite buffers as closed queueing networks with blocking. Their principle concerns decomposing the network into a set of 2-server subsystems. The population constraint of the network is taken into account by summing the average queue lengths of the different subsystems. They also derived properties pertaining to their method.

An approximation for a cyclic queue with general

service times was developed by Liu et al. (1992). They estimate the throughput of the network up to the maximum value combining Norton's theorem with a decomposition technique.

Expanded Mean Value Analysis (EMVA)

Although MVA was derived to provide an exact analysis of product form networks, its appealing algorithmic features, which are that it is numerically stable and capable of providing a reasonable physical interpretation and works directly with the desired statistics, have aroused intense research interest in developing appropriate variants for approximate analysis of non-product form closed queueing networks.

In the proposed EMVA approximation, the blocking effect of the closed queue is taken into account through an approximation of the mean effective service time of each server. The mean effective service time is embedded in the state dependent MVA algorithm. Further, marginal queue length probabilities are used to estimate the blocking probabilities along with these effective service rates.

Model description

Consider a closed queueing network, $G(M, A)$, with a finite set of M nodes and finite set of A arcs connecting nodes. For convenience, the following symbols are used in the description of queueing networks:

M	number of nodes.
N	number of jobs/customers in the system.
n_i	number of jobs/customers at the i^{th} node ($\sum_{i=1}^{M} n_i = N$).
$(n_1, n_2,..., n_M)$	the state of the network.
$\pi_i(n_1, n_2,..., n_M)$	probability that i^{th} node contains n_i jobs *(i = 1, ... ,M).*
μ_i	service rate at i^{th} node (i = *1,...,M*).
$1/\mu_i$	mean job service time of the jobs at i^{th} node (i = *1,...,M*).
V_i	visit ratio of i^{th} node (i = *1,...,M*).
r_{ij}	routing probability, probability that a job is transferred to the j^{th} node after service completion at the i^{th} node *($i \neq j$, i = 1,...,M, j = 1,...,M).*
$\pi_i(j\|n)$	conditional probability of having j jobs at the i^{th} node given that there are n jobs in the system (i = *1,...,M,* j = *0,..., n,* n = *0,...,N*).
K_i	buffer capacity of i^{th} node including the servers (i = *1,...,M).*
c_i	the number of parallel servers at the i^{th} node ($c_i \geq 1$, i = *1,...,M*).
b_i	waiting space for the i^{th} node *(i = 1,...,M).*
$P_i(b)$	probability of a customer being blocked at node i.
P_{Ki}	probability that a node i, with finite buffer size of K_i blocks a customer.

The performance measures are:

λ	overall throughput rate.
λ_i	throughput rate at i^{th} node *(i = 1,...,M).*
ρ_i	utilization of i^{th} node *(i = 1,...,M).*
W	cycle time, i.e. average time for a customer to complete one loop or cycle.
W_i	average time spent at i^{th} node *(i = 1,...,M).*
Q_i	mean number of customers at i^{th} node *(i = 1,...,M).*

The service discipline at all stations are first-come-first-served (FCFS) and service time distributions are exponential. Each station i may have a multiple number, c_i *(c_i = 1, 2, . . .),* of servers. Therefore, the station i can serve a maximum number of c_i customers, simultaneously (Figure 1). The service time at station i is$\frac{1}{\mu_i}$, and the service capacity, the rate at which a station services customers if no blocking occurs, is $c_i\mu_i$.

We assume there are b_i waiting spaces for the i^{th} node and customers are blocked after service. This type of blocking arises when the destination node j is full, at the moment the customer at node i attempts to enter. The customer at node i is blocked and resides in the server, in consequence, preventing the server from beginning service on another customer. We note that the servers provide spaces for the customers, as well, which leads the number of total buffer spaces to $K_i = b_i + c_i$ at node i effectively.

Effective service rates

The method used for estimating the effective service rates μ_i of node i, is based on the assumptions made in the EM approximation. In the EM, an artificial node h is added for each finite node in the network, effectively expanding the original network (Figure 2).

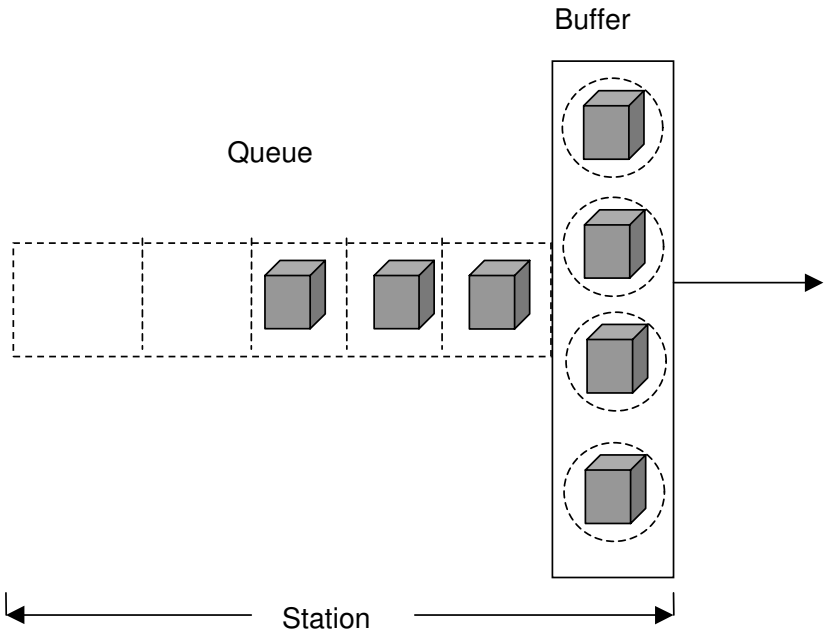


Figure 1. A single station with multiple identical parallel servers.

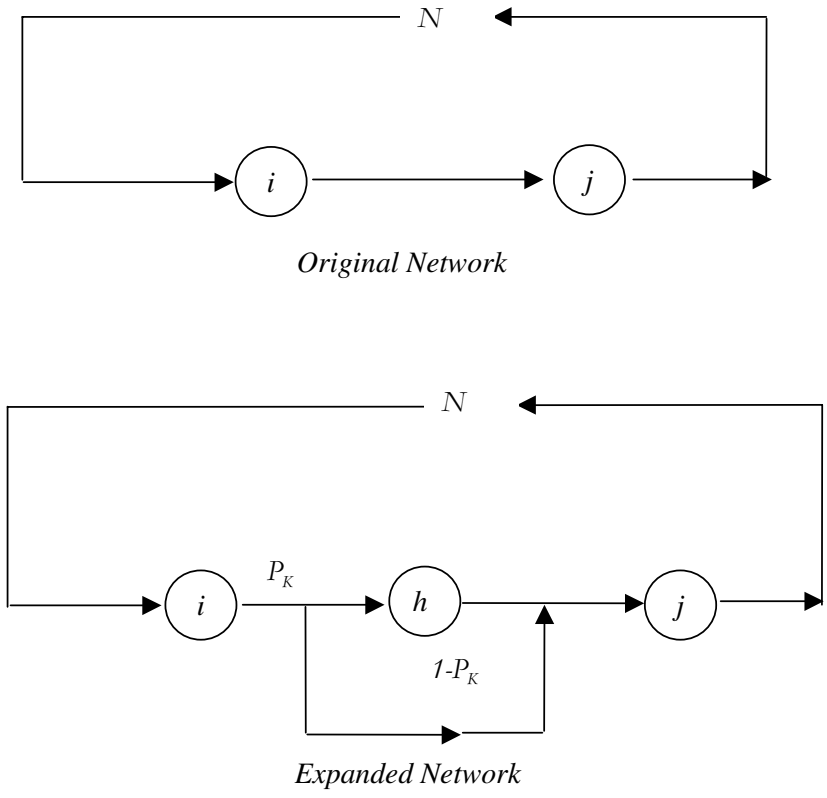


Figure 2. Expansion of network.

The name EM alludes to this first stage, which creates a holding node for the overflow of customers due to blocking. After being serviced at node i, a customer attempts to proceed to downstream node j. If node j has space to accommodate the customer, the customer successfully joins queue j with a probability of $(1 - P_K)$. However, if node j is saturated, the customer is blocked at the previous node i. An additional delay, caused to customers trying to join the queue at j when it is full, with probability P_K, is incurred at the artificial node h. The artificial node is modeled as an $M/M/$: queue. The infinite number of servers is used simply to serve the blocked customer a delay without queueing.

In this method, the mean service times at node i preceding the finite node are μ_i^{-1} and $(\mu_i^{-1} + \mu_h^{-1})$ when in the unsaturated and saturated phases, respectively. Thus, on average the mean service rate at the node i preceding a finite node is

$$\mu_i^{-1} = \mu_i^{-1} + P_{Kh}\mu_h^{-1} \tag{2}$$

For tandem networks, nodes in series, the service rate of the holding node will be the service rate of the succeeding node, which can be denoted as follows:

$$(\mu_h)_i = \mu_i + 1 \text{ for } i = 1,\ldots,M - 1$$
$$(\mu_h)_i = \mu_1 \text{ for } i = M.$$

In another node, if the destination node has multiple servers, $(c_i + 1 > 1)$, then the holding service rate would be:

$$(\mu_h)_i = c_{i+1}\mu_{i+1}.$$

This comes from the fact that, when a station causes blocking, it is full, and therefore is working at the service capacity.

Blocking probability

In this study, we assume production blocking, sometimes referred to as Blocking After Service (BAS). In the BAS mechanism, a customer upon completion of its service at node i attempts to enter destination node j. If node j at that moment is full, the customer is forced to occupy server i until the destination node becomes available, and node i is blocked. Server i cannot serve any other customer that might be waiting in its queue.

In this type of blocking, the blocked server i effectively provides an additional queueing position for customers awaiting service at j. The effective queue capacity of blocking node j increases by 1. At this instant, the node j has full capacity of customers and additionally blocks node i.

The state change of such a finite node is illustrated in Figure 3. When there are n $(n \geq K)$ customers in the system, a finite node may have at most K customers. The last state depicts the blocking state where a blocked customer awaits in the upstream node.

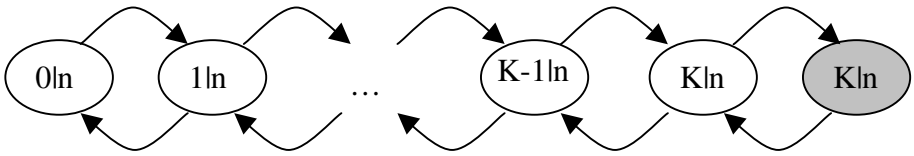


Figure 3. The states for a finite capacity node.

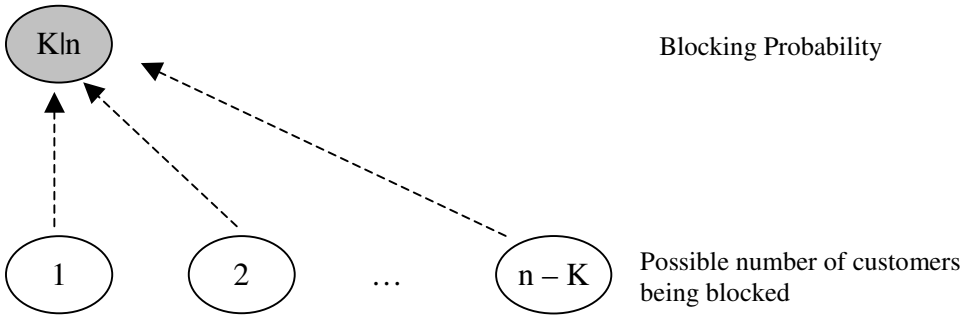


Figure 4. Partitioning the blocking probability.

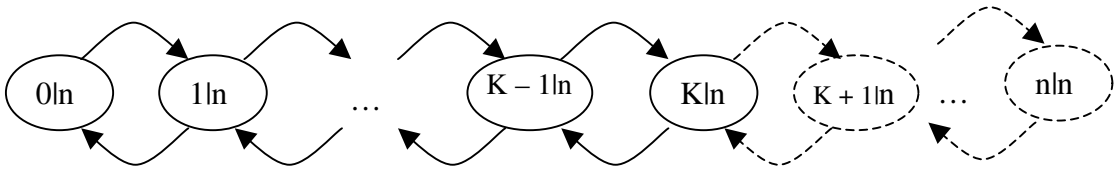


Figure 5. The states for an infinite capacity node in separable networks.

Moreover, if we map out the state space of a finite network, we will see that the blocking probability of a particular node consists of further partitions. These partitions are the marginal probabilities of having j customers blocked by node i in the upstream nodes.

For example, if node j has K customers, it will cause blocking. Since the number of customers in the network is at most n, 1 to *(n - K)* customers can be blocked, and awaiting for node j to finish. These customers are not necessarily present in the immediate upstream node but can be situated in previous nodes as well.

Hence, using these blocking state partitions (Figure 4), limited space of a finite node can be extended to virtual infinite buffer spaces which represent the possibility of having *K + 1, K + 2, ... ,n* customers. Concurrently, for an infinite capacity node in a product-form network, the state alterations will be as in Figure 5.

To show the differences from the finite node, the dissimilar states are drawn in dashed lines. With virtual buffer spaces of a finite node, the correspondence to an infinite capacity node can be established. As can be seen in Figure 5, the partitions of blocking state will be equivalent to *K + 1, K + 2, ... ,n* customers in the infinite node.

Assuming this notion, the blocking probability can be expressed as the sum of conditional probabilities of having more than K_i buffer space jobs at node i when there are n jobs in the equivalent infinite capacity node. This can be formulated as:

$$P_{Ki} = \sum_{j=Ki+1}^{n} \pi_i(j|n) = 1 - \sum_{j=0}^{Ki} \pi_i(j|n) \qquad (3)$$

Certainly, for a finite capacity node the state changes are not modeled by a continuous Markov chain (CTMC) and are not independent from other nodes. However, by introducing effective service rates, the blocking probabilities can be approximated.

Table 1. Parameters of the 4-node cyclic network.

	I	**II**	**III**	**IV**
μ_I	4	1	3	2
c_i	1	1	1	1
K_i	:	5	:	:

We note that while closed queueing networks have been widely studied, to the best of our knowledge, there has been no closed-form expression or other approximation cited for the estimation of the blocking probability for closed finite networks. The blocking probability from the $M/M/1/K$ finite capacity system is often used, and the probability of finding the finite node i with capacity K_i saturated is well known as:

$$P_{Ki} = \frac{(1-\rho)\rho^{Ki}}{1-\rho^{Ki+1}} \qquad (4)$$

where ρ is the utilization ratio.

We compare our blocking probability estimation calculated by Eq. 3 with Expression 4 in a numerical example with the parameters given in Table 1. The network consists of a serially constructed 4-node network with single servers. The server at node 1 is the fastest, while the one in node 2 is the slowest. We assign a finite space of 5 to node 2 so that it is the only source of blocking and thus the probability of blocking is high. The other nodes have infinite capacity.

The throughput rate from simulation is used for the calculation of the utilization ratio, ρ_2, which is needed for the M/M/1/K blocking probability. This is from the observation that the network is cyclic and the throughput rates of all nodes are equal by the conservation of flow.

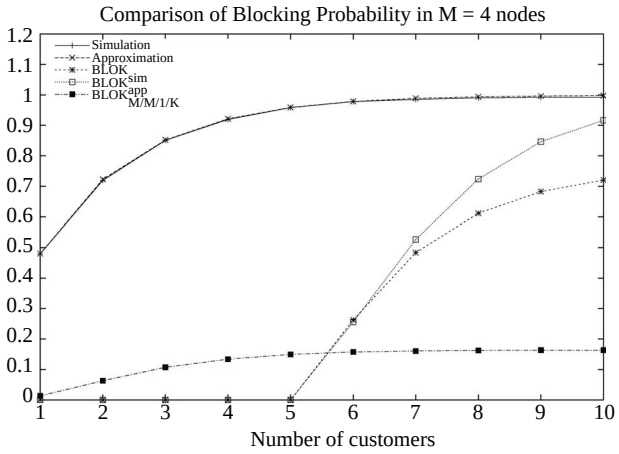


Figure 6. The blocking probability comparison in M = 4 nodes.

In Figure 6, the blocking probability estimates for node 2 are plotted on the lower curves of the figure, along with the throughputs in the top 2 curves from simulation and the approximation algorithm, EMVA, as described in Section 3.4.

When there are $N = 6$ customers in the system, blocking occurs. The probability of blocking rises gradually, as the number of customers increases. Our blocking probability estimation not only provides an exact starting point for the blocking but also mimics its behavior. The estimation from M/M/1/K fails to capture the increase in blocking, since the throughput of the system remains steady after N = 6.

Algorithm

The approximation method can be summarized as follows:

Step 1: Initialization. For i =*1, ... ,M*
$\pi_i(0|0)$, $P_i(0) = 0$

Step 2: Iteration: n = *1,2, ... ,N*

Step 2.1: For i = *1, ... ,M* compute the service rates

$$\mu_i(n) = \begin{cases} n\mu_i & if \quad n \le c_i \\ c_i\mu_i & if \quad n > c_i \end{cases}$$

Step 2.2: For i =*1, ... ,M* compute the effective service rates

$$\frac{1}{\mu_i^e(n)} = \begin{cases} \frac{1}{\mu_i} & \text{for all } i \text{ for } n = 1 \\ \frac{1}{\mu_i(n)} + P_{i+1}(n-1)\frac{1}{\mu_h(n)} & \text{for all } i \text{ for } n = 2, ..., N \end{cases}$$

μ_h is the service rate of the holding node

$\mu_{h=}\mu_{i+1}$ for exponentially distributed service rates for tandem networks

Step 2.3: For i = *1, ... ,M* compute the mean response time

$$W_i(n) = \sum_{j=1}^{n} \frac{j}{\mu_i^e(j)} \pi_i(j-1|n-1)$$

Step 2.4. Compute the system throughput

$$\lambda(\mathrm{n}) = \frac{n}{\sum_{i=1}^{M} W_i(n) * V_i}$$

Step 2.5. For i = *1, ... ,M* compute the conditional probabilities

$$\pi_i(j|n) = \frac{\lambda(n)}{\mu_i^e(j)} \pi_i(j-1|n-1) \text{ for } j = 1, \ldots, n$$

$$\pi_i(0|n) = 1 - \sum_{l=1}^{M} \pi_i(j|n)$$

Step 2.6: For i = *1, ... ,M* calculate the blocking probabilities

$$P_i(n) = 1 - \sum_{j=0}^{Ki} \pi_i(j|n)$$

The algorithm mentioned above has been implemented and was a part of the extensive study in Yuzukirmizi (2005). We give a brief study of the experiments in Section 4, which also include a comparison and assessment of the above method.

Assessment of the Method

Numerical examples

The algorithm described above was implemented and executed on several numerical experiments. We analyzed series, merge, and splitting topologies under carefully chosen system parameters to investigate algorithms in various circumstances. For each experiment, comparisons with simulation as well as with case studies by other authors are included, where applicable, to validate the proposed methods.

Simulation experiments are conducted with ARENA version 5.0 with 10,000 time units and 20 replications. System throughput, λ, is the primary measure. The percentage deviations, with respect to the simulation of listed methods, are also presented in comparison tables.

Single server comparison In this first set of tables (Tables 2-4) the algorithms are compared with existing methods published in the literature for cyclic networks with single servers to give a relative assessment. In each case, results obtained from the proposed EMVA method (see those with λ_{EMVA}) were compared with simulation results (λ_{Sim}). Results from other relevant methods are also included whenever possible. These include methods developed by Suri and Diehl (1986) (λ_{SD}), Akyildiz (1988b) (λ_{Aky}), Dallery and Frein (1986) (λ_{DF}), for cyclic

queues in which the first node has infinite capacity, and Liu et al. (1992) (λ_{Liu}), Zhuang et al. (1994) (λ_{Zhu}) and Onvural and Perros (1989) for general cyclic queues. The relative percentage errors, $\Delta\%$, are also provided.

Table 2. Throughput rate comparison of 5-node cyclic queue when the first node has infinite capacity.
$M = 5$
μ_i = (1,0.5,1,0.5,1) , K_i = (:3,3,3,3), c_i = (1, 1, 1, 1, 1)

	λ_{Sim}	λ_{EMVA}	$\Delta\%$	λ_{SD}	$\Delta\%$	λ_{Aky}	$\Delta\%$	λ_{DF}	$\Delta\%$	λ_{Liu}	$\Delta\%$	λ_{Zhu}	$\Delta\%$
5	0.367	0.366	–0.0%	0.364	–0.8%	0.367	0.0%	0.356	–3.00%	0.367	0.0%	0.367	0.0%
6	0.389	0.388	–0.1%	0.381	–2.0%	0.39	0.2%	0.382	–1.80%	0.39	0.2%	0.39	0.2%
7	0.404	0.403	–0.2%	0.392	–2.9%	0.407	0.7%	0.4	–0.99%	0.407	0.7%	0.407	0.7%
8	0.413	0.413	0.0%	0.398	–3.6%	0.42	1.6%	0.416	0.73%	0.42	1.6%	0.42	1.6%
9	0.417	0.419	0.6%	0.4	–4.0%	0.42	0.7%	0.426	2.16%	0.425	1.9%	0.425	1.9%
10	0.419	0.422	0.9%	0.4	–4.5%	0.42	0.2%	0.43	2.63%	0.425	1.4%	0.425	1.4%
11	0.419	0.424	1.2%	0.401	–4.3%	0.42	0.2%	0.43	2.63%	0.425	1.4%	0.425	1.4%
12	0.419	0.424	1.1%	0.401	–4.3%	0.42	0.2%	0.43	2.63%	0.425	1.4%	0.425	1.4%
13	0.419	0.422	0.9%	0.401	–4.3%	0.42	0.2%	0.43	2.63%	0.425	1.4%	0.425	1.4%

Table 3. Throughput rate comparison of 4-node cyclic queue.
$M = 4$
μ_i = (3, 2, 4, 2) , K_i = (6, 2, 2, 4), c_i = (1, 1, 1, 1, 1)

	λ_{Exact}	λ_{EMVA}	$\Delta\%$	λ_{OP}	$\Delta\%$	λ_{Liu}	$\Delta\%$	λ_{Zhu}	$\Delta\%$
4	1.379	1.3777	–0.09%	1.391	0.87%	1.369	–0.73%	1.377	–0.15%
5	1.477	1.474	–0.20%	1.506	1.96%	1.454	–1.56%	1.466	0.74%
6	1.541	1.5321	–0.58%	1.57	1.88%	1.578	2.40%	1.529	–0.78%
7	1.583	1.562	–1.33%	1.593	0.63%	1.603	1.26%	1.576	–0.44%
9	1.606	1.5658	–2.50%	1.597	–0.56%	1.606	0.00%	1.576	–1.87%
10	1.587	1.5512	–2.26%	1.584	–0.19%	1.603	1.01%	1.576	–0.69%
11	1.549	1.5318	–1.11%	1.559	0.65%	1.578	1.87%	1.529	–1.29%
12	1.487	1.511	1.61%	1.495	0.54%	1.454	–2.22%	1.466	–1.41%
13	1.385	1.4916	7.70%	1.385	0.00%	1.369	–1.16%	1.377	–0.58%

Table 4. Throughput rate comparison of 5-node cyclic queue with all nodes having finite capacity.
$M = 5$
μ_i = (3, 2, 4, 2, 1) , K_i = (4, 3, 2, 4, 2), c_i = (1, 1, 1, 1, 1)

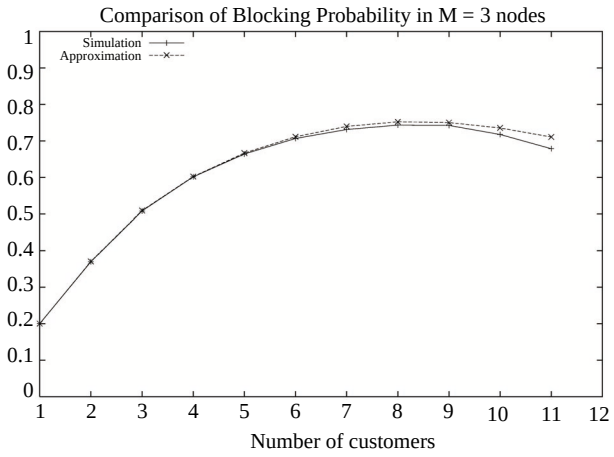
	λ_{Sim}	λ_{EMVA}	$\Delta\%$	λ_{OP}	$\Delta\%$	λ_{Liu}	$\Delta\%$	λ_{Zhu}	$\Delta\%$
4	0.849	0.8507	0.20%	0.843	–0.71%	0.854	0.59%	0.854	0.59%
5	0.895	0.9018	0.76%	0.888	–0.78%	0.908	1.45%	0.889	–0.67%
6	0.917	0.929	1.31%	0.914	–0.33%	0.931	1.53%	0.915	–0.22%
7	0.926	0.9416	1.68%	0.925	–0.11%	0.932	0.65%	0.921	–0.54%
8	0.93	0.9438	1.48%	0.93	0.00%	0.933	0.32%	0.921	–0.97%
10	0.931	0.9296	–0.15%	0.931	0.00%	0.933	0.21%	0.921	–1.07%
11	0.929	0.9169	–1.30%	0.928	–0.11%	0.932	0.32%	0.921	–0.86%
12	0.923	0.9026	–2.21%	0.924	0.11%	0.931	0.87%	0.915	–0.87%
13	0.909	0.8884	–2.27%	0.912	0.33%	0.908	–0.11%	0.889	–2.20%
14	0.87	0.875	0.57%	0.892	2.53%	0.854	–1.84%	0.854	–1.84%

Multiple server comparison In the following figures (Figures 7-12) experimental results with multiple servers are presented. The approximation results are compared only to simulation since other authors' methods are only designed for cyclic queues with single servers. The throughput of the system calculated from simulation, λ_{sim}, is compared with the value obtained from the EMVA algorithm, λ_{EMVA}. As before, the relative percentage deviations are presented to compare to simulation. The 95% confidence interval for the simulation runs, δ, is also provided.

In Figures 7-9, 3-node networks are compared with simulation. In the first 2 of these experiments, the service times are balanced in according to their server numbers, and are unbalanced in the third. The number of customers increases until deadlock occurs.

In Figures 10 and 11, a 5-node network with balanced and unbalanced service rates is presented. Except extreme blocking, the approximation method estimates the throughput accurately. Also note that the algorithm is consistent and does not have unexpected state.

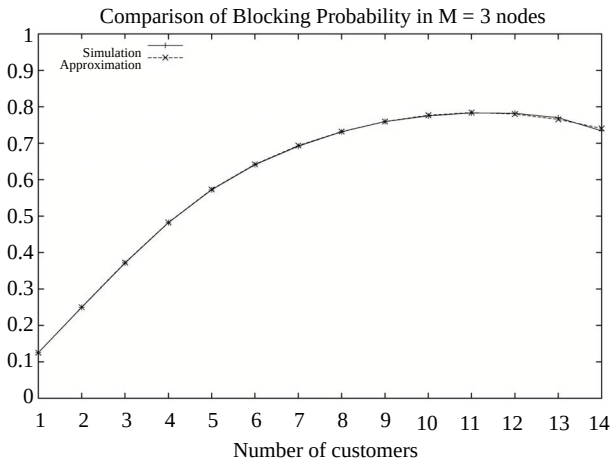
$M = 3$
$\mu_i = (.3333, 1, 1)$, $c_i = (3, 1, 1)$ and $K_i = (4, 4, 4)$



	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
1	0.2003	70.0012	0.2	0.1%
2	0.3697	70.0013	0.3704	–0.2%
3	0.5082	70.0016	0.5094	–0.2%
4	0.6014	70.0021	0.6022	–0.1%
5	0.6641	70.0028	0.6666	–0.4%
6	0.7066	70.0025	0.7115	–0.7%
7	0.7313	70.0027	0.7398	–1.2%
8	0.7434	70.0026	0.7523	–1.2%
9	0.7424	70.0026	0.7502	–1.1%
10	0.7176	70.0027	0.7354	–2.5%
11	0.6786	70.0023	0.7106	–4.7%

Figure 7. Throughput plot and comparison of a 3-node network with balanced service rates.

$M = 3$
$\mu_i = (.3333, .5, .3333)$, $c_i = (3, 2, 3)$ and $K_i = (5, 5, 5)$



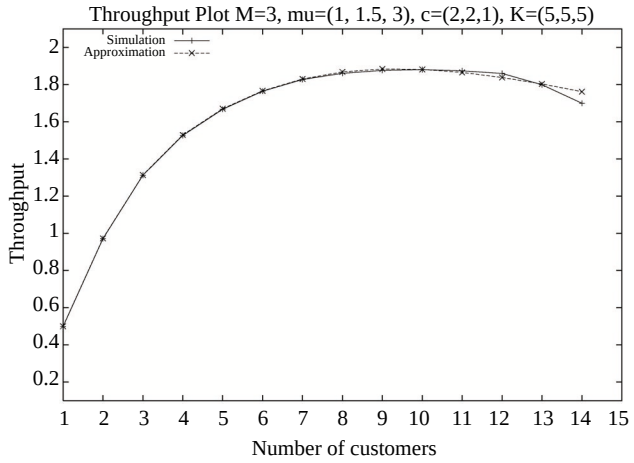
	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
1	0.1252	70.001	0.125	0.2%
2	0.2496	70.001	0.25	–0.2%
3	0.3713	70.001	0.3721	–0.2%
4	0.4818	70.002	0.4824	–0.1%
5	0.5721	70.002	0.5731	–0.2%
6	0.6409	70.002	0.6422	–0.2%
7	0.692	70.003	0.6936	–0.2%
8	0.7314	70.003	0.7319	–0.1%
9	0.7592	70.003	0.7595	0.0%
10	0.775	70.003	0.777	–0.3%
11	0.7832	70.003	0.784	–0.1%
12	0.7822	70.003	0.7801	0.3%
13	0.7699	70.003	0.7653	0.6%
14	0.734	70.003	0.7406	–0.9%

Figure 8. Throughput plot and comparison of a 3-node network with balanced service rates.

Lastly, Figure 12 displays the comparison in an 8-node network. Considering the fact that customer size is up to 45 and blocking begins with N = 5, the algorithm is quite precise.

General topology comparison In this example, we evaluate the network in Figure 13 with split and merge topologies. The service stations have arbitrary number of servers and service rates. The throughput plot as well as the comparison with the simulation values is presented in Figure 14.

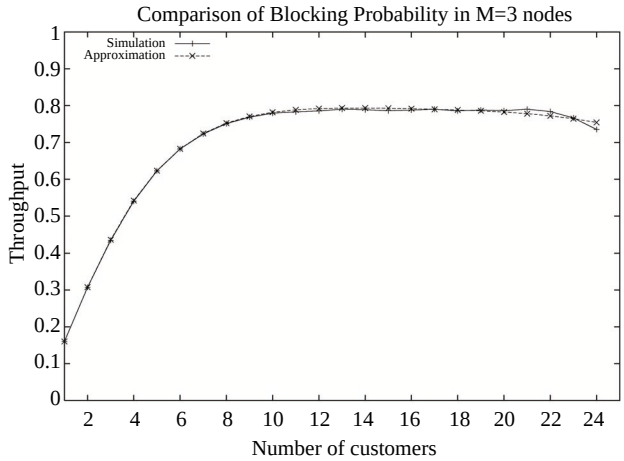
$M = 3$
$\mu_i = (1, 1.5, 3)$, $c_i = (2, 2, 1)$ and $K_i = (5, 5, 5)$



	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
1	0.4987	70.0018	0.5	–0.3%
2	0.9713	70.0028	0.973	–0.2%
3	1.3128	70.0034	1.3136	–0.1%
4	1.5268	70.0028	1.5294	–0.2%
5	1.6674	70.0039	1.6707	–0.2%
6	1.7648	70.0044	1.7665	–0.1%
7	1.8276	70.0045	1.8307	–0.2%
8	1.8608	70.0054	1.8686	–0.4%
9	1.8765	70.0053	1.8842	–0.4%
10	1.8811	70.0058	1.8818	0.0%
11	1.8741	70.0044	1.8655	0.5%
12	1.8607	70.0045	1.8387	1.2%
13	1.7997	70.0042	1.8036	–0.2%
14	1.6999	70.0044	1.7623	–3.7%

Figure 9. Throughput plot and comparison of a 3-node network with unbalanced service rates.

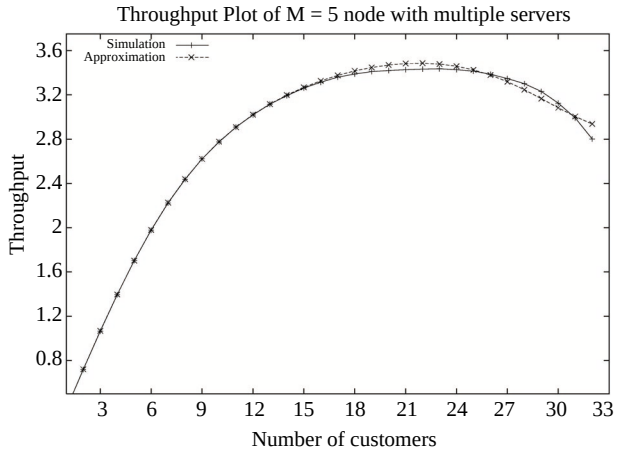
$M = 5$
$\mu_i = (0.8, 0.8, 0.8, 0.8, 0.8)$, $c_i = (3, 2, 1, 2, 3)$ and $K_i = (5, 5, 5, 5, 5)$



	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
5	0.6227	70.002	0.6231	–0.1%
8	0.751	70.003	0.7523	–0.2%
10	0.7794	70.003	0.7818	–0.3%
12	0.7854	70.005	0.7918	–0.8%
15	0.7859	70.004	0.7928	–0.9%
16	0.7878	70.004	0.7917	–0.5%
18	0.7858	70.004	0.7882	–0.3%
20	0.7862	70.004	0.7824	0.5%
22	0.7839	70.004	0.7721	1.5%
23	0.7665	70.003	0.7642	0.3%
24	0.7353	70.003	0.7543	–2.6%

Figure 10. Throughput plot and comparison of a 5-node network with unbalanced service rates.

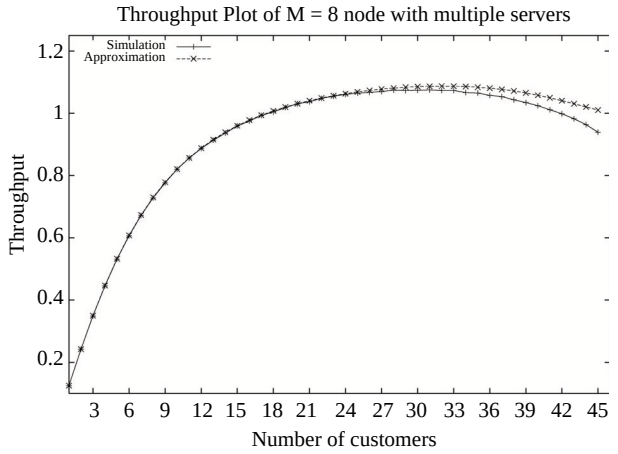
$M = 5$
$\mu_i = (4, 1, 3, 2, 1.5)$, $c_i = (1, 5, 2, 2, 3)$ and $K_i = (5, 9, 6, 6, 7)$



	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
5	1.7022	70.0025	1.7023	0.0%
8	2.4357	70.0038	2.4387	–0.1%
10	2.7744	70.0036	2.7765	–0.1%
12	3.0186	70.0047	3.0207	–0.1%
15	3.2607	70.0044	3.2675	–0.2%
18	3.3875	70.0051	3.4155	–0.8%
20	3.4193	70.0042	3.4692	–1.5%
22	3.4316	70.0064	3.4856	–1.6%
25	3.4145	70.0049	3.4253	–0.3%
28	3.3004	70.0053	3.2456	1.7%
30	3.1244	70.0056	3.0826	1.3%
32	2.8011	70.0049	2.9366	–4.8%

Figure 11. Throughput plot and comparison of a 5-node network with unbalanced service rates.

$M = 8$
$\mu_i = (1.2, 1{,}0.8, 1.2, 1, 0.8, 1.2, 1)$, $c_i = (1{,}2, 3, 1, 2, 3, 1, 2)$ and $K_i = (8, 6, 4, 8, 6, 4, 8, 6)$



	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
5	0.5317	70.0013	0.5325	–0.2%
8	0.7284	70.0021	0.7297	–0.2%
10	0.8203	70.0026	0.8206	0.0%
12	0.8869	70.0023	0.8883	–0.2%
15	0.9592	70.0023	0.9599	–0.1%
18	1.0044	70.0024	1.0078	–0.3%
20	1.0307	70.0031	1.031	0.0%
22	1.0479	70.0034	1.0491	–0.1%
25	1.0647	70.003	1.0687	–0.4%
28	1.0744	70.0034	1.081	–0.6%
30	1.074	70.0034	1.0857	–1.1%
32	1.0735	70.0044	1.0874	–1.3%
35	1.0653	70.0037	1.0839	–1.7%
38	1.0434	70.0032	1.0717	–2.7%
40	1.0243	70.0026	1.0581	–3.3%
42	0.9984	70.0032	1.0406	–4.2%
45	0.9391	70.0030	1.0104	–7.6%

Figure 12. Throughput plot and comparison of an 8-node network with unbalanced service rates.

Analysis of all numerical results demonstrates very good accuracy for the EMVA. The absolute relative errors for the throughput are less than 10% and in most cases below 2%. In general, the throughput values that correspond to the first half of the curve are estimated very precisely. Although the estimates on the second half are not as good as previous ones, they are still normally within 5% accuracy. Large errors are usually observed for systems under extreme congestion that exhibit considerable blocking and those with a high level of interdependency among the nodes.

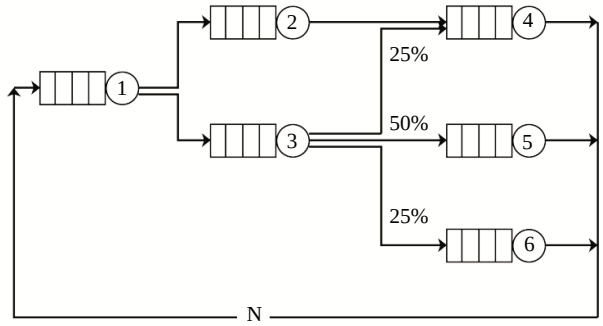
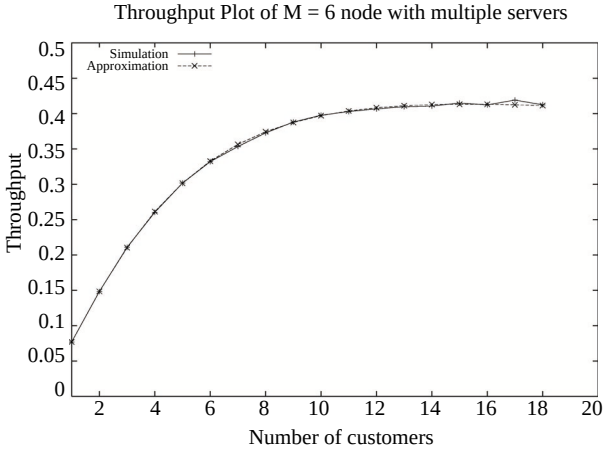


Figure 13. A 6-node network with split and merge topologies.

$$M = 6$$

μ_i= (2.5, 10, 5, 3, 6.66, 4), c_i= (2, 2, 1, 2, 2, 1) and K_i= (12, 4, 8, 5, 4, 2)



	λ_{Sim}	δ	λ_{EMVA}	$\Delta\%$
3	0.2108	70.0016	0.21	0.2%
4	0.2604	70.0018	0.261	–0.3%
5	0.3016	70.0024	0.302	0.0%
6	0.3319	70.0022	0.333	–0.2%
7	0.3533	70.0024	0.356	–0.9%
8	0.3726	70.0034	0.374	–0.5%
9	0.388	70.0042	0.387	0.2%
10	0.3976	70.0037	0.397	0.2%
11	0.4028	70.0049	0.404	–0.2%
12	0.4066	70.0039	0.408	–0.4%
13	0.4096	70.0036	0.411	–0.4%
14	0.4106	70.0044	0.413	–0.5%
15	0.4149	70.0041	0.413	0.4%
16	0.4123	70.0036	0.413	–0.2%
17	0.4191	70.0017	0.412	1.6%
18	0.4124	70.0067	0.411	0.3%

Figure 14. Throughput plot and comparison of a 6-node network with split and merge junctions.

Hypothesis testing

We can further reveal the accuracy of the approximation method using hypothesis testing. With the assumption that the absolute relative errors, $|\Delta|$, in the numerical experiments represents a random sample from a normal distribution, a t-test can be applied for hypothesis testing.

For example, to determine whether the EMVA on average produces more than 1% absolute relative error in single server experiments, we intend to test:

H_0: $\mu = 1\%$

H_1: $\mu > 1\%$

where μ is the mean of all possible values for $|\Delta|$. From our experimental data, presented in Table 2 to Table 4, the computed value of the t-statistics with the sample size of n = 28 is:

$$t = \frac{|\Delta| - \mu_0}{s_{|\Delta|}/\sqrt{n}} = \frac{0.0124 - 0.01}{0.0147/\sqrt{28}} = 0.859$$

The critical value $t_{\alpha,\theta}$, with 95% confidence interval ($\alpha = 0.05$) and θ = n - 1 = 27 degrees of freedom is 1.703. Since $t < t_{0.05,27}$, based on the data we do not reject H_0 and conclude that the average absolute relative error is not significantly greater than 1%. Note that, in all of the experiments presented, only blocking values of customers in the system are considered.

Let us also test the above hypothesis in multiple server experiments presented in Figures 7-12, including the values omitted for presentation purposes. To determine whether EMVA yields more than 1% absolute relative error, we get the computed value of the t-statistics with the sample size of n = 112:

$$t = \frac{|\Delta| - \mu_0}{s_{|\Delta|}/\sqrt{n}} = \frac{0.00984 - 0.01}{0.0135/\sqrt{112}} = -0.1264$$

The critical value of $t_{0.05,111}$ is approximately 1.659. Again, we do not reject the H_0, and conclude that absolute relative error is not significantly greater than 1%.

The results are also satisfactory for the networks with split and merge junctions, which are more difficult to model than the cyclic systems. Nonetheless, the upper bound posed on the number of customers by the deadlock-free notion avoids extreme congestion in these topologies and yields much better results.

Conclusion

In this study, we introduce an approximate MVA technique for closed finite queueing networks. The method incorporates slight changes to the MVA algorithm and uses insights from the EM. It is comparatively easy to implement and from computational tests we found that performance measures of closed finite queues are very accurate.

As validated in this study, the Expanded Mean Value Analysis (EMVA) is an effective technique to be employed in finite queueing networks. This assurance gives us the motivation to use the algorithm in the optimization of buffer allocation and apply this method to highly complex, computationally challenging multi-class finite queues. Also widely used, important real-life applications of closed queueing networks from computer systems to communication networks, job-shop manufacturing systems and recent applications of it to re-entrant lines such as semiconductor wafer fabrication make inevitable the study of these types of applications.

As an extension to this work, the assumption of the exponential service times can be relaxed and algorithms with general service times can be approximated. Further, approximation methods for the performance evaluation of networks with unreliable servers can be developed.

References

Akyildiz, I.F., "Exact Product Form Solution for Queueing Networks with Blocking" IEEE Transactions on Computers, 36, 122-125, 1987.

Akyildiz, I.F., "Mean Value Analysis for Blocking Queueing Networks." IEEE Transactions on Software Engineering, 14, 418-427, 1988.

Akyildiz, I.F., "On the Exact and Approximate Throughput Analysis of Closed Queueing Networks with Blocking", IEEE Trans. on Software Engineering, 14, 62-69, 1988.

Balsamo, S., "Properties and Analysis of Queueing Network Models with Finite Capacities", Lecture Notes in Comp. Sci., 729, 21-52, 1993.

Bard, Y., "Some Extensions to Multi-Class Queueing Network Analysis", 4th Int. Symp. on Modeling and Performance Evaluation of Computer Systems, in Queueing Networks and Markov Chains by G. Bolch, S. Greiner, H. de Meer and S. Trivedi, John Wiley and Sons, 1998, 1, 51-62, 1979.

Baskett, F., Chandy, K.M., Muntz, R.R. and Palacios-Gomez, F., "Open, Closed and Mixed Networks for Queues with Different Classes of Customers", Journal of A.C.M., 22, 248-260, 1975.

Bolch, G., de Meer, H., Greiner, S. and Trivedi, K.S., "Queueing Networks and Markov Chains, Modeling and Performance Evaluation with Computer Science Applications", John Wiley and Sons, New York, 1998.

Buzen, J., "Computational Algorithms for Closed Queueing Networks With Exponential Servers", Communications of A.C.M., 16, 527-531, 1973.

Chandy, K., Herzog, U. and Woo, L., "Parametric Analysis of Queueing Networks", IBM Journal of Research and Development, 19, 36-42, 1975.

Cooper, R.B., Introduction to Queueing Theory, North Holland, 1981.

Dallery, Y. and Frein, Y., "A Decomposition Method for the Approximate Analysis of Closed Queueing Networks with Blocking", in Queueing Networks with Blocking, eds H.G. Perros and T. Altiok (Elsevier Science/North Holland, Amsterdam), 1986.

Frein, Y. and Dallery, Y., "Analysis of Closed-Loop Manufacturing System with Finite Buffers", Applied Stochastic Models and Data Analysis, 9, 111-125, 1993.

Gonzales, E.A., "Optimal Resource Allocation in Closed Finite Queueing Networks with Blocking after Service", PhD thesis, University of Massachusetts-Amherst, Department of Mechanical and Industrial Engineering, 1997.

Gordon, W. and Newell, G., "Closed Queueing Systems with Exponential Servers." Operations Research, 15, 254-265, 1967a.

Gordon, W. and Newell, G., "Cyclic Queueing Systems with Restricted Queues", Operations Research. 15, 266-277, 1967b.

Jackson, J., "Job Shop-like Queueing Systems", Management Science, 10, 131-142, 1963.

Jain, S. and Smith, J.M., "Open Finite Queueing Networks with M/M/C/K Parallel Servers", Computers Ops. Res., 21, 297-317, 1994.

Kerbache, L. and Smith, J.M., "The Generalized Expansion Method for Open Finite Queueing Networks", European Journal of Operational Research, 32, 448-461, 1987.

Kerbache, L. and Smith, J.M., "Asymptotic Behavior of the Expansion Method for Open Finite Queueing Networks", Computers and Operations Research, 15, 157-169, 1988.

Liu, X-G, Xhuang, L. and Buzacott, J.A., "A Decomposition Method for Throughput Analysis of Cyclic Queues with Production Blocking", 2^{nd} Int. Workshop on Queueing networks with Blocking, Raleigh, North Carolina, 1992.

Nain, P., "Basic Elements of Queueing Theory, Lecture Notes", 1997-1998, 1998.

Onvural, R., "A Note on the Product Form Solutions of Multi-Class Closed Queueing Networks with Blocking", Performance Evaluation, 10, 247-253, 1989.

Onvural, R., "A Survey of Closed Queueing Networks with Blocking", ACM Computing Surveys, 22, 83-122, 1990.

Onvural, R. and Perros, H., "On Equivalences of Blocking Mechanism in Queueing Networks with Blocking", Operations Research Letters, 5, 293-298, 1986.

Onvural, R. and Perros, H, "Approximate Throughput Analysis of Cyclic Queueing Networks with Finite Buffers", IEEE Trans. on Software Eng., 15, 800-808, 1989.

Papadopoulos, H.T., Heavey, C. and Browne, J., "Queueing Theory in Manufacturing Systems, Analysis and Design", Chapman and Hall, London, 1993.

Reiser, M. and Lavenberg, S., "Mean Value Analysis for Closed Multi-chain Queueing Networks", Journal of A.C.M., 27, 313-322, 1980.

Schmidt, R., "An Approximate MVA Algorithm for Exponential, Class Dependent Multiple Servers", Performance Evaluation, 29, 245-254, 1997.

Suri, R. and Diehl, G.W., "A Variable Buffer Size Model and Its Use in Analyzing Closed Queueing Networks with Blocking", Management Science, 32, 206-224, 1986.

Wang, H., "Approximate MVA Algorithms for Solving Queueing Network Models", Master's thesis, University of Toronto, Dept. of Computer Science, 1997.

Yuzukirmizi, M., "Finite Closed Queueing Networks with Multiple Servers and Multiple Chains", PhD thesis, University of Massachusetts-Amherst, Department of Mechanical and Industrial Engineering, 2005.

Zhuang, L., Buzacott, J. and Liu, X.G., "Approximate Mean-Value Performance Analysis of Cyclic Queuing-Networks with Production Blocking", Queueing Systems, 16, 139-165, 1994.