

The MVA Priority Approximation

RAYMOND M. BRYANT and ANTHONY E. KRZESINSKI

IBM Thomas J. Watson Research Center

and M. SEETHA LAKSHMI and K. MANI CHANDY

University of Texas at Austin

A Mean Value Analysis (MVA) approximation is presented for computing the average performance measures of closed-, open-, and mixed-type multiclass queuing networks containing Preemptive Resume (PR) and nonpreemptive Head-Of-Line (HOL) priority service centers. The approximation has essentially the same storage and computational requirements as MVA, thus allowing computationally efficient solutions of large priority queuing networks. The accuracy of the MVA approximation is systematically investigated and presented. It is shown that the approximation can compute the average performance measures of priority networks to within an accuracy of 5 percent for a large range of network parameter values. Accuracy of the method is shown to be superior to that of Sevcik's shadow approximation.

Categories and Subject Descriptors: D.4.4 [Operating Systems]: Communications Management—*network communication*; D.4.8 [Operating Systems]: Performance—*modeling and prediction*; *queuing theory*

General Terms: Performance, Theory

Additional Key Words and Phrases: Approximate solutions, error analysis, mean value analysis, multiclass queuing networks, priority queuing networks, product form solutions

1. INTRODUCTION

Multiclass queuing networks with product-form solutions [3] are widely used to model the performance of computer systems and computer communication networks [11]. The effective application of these models is largely due to the efficient computational methods [5, 9, 13, 18, 21] that have been developed for the solution of product-form queuing networks. However, many interesting and significant system characteristics cannot be modeled by product-form networks. Priority service disciplines are one such system characteristic.

An earlier version of this paper was presented at the 1983 ACM Sigmetrics Conference [4]. This paper combines the results of [4] with those of [7]. M. Seetha Lakshmi and K. Mani Chandy's research was supported in part by National Science Foundation grant MCS-8101911.

Authors' present addresses: R. M. Bryant and M. S. Lakshmi, IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598; K. M. Chandy, Department of Computer Science, The University of Texas at Austin, Austin, TX 78712; A. E. Krzesinski, Institute for Applied Computer Science, University of Stellenbosch, 7600 Stellenbosch, South Africa.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1984 ACM 0734-2071/84/277-0335 \$00.75

Priority service disciplines are commonly used in processor scheduling algorithms to give preferential service to interactive processing and in communications networks to give preferential service to message (as opposed to file transfer) tasks. Solutions of queuing networks containing priority centers are therefore important in the proper representation of such systems. However, exact solutions of priority networks have produced only a few results because of the computational expense of solving the global balance equations for non-product-form networks. For example, the preemptive and nonpreemptive $M/G/1//L$ priority queue [15] can be regarded as a closed two-center network with L customers where the second (nonpriority) center is an infinite server. Solutions have been found [1] for homogeneous central server networks where all the priority classes have identical service times and routing frequencies. A recent analysis [16] has presented exact results for nonhomogenous two-center networks where each center has a priority discipline.

Rather than develop exact solutions which, of necessity, are confined to small networks with small customer populations, an alternative approach is to develop computationally efficient and accurate approximate solutions. Typically, such approximations achieve computational efficiency by assuming that a network which does not satisfy local balance, nonetheless, has a product-form solution. Accuracy of the approximation is established by comparing the exact solution (or a simulation analysis) of the non-product-form network with the approximation solution. Careful construction of test cases for this comparison can provide guidance as to when the approximation will work well.

This paper presents a computationally efficient and accurate method for computing approximate solutions for priority queuing networks. The approximation is based upon the Mean Value Analysis (MVA) [21] solution method, which is particularly suited [2] for the development of heuristic solutions. The MVA priority approximation has the following advantages. The approximation, which applies to both preemptive resume (PR) and nonpreemptive Head-of-Line (HOL) priority centers can readily be installed in existing MVA solvers. The approximation also applies to open and mixed queuing networks, and to networks with more than one priority center. The approximation does not require either class or center aggregation [22] and is computationally more efficient than previously reported approximations for priority networks [22, 24].

Previous approximations for priority queuing networks are summarized in Section 2. Section 3 reviews the MVA equations for product-form networks. Section 4 develops the MVA priority approximation. Section 5 compares exact and approximate solutions for queuing networks containing PR or nonpreemptive HOL priority centers.

2. APPROXIMATIONS FOR PRIORITY QUEUING NETWORKS

Consider a Multiclass Queuing Network consisting of M service centers labeled $(1 \dots M)$. Let the customers belong to J closed chains with index set $(1 \dots J)$. Customers within each chain j belong to the same customer class. The priority classes are linearly ordered, class 1 having the highest priority and class J the lowest priority. Let μ_{ij} represent the service rate of a class j customer at service center i , $s_{ij}(= 1/\mu_{ij})$ be the equivalent mean service time, and ρ_{ij} represent the

utilization due to class j customers at service center i . The following sections summarize previous algorithms for approximate solutions to queuing networks containing priority centers.

2.1 Reduced Work-Rate Approximation

The reduced work-rate approximation [19, 20] is based on the observation that in a PR queuing system, lower priority customers “see” a server whose average capacity for work is reduced because of servicing of higher priority customers. For example, in a network consisting of one PR center and one Infinite Server (IS) center, one can obtain the class 1 performance measures by solving the network with all lower class customers removed from the model. If ρ_{p1} gives the class 1 utilization at the priority center, one can then solve for the class 2 measures using the adjusted service rate $\hat{\mu}_{p2} = \mu_{p2}(1 - \rho_{p1})$. This process can be repeated until all performance measures have been calculated. In general,

$$\hat{\mu}_{pj} = \mu_{pj} \left(1 - \sum_{k=1}^{j-1} \rho_{pk} \right).$$

Limited evidence [19] suggests that this approach is essentially as accurate as simulation.

The approach can be extended to networks consisting of several PR and IS centers provided that the service priorities of the customer classes at each PR center are identical. The approximation does not directly apply to networks where high-priority customers must compete for service with lower priority customers at one or more centers in the network (e.g., if the network contains a processor sharing center that serves both high- and low-priority customers or if high-priority customers at one center become low-priority customers at another center).

2.2 Shadow Approximation

The shadow approximation [24] can be thought of as an algorithm to apply the reduced work-rate approximation to a more general class of networks. It is used to solve networks containing PR (and not HOL) centers. Although, in principle, the approximation can be applied to networks with multiple PR centers, details of or experience with the shadow approximation in this case have not been reported in the literature.

In the shadow approximation method each priority center is replaced by J shadow centers where J is the number of priority classes. Each shadow center is visited by one priority class only. The service rate μ_{sj} of a class j customer at its dedicated shadow center s is given by $\mu_{sj} = \mu_{pj}(1 - \sum_{k=1}^{j-1} \rho_{pk})$. Each μ_{sj} thus represents the average service rate for class j customers at the priority center after the service demand of the higher priority customers has been met.

Exact values for the utilization ρ_{pk} are not available when computing the shadow service rates μ_{sj} . The following iteration is used to compute the μ_{sj} :

<i>INITIALIZE:</i> ρ_{pk}	$\forall 1 \leq k \leq J$
<i>WHILE</i> ρ_{pk} NOT CONVERGED DO	$\forall 1 \leq k \leq j$
$\mu_{sj} = \mu_{pj} \left(1 - \sum_{k=1}^{j-1} \rho_{pk} \right)$	$\forall 1 \leq j \leq J$

```

SOLVE  $(M + J - 1)$  CENTER BCMP1 NETWORK
COMPUTE  $\rho_{pk}$   $\forall 1 \leq k \leq J$ 
END

```

An efficient technique is available for initializing the ρ_{pk} for networks containing only two priority classes. The amount of storage and computation required to solve the shadow network is of the same order as that required to solve a $(M + J - 1)$ center product-form network where J fixed-rate centers are substituted as shadow replacements for each priority center.

If low-priority customers never compete for service with high-priority customers, then the iteration above converges immediately and the shadow approximation is equivalent to the reduced work-rate approximation.

Apart from small (four-center) networks with small population, no systematic investigation of the accuracy of the shadow approximation has been reported. Shadow solutions for priority networks with three to four customers in each of two priority classes have errors of about 20 percent [24].

2.3 The Composite Center and HAM Approximations

The composite center approximation [22] assumes a central server priority network to be locally balanced and aggregates the $(M - 1)$ nonpriority centers into a single composite center [6, 12] with a queue-length-dependent service rate. The reduced two-center model is solved exactly using global balance techniques. The method can be applied to networks containing PR or HOL priority centers.

However, as the number of classes and/or the number of customers per priority class reaches even moderate values (e.g., for classes with four customers in each class), the global analysis of the reduced model becomes too complex to be of practical value.

The complexity can be reduced [22] by partitioning the priority classes into three disjoint sets, namely, a designated class and two composite classes. One composite class contains a suitably weighted representation of all the network customers that have higher priority than the customers in the designated class. The other composite class represents the remaining customers that have a lower priority than the designated class. Each priority class is designated in turn, and center aggregation is applied to reduce each three-class M -center network to a three-class, two-center network. Approximate values for the performance measures of each customer class in the original priority network can be recovered from the global balance analysis of the two-center reduced networks.

Computational complexity limits the application of the composite center approximation to the solution of priority networks having a small number of centers and customers. The approximation was tested [22] on 36 priority models containing four to five centers, three to six classes and four to six customers. Errors on the order of 10 percent were reported for priority-center utilizations of about 0.6.

In principle, the composite center approximation can be applied to networks with more than one priority center. However, the computational cost of the global balance solution increases rapidly with the number of priority centers.

¹ BMCP is an acronym for Basket, Chandy, Muntz, and Palacios. The BMCP Network is also known as a product-form or separable network.

The Heuristic Aggregation Method (HAM) [17] essentially reduces to the composite center approximation when there is only one priority center in the network. If there are multiple priority centers (or, more generally, multiple non-BCMP centers in the network) the HAM attempts to reduce computational complexity by exactly solving a series of two-center, non-product-form networks. Each two-center network consists of one of the priority centers and an aggregate server intended to represent the rest of the network. From the solutions of these networks, the HAM creates a product-form representation for each priority center. This gives a product-form approximation to the original queuing network. Solving the approximate network produces a new set of parameters to be inserted in the two-center networks. This process is repeated until the performance measures converge.

Experimental evidence [17] for models with at most two priority queues shows that typical errors could be expected to be less than 10 percent; errors as high as 37.5 percent were reported. (These errors are *tolerance* errors [8]; for a discussion of this error measure see Section 5.)

3. MEAN VALUE ANALYSIS

The approximations discussed in this paper are based on the Mean Value Analysis (MVA) [21] method for computing the performance measures of product-form networks. Zahorjan and Wong [26] discuss the application of MVA to mixed networks, that is, product-form networks that contain open and closed chains. The MVA results for mixed networks of load-independent service centers are summarized here to illustrate how this method provides a computational framework within which the PR and HOL approximations are applied. (Approximations for priority centers with variable service rates are not considered in this paper.)

Consider a multiclass queuing network consisting of M service centers labeled $(1 \dots M)$. Let the customers belong to J chains labeled $(1 \dots J)$. A chain can be either *open* or *closed*. Let \mathcal{O} be the set of indices for open chains and \mathcal{C} be the set of indices for closed chains. (The terms *class* and *chain* are used synonymously.) For $j \in \mathcal{C}$, let N_j be the number of customers in chain j , and let $\mathbf{N} = (N_j | j \in \mathcal{C})$ denote the population vector of the closed chains. For a closed chain j , let $\mathbf{N} - \mathbf{1}_j$ denote a population vector with one fewer customer in class j . Let θ_{ij} denote the average number of visits a chain j customer makes to center i between successive visits to an arbitrarily chosen center v and s_{ij} represent the mean service time requirement of chain j customers at center i . For $j \in \mathcal{O}$, let Λ_j be the arrival rate of class j customers at center v . Let $W_{ij}(\mathbf{N})$, $L_{ij}(\mathbf{N})$, $T_{ij}(\mathbf{N})$, and $\rho_{ij}(\mathbf{N})$ denote the average wait time, queue length, throughput, and utilization of chain j customers at center i when the closed chain population is \mathbf{N} .

MVA for mixed networks [26] begins by first calculating the throughputs and utilizations for the open classes. These quantities are independent of the closed chain population and can be calculated directly from Λ_j , θ_{ij} , and s_{ij} . Next, a closed network containing only the closed chains is solved by the normal MVA recursion. In this closed model the service times s_{ij} are adjusted by dividing by $1 - \sum_{k \in \mathcal{O}} \rho_{ik}$ to allow for the presence of the open chain customers. Once the performance characteristics of the closed network at population \mathbf{N} are known, the wait times and mean queue lengths for the open chain customers can be calculated.

This algorithm can be summarized as follows:

INITIALIZE:

$$\begin{aligned} L_{ij}(\mathbf{0}) &= 0 & \forall 1 \leq i \leq M, \forall j \in \mathcal{L} \\ T_{ij} &= \theta_{ij} \Lambda_j; \quad \rho_{ij} = T_{ij} s_{ij} & \forall 1 \leq i \leq M, \forall j \in \mathcal{O} \end{aligned}$$

REPEAT

$$\begin{aligned} \forall \mathbf{n} \ni \mathbf{0} \leq \mathbf{n} \leq \mathbf{N}: \\ W_{ij}(\mathbf{n}) &= \frac{s_{ij} [1 + \sum_{k \in \mathcal{L}} L_{ik}(\mathbf{n} - \mathbf{1}_j)]}{1 - \sum_{k \in \mathcal{O}} \rho_{ik}} & \forall 1 \leq i \leq M, \forall j \in \mathcal{L}, i \text{ a FCFS} \\ & & \text{or PS center} \end{aligned} \quad (3.1)$$

$$W_{ij}(\mathbf{n}) = s_{ij} \quad \forall 1 \leq i \leq M, \forall j \in \mathcal{L}, i \text{ an IS center}$$

$$T_{ij}(\mathbf{n}) = \theta_{ij} n_j \sum_{i=1}^M \theta_{ij} W_{ij}(\mathbf{n}) \quad \forall 1 \leq i \leq M, \forall j \in \mathcal{L}$$

$$\begin{aligned} L_{ij}(\mathbf{n}) &= T_{ij}(\mathbf{n}) W_{ij}(\mathbf{n}) & \forall 1 \leq i \leq M, \forall j \in \mathcal{L} \\ W_{ij}(\mathbf{n}) &= \frac{s_{ij} [1 + \sum_{k \in \mathcal{L}} L_{ik}(\mathbf{n})]}{1 - \sum_{k \in \mathcal{O}} \rho_{ik}} & \forall 1 \leq i \leq M, \forall j \in \mathcal{O}, i \text{ a FCFS} \\ & & \text{or PS center} \end{aligned} \quad (3.2)$$

$$W_{ij}(\mathbf{n}) = s_{ij} \quad \forall 1 \leq i \leq M, \forall j \in \mathcal{O}, i \text{ an IS center}$$

$$L_{ij}(\mathbf{n}) = W_{ij}(\mathbf{n}) T_{ij} \quad \forall 1 \leq i \leq M, \forall j \in \mathcal{O}$$

END LOOP

For $\mathbf{N} = (N_j | j \in \mathcal{L})$ the *REPEAT* loop in the above algorithm will be executed $\prod_{j \in \mathcal{L}} (N_j + 1)$ times. Utilizations for the closed chains can be calculated for any population of interest using $\rho_{ij}(\mathbf{n}) = T_{ij}(\mathbf{n}) s_{ij}$. The calculations for the open chains can be moved out of the *REPEAT* loop; the algorithm as stated is simpler to extend to the priority case.

4. MVA PRIORITY APPROXIMATIONS

4.1 The PR and HOL Priority Queues

Consider a high-priority arrival that finds a lower priority customer in service at the priority center. Two types of priority scheduling, preemptive and nonpreemptive scheduling, are investigated.

Under preemptive (PR) scheduling the high-priority customer is immediately admitted to service. Only when all higher priority customers have been served is the interrupted lower priority customer returned into service. If we assume that the service time distribution of the lower priority customer is exponential, it does not matter whether the service of the lower priority customer is resumed from its point of interruption or whether its service is restarted with a randomly selected service interval. Preemption is assumed to incur no overhead and the tie breaking rule within each priority class is first come, first served (FCFS).

Under nonpreemptive or head-of-line (HOL) scheduling the lower priority customer in service is allowed to complete before the high-priority customer goes into service. Any lower priority customer in the queue is admitted to service only after all the higher priority customers' service is completed.

The MVA approximations for queuing networks with priority centers are developed in this section.

4.2 Average Wait Time at a Priority Center

Consider an M/M/1 PR queue subject to J Poisson arrival streams with parameters $\lambda_1 \cdots \lambda_J$. Let $s_j (= 1/\mu_j)$ denote the mean service time requirement of chain

j customers. Consider the arrival of a class j customer (the tagged customer) to the priority center. Upon arrival at the priority center the tagged class j customer will have to wait for the completion of service of all the higher priority customers (including his own class) that are already in the queue. In addition, there is a delay due to the service of higher priority customers of class $k < j$ which arrive after the tagged customer and which complete ahead of the tagged customer. If the average (total) delay of a class j customer is W_j , then the average number of class k customers that arrive after the tagged customer is $\lambda_k W_j$, and the delay these additional arrivals introduce is $\lambda_k s_k W_j$. A final delay of s_j is required to complete the tagged customer's own service requirement. Thus the average waiting time spent by the tagged customer at a preemptive priority center is given by [10, 15]

$$W_j - s_j + \sum_{k=1}^j L_k s_k + \sum_{k=1}^{j-1} W_j \lambda_k s_k. \tag{4.1}$$

Solving for W_j :

$$W_j = \frac{s_j + \sum_{k=1}^j L_k s_k}{1 - \sum_{k=1}^{j-1} \rho_k}, \tag{4.2}$$

where $\rho_k = \lambda_k s_k$ is the chain k utilization and L_k the class k queue length (including the customer in service).

For the M/M/1 HOL queue, once the tagged customer has begun service, it is allowed to complete uninterrupted. The additional delay due to arrivals of class $k < j$ is given by $\lambda_k s_k (W_j - s_j)$, since additional arrivals during the service time s_j do not increase the tagged customer's waiting time. Also, the tagged customer experiences an initial delay due to the service completion of the customer in service. Thus, the average wait time at a HOL center is given by

$$W_j = s_j + \sum_{k=1}^j (L_k - \rho_k) s_k + \sum_{k=1}^j \rho_k s_k + \sum_{k=1}^{j-1} (W_j - s_j) \lambda_k s_k. \tag{4.3}$$

Solving for W_j :

$$W_j = s_j + \frac{\sum_{k=1}^j L_k s_k + \sum_{k=j+1}^j \rho_k s_k}{1 - \sum_{k=1}^{j-1} \rho_k}. \tag{4.4}$$

4.3 MVA PR and HOL Priority Approximations

Equations (4.1)–(4.4) provide the exact value for W_j for M/M/1 preemptive and nonpreemptive queuing systems [10, 15]. However, the arrival process to a center in a queuing network is, in general, not Poisson, so these equations do not provide an exact solution for a priority center in a queuing network. In addition, these equations allow class-dependent service time distributions (parameter s_j), whereas product-form networks require the service time distributions at FCFS centers to be class independent.

Nevertheless, our approximations for priority networks are based upon eqs. (4.1)–(4.4). To derive the approximation, let $L_{ik}(\mathbf{N})$ be the average number of class k customers at center i seen by an arriving class j customer, given that the network population is \mathbf{N} . Then eq. (4.2) can be generalized to apply to a queuing

network as follows:

$$W_{ij}(\mathbf{N}) = \frac{s_{ij} + \sum_{k=1}^j L_{ik}(\mathbf{N})s_{ik}}{1 - \sum_{k=1}^{j-1} \rho_{ik}}. \quad (4.5)$$

To calculate the value of $L_{ik}(\mathbf{N})$, we assume that the Arrival Theorem [14, 25] applies. Under this assumption $L_{ik}(\mathbf{N}) = L_{ik}(\mathbf{N})$ if $j \in \mathcal{O}$ and $L_{ik}(\mathbf{N}) = L_{ik}(\mathbf{N} - \mathbf{1}_j)$ if $j \in \mathcal{L}$. In the case where $j \in \mathcal{O}$, the fact that $L_{ij}(\mathbf{N}) = \Delta_j \theta_{ij} W_{ij}(\mathbf{N})$ can be used to eliminate $L_{ij}(\mathbf{N})$ from the right-hand side of (4.5), since $L_{ij}(\mathbf{N})$ is unknown at this point in the calculation. The result of these substitutions is

$$W_{ij}(\mathbf{N}) = \begin{cases} \frac{s_{ij} + \sum_{k=1}^j L_{ik}(\mathbf{N} - \mathbf{1}_j)s_{ik}}{1 - \sum_{k=1}^{j-1} \rho_{ik}}, & j \in \mathcal{L}, \\ \frac{s_{ij} + \sum_{k=1}^{j-1} L_{ik}(\mathbf{N})s_{ik}}{1 - \sum_{k=1}^j \rho_{ik}}, & j \in \mathcal{O}. \end{cases} \quad (4.6)$$

Before this equation can be used, values for ρ_{ik} must be determined. Now if $k \in \mathcal{O}$, then $\rho_{ik} = T_{ik}s_{ik}$ and thus is independent of the closed network population. However, for $k \in \mathcal{L}$, ρ_{ik} depends on the closed network population, and it is not obvious which value of $\rho_{ik}(\mathbf{n})$, $\mathbf{0} < \mathbf{n} \leq \mathbf{N}$ should be used.

Several approaches to this problem have been tried. Teunissen [private communication, Nov. 1982] suggested using $\rho_{ik} = \rho_{ik}(\mathbf{N} - \mathbf{1}_j)$. However, experience has shown that this approximation has unacceptable accuracy. Bryant, Krzesinski, and Teunissen [4] studied the approximation where $\rho_{ik} = \rho_{ik}(\mathbf{N})$. This approximation works well except at priority node utilizations larger than about 0.7. Finally, Chandy and Lakshmi [7] proposed using $\rho_{ik} = \rho_{ik}(\mathbf{N} - \mathbf{L}_{ik})$, where $\mathbf{L}_{ik} = L_{ik}(\mathbf{N})\mathbf{1}_k$ is the average queue length of class k customers at center i when the population is \mathbf{N} . Here $(\mathbf{N} - \mathbf{L}_{ik})$ represents a closed chain population with L_{ik} fewer class k customers. (When $L_{ik}(\mathbf{N})$ is not an integer, linear interpolation can be used to estimate an appropriate value for $\rho_{ik}(\mathbf{N} - \mathbf{L}_{ik})$). The rationale is that, when there are \mathbf{L}_{ik} customers already at center i , the arrival rate of class k customers at center i is determined by the remaining $\mathbf{N} - \mathbf{L}_{ik}$ customers in the rest of the network. Under this assumption, $\rho_{ik} = T_{ik}(\mathbf{N} - \mathbf{L}_{ik})s_{ik}$, but this is exactly the definition of $\rho_{ik}(\mathbf{N} - \mathbf{L}_{ik})$. Empirical evidence [4], similar to that reported below, indicates that this approximation has the best overall accuracy.

Thus, the mean waiting time of a class j customer at a preemptive priority center i , when the closed chain population is \mathbf{N} is given by

$$\text{PR: } W_{ij}(\mathbf{N}) = \begin{cases} \frac{s_{ij} + \sum_{k=1}^j L_{ik}(\mathbf{N} - \mathbf{1}_j)s_{ik}}{1 - \sum_{k=1}^{j-1} \rho'_{ik}}, & j \in \mathcal{L}, & (4.7a) \\ \frac{s_{ij} + \sum_{k=1}^{j-1} L_{ik}(\mathbf{N})s_{ik}}{1 - \sum_{k=1}^j \rho'_{ik}}, & j \in \mathcal{O}, & (4.7b) \end{cases}$$

where

$$\rho'_{ik} = \begin{cases} \rho_{ik}(\mathbf{N} - \mathbf{L}_{ik}), & k \in \mathcal{L} \\ \rho_{ik}, & k \in \mathcal{O} \end{cases} \quad \text{and} \quad \mathbf{L}_{ik} = L_{ik}(\mathbf{N})\mathbf{1}_k.$$

Note that (4.7) requires values of $\rho_{ik}(\mathbf{n})$ for $\mathbf{n} < \mathbf{N}$. This is not a problem for direct implementations of MVA since performance statistics for all intermediate populations are calculated. For large networks direct MVA becomes impractical, and instead one must use fast approximate MVA methods such as the Schweitzer [23] or Linearizer [8] methods. These methods have the characteristic that they only calculate performance statistics for network populations in the vicinity of the target population \mathbf{N} . The Bryant, Krzesinski, and Teunissen algorithm can be used in this case instead, but this algorithm is known to have unacceptable accuracy at high utilizations. It remains an open problem to develop MVA priority approximations that can be used in the Schweitzer or Linearizer methods and that also have acceptable accuracy when the utilization of the priority center is high.

The mean waiting time at a HOL priority center, when the closed chain population is \mathbf{N} , is given by

$$\text{HOL: } W_{ij}(\mathbf{N}) = \begin{cases} s_{ij} + \frac{\sum_{k=1}^j L_{ik}(\mathbf{N} - \mathbf{1}_j) s_{ik} + \sum_{k=j+1}^J \rho_{ik}(\mathbf{N} - \mathbf{1}_j) s_{ik}}{1 - \sum_{k=1}^{j-1} \rho'_{ik}}, & j \in \mathcal{L} \quad (4.8a) \\ \frac{s_{ij}(1 - \sum_{k=1}^{j-1} \rho'_{ik}) + \sum_{k=1}^{j-1} L_{ik}(\mathbf{N}) s_{ik} + \sum_{k=j+1}^J \rho_{ik}(\mathbf{N}) s_{ik}}{1 - \sum_{k=1}^j \rho'_{ik}}, & j \in \mathcal{O}. \quad (4.8b) \end{cases}$$

Equations (4.7a) and (4.8a) are used in place of eq. (3.1), and eqs. (4.7b) and (4.8b) are used in place of eq. (3.2) of the MVA algorithm when the center under consideration is a PR or HOL priority center, respectively. The use of these equations does not significantly add to the computational requirements of the original MVA algorithm. Unlike the algorithm of Section 3, waiting times and queue lengths for the open classes must be calculated at intermediate closed-chain populations \mathbf{n} . Thus, in the priority network case, one cannot remove the open-chain calculations from the main *REPEAT* loop of the MVA recursion. Additional storage is required to keep values of $\rho_{ik}(\mathbf{n})$, $1 \leq j \leq J$, $W_{ij}(\mathbf{n})$, and $L_{ij}(\mathbf{n})$, $j \in \mathcal{O}$, $\mathbf{0} \leq \mathbf{n} \leq \mathbf{N}$. Provided that performance measures are calculated on a class-by-class basis (from the highest to the lowest priority), evaluation of eqs. (4.7) and (4.8) is direct, that is, no iteration is required.

It should be emphasized that the Arrival Theorem *does not apply* to queuing networks containing priority centers. For example, in a closed queuing network, it is clear that whenever a low-priority customer leaves a PR center, then all of the higher priority customers must be located at centers other than that PR center. Therefore, the arrival theorem *cannot apply* at those other centers for customers of lower priority classes.

In the next section a study of the accuracy of the approximation based on eqs. (4.7) and (4.8) is presented.

5. ERROR ANALYSIS

This section presents an investigation of the accuracy of the MVA approximation technique for queuing networks with priority centers. Additionally, for closed

networks, accuracy of the MVA-PR approximation is compared to Sevcik's Shadow approximation.

For closed networks, accuracy of the approximations is first evaluated by comparing exact and approximate solutions for two-class, two-center networks consisting of a single priority center and one nonpriority center. Owing to the limited number of parameters in these small networks, an exhaustive exploration of the relation between the accuracy of the approximation and network parameters (such as population, utilization of the priority center) is possible. This allows us to draw definitive conclusions as to when the MVA priority approximations can be used with confidence.

Conclusions from the study of the two-center networks are extended to larger networks by comparing the MVA approximation to simulations of a selected group of test networks. Whereas for the two-center case an exhaustive exploration of the parameter space is feasible, this cannot be done for the larger networks. Instead, we must be satisfied with test networks whose parameter values are representative of real systems.

For open and mixed networks, the exact solutions are difficult to obtain, and only comparisons to simulation results are presented. Once again we must be satisfied by comparisons with a few "representative" networks.

5.1 Error Measures

The error analysis is presented in terms of *tolerance error* [8] on selected performance measures. Let $E_e(x_{ij})$ and $E_a(x_{ij})$ denote the expected values of x_{ij} as calculated by the exact global balance technique and by the priority approximation respectively. The tolerance error $\Delta(x_{ij})$ in x_{ij} is given by

$$\Delta(x_{ij}) = \frac{|E_e(x_{ij}) - E_a(x_{ij})|}{\sum_{i=1}^M E_e(x_{ij})}.$$

The tolerance error $\Delta(x_{ij})$ express the error in $E_a(x_{ij})$ as a fraction of the total class j performance measure in the entire network. Unlike the relative error, the tolerance error will not emphasize a large error in a numerically small (and thus less significant) component of a performance measure. For example, if the class 1 customer population is large, an error in the mean queue length of one customer at a server with a true mean queue length of less than one customer could result in large relative errors, but from the standpoint of overall network performance measures, such an error is not significant.

Tolerance errors for utilizations, mean queue lengths, and waiting times were calculated for each of the test networks and each approximation method. To conserve space, the accuracy metric used in this section is the *maximum tolerance error* (the maximum of the tolerance error of any of the indicated performance measures in the test network).

5.2 Accuracy Comparison for Two-Center Closed Networks

Accuracy comparisons for two-class, two-center networks consisting of a single priority center and one nonpriority center were conducted by comparing the exact solution (obtained via global balance) to the MVA and Shadow approximations.

In the test networks, the nonpriority center is intended to represent an aggregate of $(M - 1)$ BCMP centers connected to a single priority center. The problem of determining a state-dependent service for the aggregate center equivalent (or approximate) to $(M - 1)$ nonlocally balanced centers is a separate study unrelated to the accuracy of the approximations and is not discussed here.

Convenient choices for the service discipline at the nonpriority center are Processor Sharing (PS) and Infinite Server (IS). These two choices represent opposite extremes in the following sense: In the PS case, competition between class 1 and 2 customers at the nonpriority center is severe, and the arrival process of each class of customer at the priority center is influenced by the loading at the nonpriority center. In the IS case, competition between the customer classes at the nonpriority center is nonexistent and resource contention is concentrated at the priority center. The true situation in most queuing network models of computer systems lies between these two extremes. If we assume that the priority center represents the CPU in the system, then the nonpriority center would represent the I/O subsystem. Typically, I/O subsystems have substantial independent processing power and can be serving several requests simultaneously. Also, the devices in the I/O subsystem are normally loaded so that no one device becomes a system bottleneck. This suggests that an IS discipline is more appropriate than a PS discipline for the nonpriority center in the test networks. Results for both types of test networks are presented in this paper. However, as outlined above, it is felt that the IS cases are more representative of the errors that one would observe in practice.

Three types of comparisons were performed using these test networks. Each type of comparison involved generating a collection of test networks with known properties. These test networks were generated by varying the service times at center 1 (the priority node) until the network had the desired properties according to the exact solution.

5.3 Error Contour Diagrams

The first set of comparisons examines the accuracy of the approximations as a function of the utilizations ρ_{11} and ρ_{12} , which the customer population was held fixed at $\mathbf{N} = (10, 10)$. The service times at the nonpriority node were equal. (Since the networks are designed to have predetermined utilizations at the priority center, only the relative values of s_{11} and s_{12} are significant.)

The results of this comparison are presented as contour plots generated as follows. Let (x, y) denote a point in the unit square $0 \leq x \leq 1.0$ and $0 \leq y \leq 1.0$. For each such point, generate a test network with $\rho_1 = y$ being the total utilization of the priority center, $\rho_{11} = xy$ being the class 1 utilization of the priority center, and $\rho_{12} = (1 - x)y$ being the class 2 utilization of the priority center. Thus, y specifies the total utilization at the priority center, whereas x specifies the fraction of that load that is due to the high-priority class.

Let $\Delta(x, y)$ denote a tolerance error on a selected performance measure for the network generated as described above. Therefore, $\Delta(x, y)$ defines a surface above the unit rectangle that represents the error versus the utilizations specified by x and y . One way to display this surface in a two-dimensional plot is to plot the intersection of the surface with a plane of fixed height above the (x, y) plane.

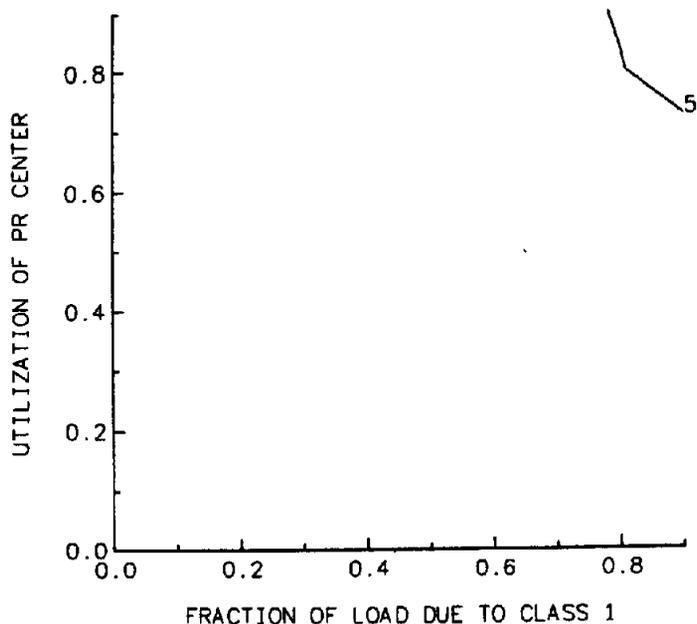


Fig. 1. Maximum tolerance error in PR-IS case using MVA approximation.

Such diagrams are commonly called contour plots and give the projection of this intersection onto the (x, y) plane.

A contour diagram thus consists of a set of contours for several (equally spaced) values of $\Delta(x, y)$. The contour diagram reveals where the priority approximation yields good results according to error measure Δ . The spacing between the contour lines indicates where the approximation gradually becomes worse (contour lines widely spaced) and where the approximation rapidly deteriorates (contour lines closely spaced).

For the contour plots presented here, the minimum contour is 5 percent tolerance error; the contour interval is 5 percent up to the 50 percent contour and 10 percent thereafter. Each contour diagram is based on the solutions for 81 queuing networks generated as described above with $x = 0.1, 0.2, \dots, 0.9$, and $y = 0.1, 0.2, \dots, 0.9$.

5.3.1 *Summary of Results.* Figures 1–6 give error contour diagrams of the maximum tolerance error.

The difference between the accuracies of the MVA and Shadow approximation methods in the PR-IS case (Figures 1 and 3) is striking. Although both methods calculate performance statistics with errors of less than 5 percent for a large range of parameter values, the maximum tolerance error observed for the MVA method is less than 10 percent, while the Shadow method has tolerance errors larger than 50 percent. Similar diagrams for the other error measures show that the maximum error for the MVA approximation typically occurs in W_{12} , whereas the maximum error for the Shadow approximation occurs at the IS center.

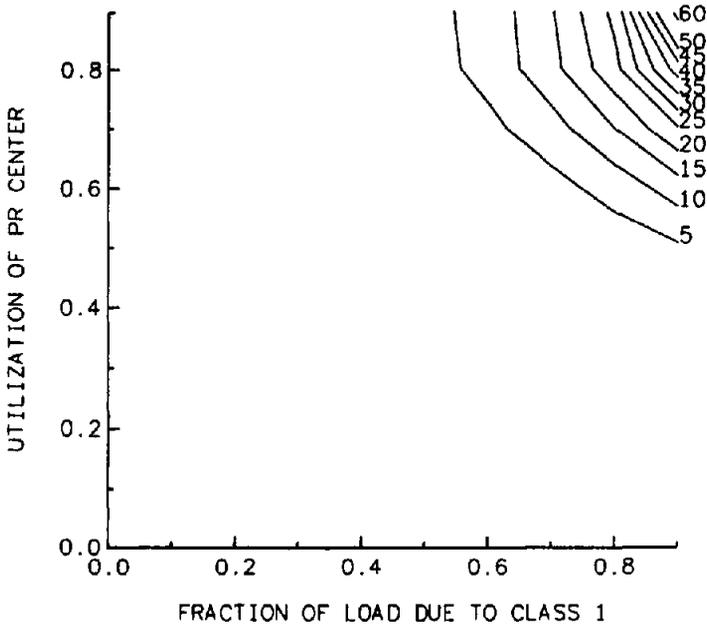


Fig. 2. Maximum tolerance error in PR-PS case using MVA approximation.

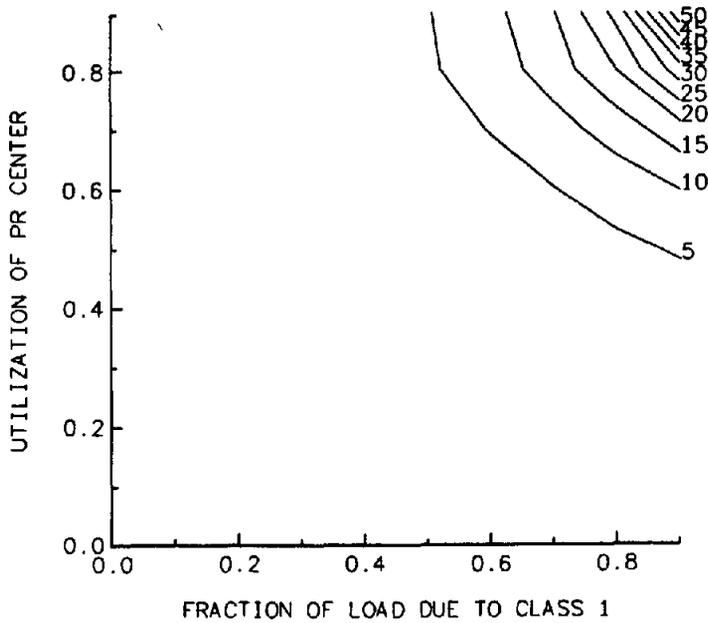


Fig. 3. Maximum tolerance error in PR-IS case using shadow approximation.

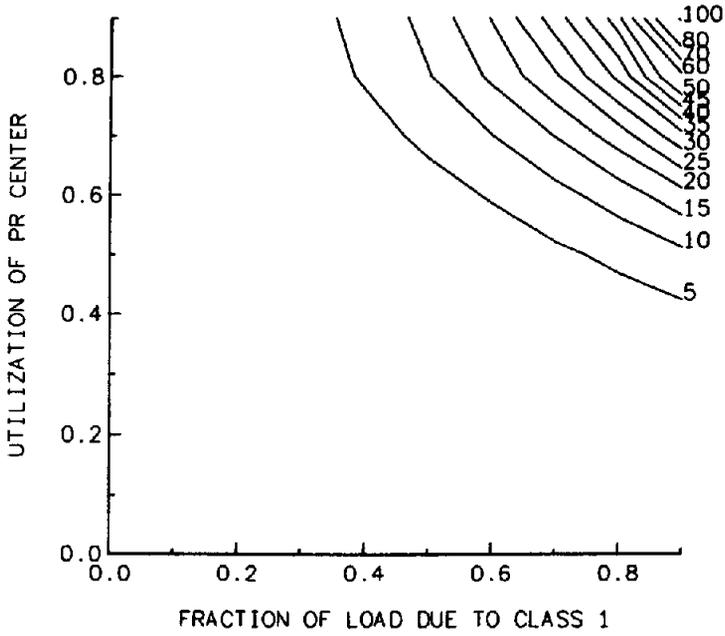


Fig. 4. Maximum tolerance error in PR-PS case using shadow approximation.

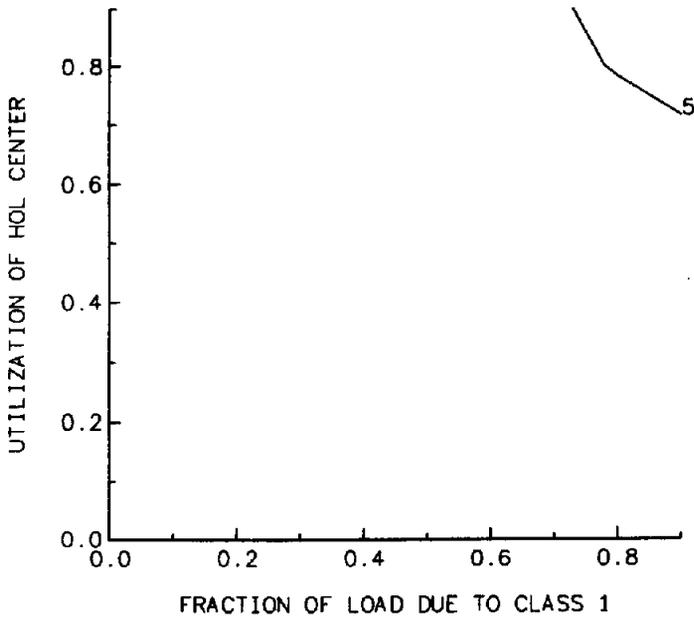


Fig. 5. Maximum tolerance error in HOL-IS case using MVA approximation.

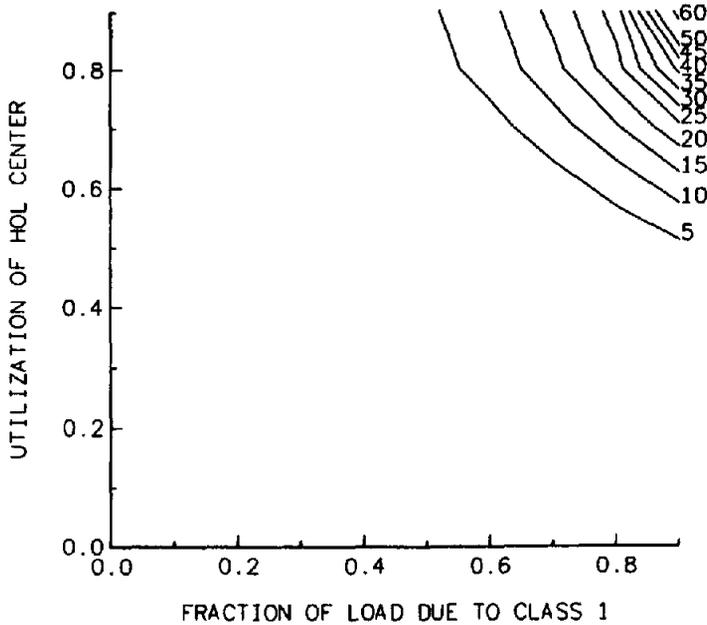


Fig. 6. Maximum tolerance error in HOL-PS case using MVA approximation.

Since the errors are larger in the upper right corner of the contour plots, it follows that the approximations are more sensitive to high utilizations when most of the load is due to the high-priority class. For the MVA approximations this suggests that the error occurs because the utilizations in the denominators of eqs. (4.7) or (4.8) are slightly in error. When these utilizations total nearly 1.0, small errors in estimating these quantities causes large errors in estimating W_{ij} and other performance measures. It follows that for more than two customer classes, the tolerance error will increase as the customer index increases (the class decreases in priority). See also the discussion in Section 5.6.

Figures 2 and 4 contain the error contours for the MVA and Shadow approximations for the PR-PS case. As can be seen from these figures, the MVA approximation has better accuracy than the Shadow approximation (both in terms of area outside of the 5 percent contour and maximum error encountered), but both approximations are less accurate in this case than they were in PR-IS case. For these cases, the maximum error always occurred at the PS center. This can be attributed to the violation of the Arrival Theorem hypothesis at the PS center. As a general rule, we would expect the MVA approximation to be less accurate when there is heavy interclass competition for resources away from the priority center, since this approximates the PR-PS network case.

Figures 5 and 6 give the error contours for the HOL-IS and HOL-PS cases. The results in these cases are similar to those observed for the PR-IS and PR-PS cases.

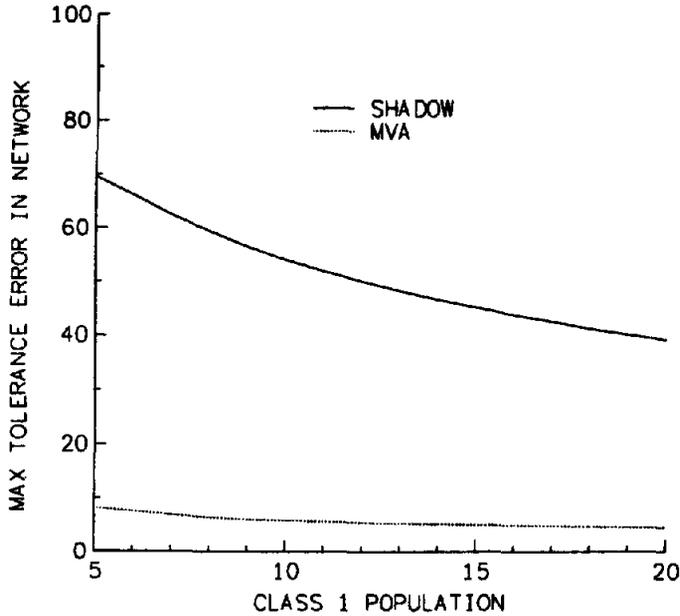


Fig. 7. PR-IS case.

5.4 Error Versus Class 1 Population

The second set of accuracy comparisons for closed networks examines the relationship of the tolerance error to the class 1 population. Figures 7–9 present the maximum tolerance error as a function of the number N_1 of high-priority customers in the network. The number N_2 of low-priority customers in the network is kept fixed at $N_2 = 10$.

The purpose of this set of figures is to determine how the accuracy of the approximations changes with increasing customer population. However, the contour plots of the last section show that accuracy of the approximations decreases as the utilization of the priority server increases (especially when the total utilization of the priority server exceeds 0.90). Since increasing the number of customers in the network increases the utilization of the servers, service times in the test networks were adjusted to keep the priority center's utilization constant. In this way, the effect of changing the customer population can be observed without introducing additional errors due to changes in server utilization.

In each of the graphs of Figs. 7–9, the total utilization ρ_1 at the priority center and the fraction of load due to class 1 at that center (ρ_{11}/ρ_1) is kept constant as the class 1 population varies by assigning appropriate chosen values to the mean service times s_{11} and s_{12} . To conserve space, the only graphs presented here are for the case $\rho_1 = 0.90$ and $\rho_{11}/\rho_1 = 0.90$. Errors for smaller values of ρ_1 and ρ_{11}/ρ_1 would be correspondingly less, as indicated by the results of the last section.

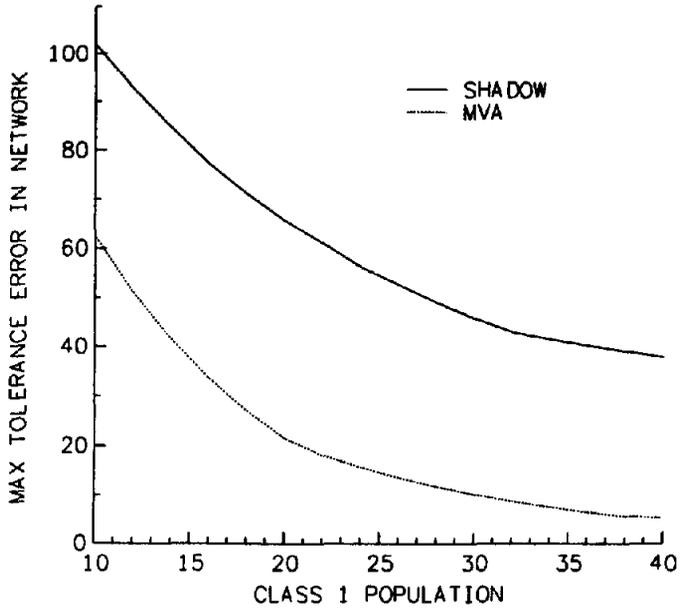


Fig. 8. PR-PS case.

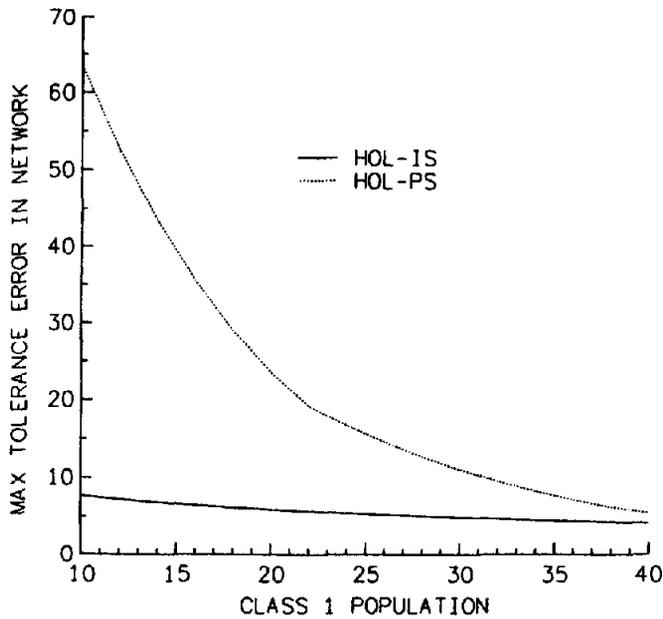


Fig. 9. HOL case.

5.4.1 *Summary of Results.* Figures 7–9 present graphs of maximum tolerance error versus number of high-priority customers. As before, for the cases in which the nonpriority center used an IS discipline, the low-priority waiting times W_{12} at the center were observed to give rise to the maximum errors in the network. When the nonpriority center used a PS discipline, the maximum errors occurred at the PS node.

Figures 7–9 show that for each approximation method the tolerance error decreases with increasing class 1 population. This can be explained by recalling that the test networks were generated so that each network had the same priority center utilization (load) regardless of the class 1 population. As the population increases, this effectively means that the same load is distributed among more and more customers. The methods are therefore more accurate when the frequency of interruptions due to higher priority arrivals is high and the duration of service per interruption is small.

For very small class 1 populations ($N_1 < 5$), both of the approximations have large errors. Thus, neither approximation can be recommended when the load due to class 1 is generated by one or very few customers with large service time requirements. Since one is normally interested in applying these solution techniques to networks consisting of dozens of customers, this does not appear to be a significant limitation of the method.

Figures 7 and 8 show that the MVA approximation has considerably better accuracy than the Shadow approximation, especially when there is little contention away from the priority center (i.e., in the PR-IS and HOL-IS cases). Tolerance errors for the MVA approximation are less than 10 percent for all reasonable customer populations. In the PR-PS case (Fig. 8), more than 30 class 1 customers are required in order to obtain this accuracy. Finally, as shown in Fig. 9, accuracy of the MVA approximation in the HOL case is similar to that observed for the PR cases.

5.5 Effect of Service Time Ratio on Accuracy

In this set of accuracy comparisons, the effect of the remaining parameter, $R = s_{21}/s_{22}$, is considered. In both of the previous cases, $R = 1$. Since the test networks were designed to have specified utilizations at the priority node, only R , and not the absolute magnitudes of s_{21} and s_{22} , is significant. It might be more appropriate to study the accuracy of the approximations versus the ratio s_{11}/s_{12} . However, this turns out to be an overconstrained problem and test networks meeting the rest of the design goals cannot be constructed.

Figures 10–12 give the relationship between the maximum tolerance error and R , the service time ratio at the nonpriority center. In each of these figures, the total priority center utilization was 0.90 and the fraction of this load that was due to class 1 was 0.90. There were 10 customers in each class for this set of comparisons. Accuracy of the approximations would be better than indicated here when either the priority center utilization or the class 1 fraction of the load were smaller, or the number of customers in the network was larger.

Examination of the figures indicates that the MVA approximation has good accuracy provided $R < 1.0$ and there is no contention away from the priority node (PR-IS and HOL-IS cases). If there is significant contention away from

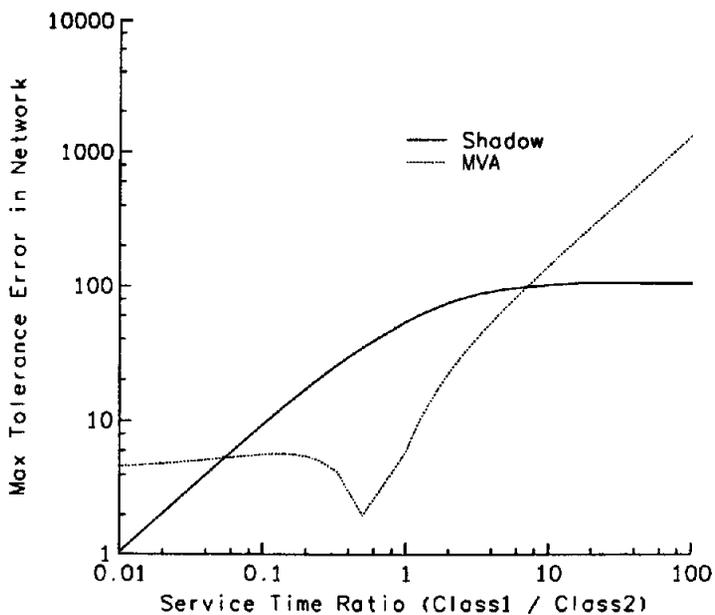


Fig. 10. PR-IS case.

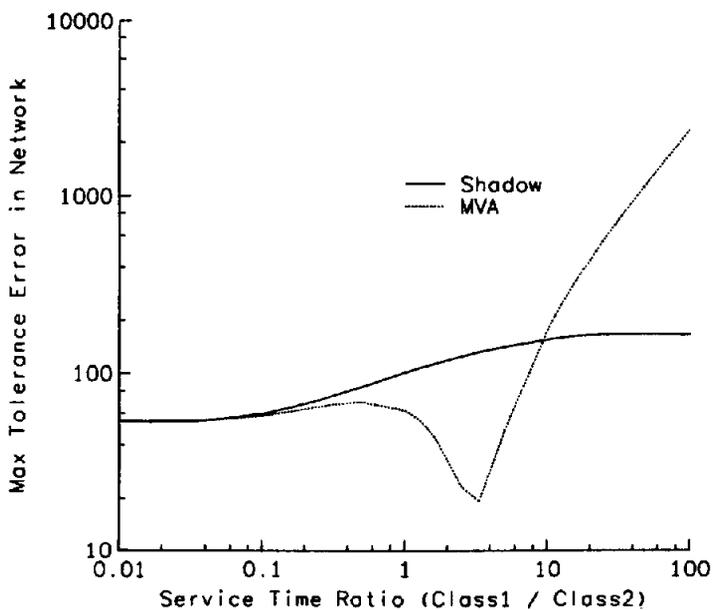


Fig. 11. PR-PS case.

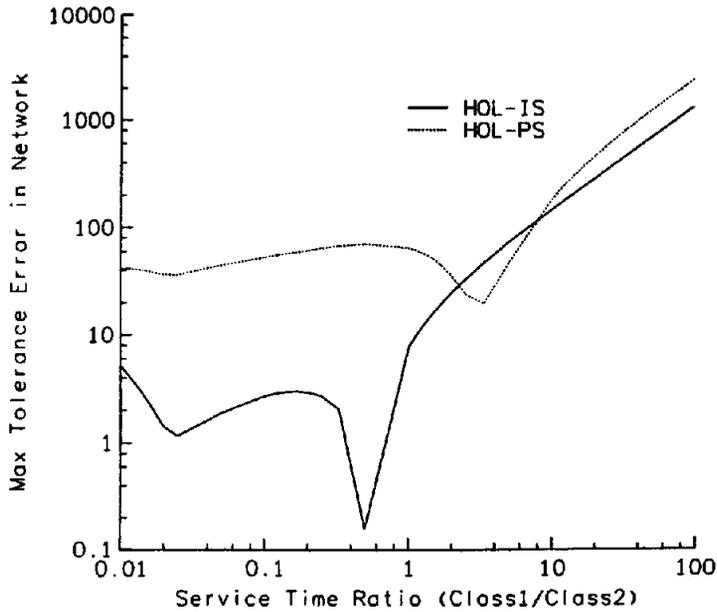


Fig. 12. HOL cases.

the priority node (PR-PS and HOL-PS cases), the MVA approximation has acceptable accuracy only when $R \sim 1.0$. In the PR-IS case, for $R < 0.6$, the Shadow approximation has better accuracy than the MVA approximation. In all cases, for sufficiently large values of R , the tolerance errors can become arbitrarily large.

5.6 Studies of Larger Closed Networks

In previous sections, accuracy of the MVA priority approximation was examined for networks consisting of only two service centers and two customer classes. In this section, accuracy of the approximation is examined for larger test networks.

Many of these test networks were too large to be solved exactly and hence simulation was used instead. Also, the number of parameters involved made exhaustive comparisons like those of the previous sections infeasible. Instead, the approach taken here was to randomly generate a collection of networks each of which had a "reasonable" set of parameter values. ("Reasonable" in this context is taken to mean parameter values that one might encounter in practice.) This collection of networks was then solved by simulation and by the MVA approximation; the two solutions were compared to estimate the size of the error in the approximation.

The results of these comparisons are summarized in Tables I and II. These tables give the percentage of test cases for which the tolerance error in the mean wait time at the priority center was in the indicated range. Assuming that the parameters of the test networks are representative of those encountered in practice, these tables provide an indication of how often errors in a given range might be expected to occur.

Table I. Frequency Distribution of Tolerance Error: Results from Large Closed Networks with Two Customer Classes

Tolerance error (percent)	Frequency ^a	
	Class 1	Class 2
0-3	100	78
4-7		7
8-10		6
>10		9

^a Percent of test cases (in 32 trials) where the maximum tolerance error in the mean wait time for the indicated class fell within the indicated range.

Table II. Frequency Distribution of Tolerance Error: Results from Large Closed Networks with Three Customer Classes

Tolerance error (percent)	Frequency ^a		
	Class 1	Class 2	Class 3
0-3	100	75	68
4-7		8	7
8-10		0	5
>10		17	20

^a Percent of test cases (in 59 trials) where the maximum tolerance error in the mean wait time for the indicated class fell within the indicated range.

In Tables I and II the test networks consisted of closed networks with two to five service centers and two or three customer classes. Some of the test networks were of central server model type, while the others had general topology. There were one or two preemptive priority centers in each network, and the service discipline at the other centers were PS, FCFS, or IS. (Previous experiments had shown that the accuracy of the HOL approximation was similar to that of the PR approximation; thus it was felt unnecessary to run the simulations for the HOL case.)

A total of 91 test cases (32 with two-customer classes and 59 with three-customer classes) were constructed. The test network parameters were chosen as follows: (1) the service times at the priority center were class dependent and the mean value ranged from 2 to 150 ms; (2) the ratio of priority center service time to nonpriority center service times ranged from 2 to 25; (3) the service times at the nonpriority centers were class independent; (4) the number of customers in each class was between 1 and 15. These parameters resulted in priority center utilizations from 0.2 to 0.92.

Tables I and II show that the accuracy of the MVA priority approximation for larger networks is also very encouraging. At least 80 percent of the test cases had less than 10 percent tolerance error. The source of larger tolerance errors in

other test cases can be attributed to one or more of the following characteristics of the test networks:

- the priority center utilization exceeds 0.9;
- the priority center utilization due to higher priority classes is close to 0.9;
- a nonpriority center utilization is close to 1.0.

In general, the wait time estimates for class 1 were always the most accurate. As before, errors in the waiting times for the class 2 customers are larger than the errors in the waiting times of the class 1 customers. Table II shows that in the three-class networks, errors in the waiting times for class 3 are larger than errors in the higher priority classes. Although networks with more customer classes were not considered, it seems likely that for J customer classes, the largest errors will be associated with class J , and the smallest errors will be associated with class 1.

Overall, the limited experimentation with larger networks provided striking confirmation of the conclusions made by the studies involving two-center networks in Sections 5.3 and 5.4.

5.7 Open and Mixed Networks

In this section, the accuracy of the MVA-priority algorithm in solving open and mixed networks is considered. The test cases were either two-center (loop) networks or central server networks with six service centers. These networks consisted of a single preemptive priority center and one or more nonpriority centers. (Since accuracy of the approximation for the PR and HOL cases was seen to be very similar for the closed, two-center test cases, HOL priorities were not considered here.) In the two-center networks, the nonpriority center was an IS; in the central server networks the nonpriority centers were FCFS servers. The priority center can be thought of as the CPU, the IS center may represent the I/O subsystem, and the FCFS centers may represent the disks with associated channels.

Customers belonging to open classes arrive from an external source, circulate between the CPU and I/O for five times, and then leave the network. Fifty two test cases with open networks and 32 test cases with mixed networks were performed. These test networks had three customer classes. The mixed networks consisted of at least one open and one closed class. In half the test cases, the highest priority class was open and in the other half it was closed. The other customer classes were arbitrarily chosen to be open or closed. The service time distributions were assumed to be exponential and class dependent at the CPU. The external arrival rate for the open classes and the network population for the closed classes were varied such that the priority center utilization ranged from 0.2 to 0.93 over all the test cases.

The solutions were compared against simulation results. Again, the largest errors were found to occur in the wait time estimates. The frequency distributions of the tolerance error in mean wait time at the priority center are presented in Tables III and IV. For the cases considered, it can be observed that the wait time estimates for the highest priority class are very accurate, and for other classes

Table III. Frequency Distribution of Tolerance Error: Results from Open Networks

Tolerance error (percent)	Frequency ^a		
	Class 1	Class 2	Class 3
0-3	100	96	87
4-7		2	10
8-10		2	0
>10			3

^a Percent of test cases (in 52 trials) where the maximum tolerance error in the mean wait time for the indicated class fell within the indicated range.

Table IV. Frequency Distribution of Tolerance Error: Results from Mixed Networks

Tolerance error (percent)	Frequency ^a		
	Class 1	Class 2	Class 3
0-3	100	72	73
4-7		13	9
8-10		0	6
>10		15	12

^a Percent of test cases (in 32 trials) where the maximum tolerance error in the mean wait time for the indicated class fell within the indicated range.

they are well within an acceptable limit. Errors greater than 10 occurred when the priority center utilization exceeded 0.9. We emphasize that the network parameters were chosen to be representative of real systems.

6. CONCLUDING REMARKS

This paper presents an MVA-based algorithm for computing approximate performance measures of closed, open, and mixed queuing networks containing preemptive and nonpreemptive priority centers. The approximation has lower computational complexity and appears to have better accuracy than previously published algorithms. The MVA approximation for priority networks is easy to install in existing MVA solution packages, and can be used when there are multiple priority centers in the network.

Accuracy of the approximation for two-center, two-class networks was evaluated by comparing exact and approximate solutions for a set of carefully constructed test networks. This evaluation demonstrated that the accuracy of the approximation increased with increasing customer population, while the priority center utilization was held constant. For fixed populations, the accuracy of the approximation was seen to be good except at high utilizations of the priority center and when most of the load at that center was due to the high-priority class. In the case of PR-IS networks with populations of more than 10 high-

priority and 10 low-priority customers, the algorithm calculates approximate performance measures with tolerance errors of less than 10 percent. When there was significant contention away from the priority server (PR-PS case), significantly larger populations were required to obtain the same accuracy. Results for the HOL cases were similar.

Accuracy of the approximation for larger closed networks, open networks, and mixed networks was established by comparing simulation results with the approximate solution. Provided that the network parameters were representative of real systems, the accuracy of the approximation was seen to be good in these cases as well.

ACKNOWLEDGMENTS

The global balance solver used to calculate the exact solutions was written by Bryan Rosenburg of the University of Wisconsin-Madison. This program was an essential part of the research reported here. Dinkar Sitaram, also of the University of Wisconsin-Madison, provided valuable assistance during the early phase of testing of the MVA algorithms reported here.

REFERENCES

1. AVI-ITZHAK, B., AND HEYMAN, D. Approximate queueing models for multiprogramming computer systems. *Oper. Res.* 21, 6 (Nov./Dec. 1973), 1212-1230.
2. BARD, Y. Some extensions of multiclass queueing network analysis. In *Proceedings of the 4th International Symposium on Modelling and Performance Evaluation of Computer Systems*. North Holland, Amsterdam 1979.
3. BASKETT, F., CHANDY, K., MUNTZ, R., AND PALACIOS, P. Open, closed, and mixed networks of queues with different classes of customers. *J. ACM* 22, 2 (Apr. 1975), 248-260.
4. BRYANT, R. M., KRZESINSKI, A. E., AND TEUNISSEN, P. The MVA preempt-resume priority approximation. In *Proceedings of the 1983 ACM Sigmetrics Conference* (Minneapolis, Minn., August 29-31). ACM, New York, 1983, pp. 12-27.
5. BUZEN, J. Computational algorithms for closed queueing networks with exponential servers. *Commun. ACM* 16, 9 (Sept. 1973), 527-531.
6. CHANDY, K., HERZOG, U., AND WOO, U. Parametric analysis of queueing networks. *IBM J. Res. Devel.* 19 (Jan. 1975), 36-42.
7. CHANDY, K., AND LAKSHMI, M. S. An approximation technique for queueing networks with preemptive priority queues. Tech. Rep. Dept. Computer Sciences, The Univ. of Texas at Austin, Austin, Tex., Feb. 1983.
8. CHANDY, K., AND NEUSE, D. Linearizer: A heuristic algorithm for queueing network models of computer systems. *Commun. ACM* 25, 2 (Feb. 1982), 126-134.
9. CHANDY, K., AND SAUER, C. Computational algorithms for product form queueing networks. *Commun. ACM* 23, 10 (Oct. 1980), 573-583.
10. COBHAM, A. Priority assignment in waiting line problems. *Oper. Res.* 2 (Feb. 1954), 70-76.
11. GRAHAM, G. Ed. Special Issue: Queueing network models of computer system performance. *ACM Comput. Surv.* 10, 3 (Sept. 1978), 219-359.
12. KRITZINGER, P., VAN WYK, S., AND KRZESINSKI, A. A generalization of Norton's theorem for multiclass queueing networks. *Perform. Eval.* 2, 2 (July 1982), 98-107.
13. LAM, S., AND LIEN, Y. A tree convolution algorithm for the solution of queueing networks. Tech. Rep. TR-165. Dept. of Computer Sciences, Univ. of Texas, Austin, Tex., Jan. 1981.
14. LAVENBERG, S., AND REISER, M. Stationary state probabilities at arrival instants for closed queueing networks with multiple types of customers. *J. Appl. Prob.* 17, (1980), 1048-1061.
15. JAISWAL, N. *Priority Queues*. Academic Press, New York, 1965.

16. MORRIS, R. Priority queueing networks. *Bell Syst. Tech. J.* 60, 6 (Oct. 1981), 1745-1769.
17. NEUSE, D., AND CHANDY, K. M. HAM: The heuristic aggregation method for solving general closed network models of computer systems. *Perform. Eval. Rev.* 11, 4 (Winter 1982-1983), 195-212.
18. REISER, M., AND KOBAYASHI, H. Queueing networks with multiple closed chains: Theory and computational algorithms. *IBM J. Res. and Devel.* 19, 3 (May 1975), 283-294.
19. REISER, M. Interactive modeling of computer systems. *IBM Syst. J.* 15, 4 (1976), 309-327.
20. REISER, M. A queueing network analysis of computer communication networks with window flow control. *IEEE Trans. Commun. COM-27*, 8 (Aug. 1979), 1199-1209.
21. REISER, M., AND LAVENBERG, S. Mean value analysis of closed multichain queueing networks. *J. ACM* 27, 2 (Apr. 1980), 313-322.
22. SAUER, C., AND CHANDY, K. Approximate analysis of central server models. *IBM J. Res. Devel.* 19 (May 1975), 301-313.
23. SCHWEITZER, P. Approximate analysis of multiclass closed networks of queues. In *International Conference on Stochastic Control and Optimization*, (Amsterdam), 1979.
24. SEVCIK, K. Priority scheduling disciplines in queueing network models of computer systems. In *Proceedings of the IFIP Congress 77* (Toronto, Aug 8-12), North Holland, Amsterdam, 1977, pp. 565-570.
25. SEVCIK, K., AND MITRANI, I. The distribution of queueing network states at input and output instants. *J. ACM* 28, 2 (Apr. 1981), 358-371.
26. ZAHORJAN, J., AND WONG, E. A solution of separable queueing network models using mean value analysis. *Perform. Eval. Rev.* 10, 3 (Fall 1981), 80-85.

Received June 1983; revised July 1984; accepted July 1984