Approximate Analysis of Load Dependent General Queueing Networks

I. F. AKYILDIZ, MEMBER, IEEE, AND ALBRECHT SIEBER

Abstract—**A method for approximate solutions to load dependent closed queueing networks containing general service time distributions and FCFS scheduling disciplines is presented. The technique demonstrated is an extension of the well-known method of Marie. A new formula for the conditional throughputs is derived. After each iteration a check is performed to guarantee that the results obtained are within a tolerance level ϵ. These iterations are repeated whenever invalid results are detected. On the average, the solutions obtained vary by less than 5 percent from their respective exact and simulation results.**

Index Terms—**Conditional throughputs, performance analysis, performance measures, queueing networks.**

I. Introduction

Queueing networks have been used extensively in the modeling and analysis of computer systems and communication networks since the last two decades. The low computational cost and adequate accuracy of queueing network models in predicting the performance of computer systems has been generally established [12], [14], [32]. This is primarily due to their ability to model multiple independent resources and the sequential use of these resources by different jobs. The basic results of network queueing theory were presented by Jackson, Gordon and Newell, and Buzen [5], [15], [17]. They demonstrated that solutions to both open and closed queueing networks with exponentially distributed arrival and service times implementing a first-come-first-served queueing discipline have a *product form.*

A product form implies that all stations have equilibrium state probabilities consisting of factors representing the individual stations within the network. The resulting implication is that the individual stations behave as if they were separate queueing systems. Baskett, Chandy, Muntz, and Palacios [3] extended these results to obtain product form solutions for open, closed, and mixed queueing networks with multiple job classes, nonexponential service time distributions, and different queueing disciplines. Such queueing networks are known to have local balance [8], Markov implies Markov property [26], or the station balance property [8]. Networks containing four types of stations with queueing disciplines FCFS (first-come-first-served), PS (processor sharing), IS (infinite server), and LCFS-PR (last-come-first-served preemptive resume), have product form solution. The service times per visit in FCFS stations should have a negative exponential distribution with the same mean for all classes. The other queueing disciplines allow general service time distributions (with rational Laplace transforms).

Several extensions to the existing product form networks have been proposed in recent years. Three new station types with exponential service time distributions and the queueing disciplines: SIRO (service-in-random-order) [35], LBPS (last batch processor sharing where jobs are processed as per last-come-first-served, but the arrival time is determined for a batch of jobs, not individual jobs) [27] and WEIRDP (a portion p of the processor is allocated to the first job in the queue and the remainder ($1 - p$) is allocated to the remaining jobs in the queue) [9], have been shown to have product form solutions.

Product form networks can be solved efficiently using the convolution algorithms of Buzen [5] and Chandy/Herzog/Woo [6] as well as mean value analysis of Reiser and Lavenberg [29]–[31], as well as their variations [32].

Despite their popularity, several drawbacks do exist with product form networks. Probably the most significant of these is the assumptions that must be made when designing the system model. It is these assumptions that allow us to fit a given model to the format required for obtaining performance measures using product form network algorithms. Nevertheless, not all queueing networks will conform to one of the classes covered by product form algorithms. A queueing model containing even a single station not meeting one of the above mentioned seven basic types does not have a product form solution. This introduces problems when one considers the fact that service time distributions tend to demonstrate a high variance at CPU's (hyperexponential) and low variances at the I/O devices (Erlang). Furthermore, incorrectly assuming an exponential service time distribution can introduce significant errors into the results of performance evaluation for actual systems.

If a queueing network model is not amenable to a product form solution, it is often necessary to build and solve a Markov chain [36] which correctly accounts for the nonproduct form characteristics. However, such an approach

Manuscript received August 29, 1986; revised February 15, 1987. This work was supported in part by the Air Force Office of Scientific Research under Grant AFOSR-87-0160.

I. F. Akyildiz was with the Department of Computer Science, Louisiana State University, Baton Rouge, LA 70803. He is now with the School of Information and Computer Science, Georgia Institute of Technology, Atlanta, GA 30332.

A. Sieber was with the Department of Computer Science, Louisiana State University, Baton Rouge, LA 70803. He is now with the Department of Computer Architecture and Traffic Theory, University of Erlangen—Nuernberg, 8520 Erlangen, West Germany.

IEEE Log Number 8823661.

0098-5589/88/1100-1537$01.00 © 1988 IEEE

becomes infeasible for models requiring a Markov chain with a very large number of states. In such cases, simulation or approximate analytic techniques are required.

A large variety of classical approximation methods exist for dealing with distributions and/or scheduling disciplines not containing product form, for example, product form approximation methods such as diffusion approximation [13], [18], [19], EPF-technique (extended product form) of Shum and Buzen [34] or aggregation/decomposition methods such as iterative approximation [7], decomposition approximation [11], method of Kuehn [21], method of Marie [22]–[24] or maximal entropy method [20], [39] or response time preservation method [1] or mean value analysis approximations [2].

It should be noted that the above list does not contain all approximation methods. It is merely provided as a framework outlining the basic methods that currently exist for approximate methods to queueing networks.

Our experience indicates that existing or suggested approximate solution methods contain inherent disadvantages or restrictions. In some cases they provide results which differ widely from the exact values. In addition, these methods are restricted to networks with load-independent stations. The service times, visit ratios and routing probabilities do not vary with job population changes. These assumptions are too rigid for many real systems. For example, if the moving-arm disk employs a scheduler that minimizes arm movement, a measurement of the mean seek time during a lightly loaded baseline period will differ significantly from the average seek time observed in a heavily loaded projection period. Similarly, the visit ratios for a swapping device will differ in baseline and projection periods having different average levels of multiprogramming. Another example is in the modeling of multiprocessors, where account must be taken of the degradation in performance due to the memory interference and software lockout (mutually exclusive access to share data structures). All of these examples illustrate the importance of considering load-dependent behavior.

Analytical methods can deal with only two kinds of load-dependent behavior:

1) A station's service function may depend on the length of that station's queue [33].

2) The visit ratios and transition probabilities may depend on the total number of jobs in the system [38].

We will consider load-dependency in which the service time of a station may be dependent on the number of jobs at that station.

Load-dependent stations also allow us to analyze:

1) Multiserver Stations ($m_i > 1$): In this case, each server has a service rate μ_i. The total service rate of this station as a function of the number of jobs is:

$$\mu_i(k) = \min \{k\mu_i, m_i\mu_i\}.$$

2) Infinite Server Stations ($m_i = \infty$): An infinite server station is a special multiserver station with:

$$\mu_i(k) = k\mu_i.$$

The method proposed is an extension of the well-known method of Marie [22]–[24] to load-dependent general queueing networks. Our tests have shown that the method of Marie is the most reliable of the previously mentioned approximate methods. It provides satisfactory results for a wide range of queueing networks. It does, however, contain the inherent disadvantage of being limited to load independent networks.

The technique proposed is iterative in nature and is based on the concept of the conditional throughputs $v_i(k)$ of a station. In a closed network each station is analyzed under a Markovian arrival process with load-dependent arrival rate $\lambda_i(k)$. As a result we are able to analyze networks with $\lambda(k)/C_2/ld$ – FCFS (ld; load-dependent) station types.

II. Approximate Analysis of Load Dependent Networks

We consider closed queueing networks with the following characteristics:

1) There are N stations and K single class jobs.

2) The service time of each station is distributed with load-dependent mean value $1/\mu_i(k)$ (for $k = 1, \cdots, K$) and general distribution function $F_i(t)$ having a rational Laplace transform.

3) Each station has FCFS scheduling discipline and infinite capacity.

4) A job serviced by station i proceeds to station j with probability p_{ij} for $i, j = 1, 2, \cdots, N$.

A. Load Dependent Arrival Rates

To determine the load-dependent arrival rates $\lambda_i(k)$ (for $k = 0, 1, 2, \cdots, K$) of a station i (for $i = 1, \cdots, N$), the ith station is shorted, i.e., its service time is set to zero as shown in Fig. 1.

It is assumed that the subnetwork satisfies local balance and has product form solution. The throughput values $\lambda_c^i(k)$ of the subnetwork can be obtained by any product form network algorithm such as mean value analysis [31] or convolution algorithms [5], [6], [33] for load-dependent networks.

The load-dependent arrival rates $\lambda_i(k)$ to the ith station are then:

$$\lambda_i(k) = \lambda_c^i(K - k) \qquad \text{for } k = 0, 1, \cdots, K - 1. \tag{1}$$

It is clear that if k jobs are present at the ith station, then $(K - k)$ jobs remain in the subnetwork. Thus the throughput of the subnetwork with $(K - k)$ jobs is equal to the arrival rate of the ith station with k jobs. Note that

$$\lambda_i(K) = 0 \tag{2}$$

since no job is in the subnetwork and consequently the throughput will be zero, $\lambda_c^i(0) = 0$.

In this way each station is shorted and the throughput λ_c^i of the corresponding subnetwork is computed and assigned to the according arrival rates λ_i.

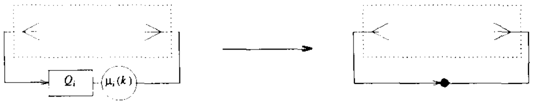


Fig. 1.

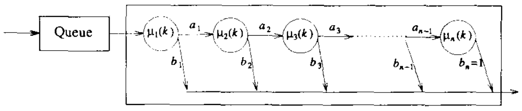


Fig. 2.

Note that in the first iteration the service rates of each station are the originally given values μ_i. In future iterations the conditional throughputs $v_i(k)$, computed in Section II-B, are used as the adjusted service rates for the stations.

B. Load Dependent Service Rates

We have assumed that the service times follow an arbitrary distribution with a rational Laplace transform. Cox [10] demonstrated that any distribution with rational Laplace transform can be represented by a sequence of fictitious phases as shown in Fig. 2.

Each time a job completes a phase it may either depart from the station or it may proceed to the next phase. The total time a job spends in a phase is exponentially distributed. In general, each phase has a different mean service rate. Let $\mu_{ij}(k)$ denote the load-dependent service rate at the jth phase ($j = 1, 2, \cdots, n$) of the ith station ($i = 1, \cdots, N$). Also, let a_j be the probability that a job upon completion of its service at the jth phase will proceed to the $(j + 1)$th phase. b_j denotes the probability that a job upon completion of its service at phase j will depart from the station. This type of service distribution is known as the Coxian distribution and it is denoted by C_n where n is the number of phases. Jobs are assumed to arrive at the station in a Poisson fashion at the load-dependent rate $\lambda(k)$ (computed in Section II-A). This type of station has the shorthand notation $\lambda(k)/C_n/ld - FCFS$. Cox's representation of arbitrary distributions with rational Laplace transforms is useful when dealing with nonexponential distributions. In the modeling of nonexponential service time distributions we have an estimation of the first and second moments of the service law, i.e., the expected value and the variance. Marie [22] has shown that for any mean μ (first moment) and any squared coefficient of variation (c^2 = variance$/\mu^2$) (second moment) such that ($0.5 \leq c^2 < \infty$), it is possible to represent a station's server by a Cox-model with two phases.

The parameters $\mu_{i1}(k)$, $\mu_{i2}(k)$, a_i and b_i are determined as follows [24]:

$$\mu_{i1}(k) = 2\mu_i(k) \tag{3a}$$

$$\mu_{i2}(k) = \frac{\mu_i(k)}{c_i^2} \tag{3b}$$

$$a_i = \frac{1}{2c_i^2} \tag{3c}$$

$$b_i = 1 - a_i. \tag{3d}$$

Note that for values of c_i less or equal to 0.5, Marie [24] suggested an Erlang type of distribution.

Various studies of single server and multiple server stations with Coxian distributions have been reported in the literature. These studies concentrated on the derivation of the probability distribution of the number of jobs in the system using various recursive procedures. In particular, Herzog, Woo, and Chandy [16] and Bux and Herzog [3] obtained numerical results for a single server station with state dependent arrival and service rates, and assuming that the interarrival times as well as the service times follow a Coxian distribution.

Marie [24] studied the queue length probability distribution of a single server station with a Coxian service time distribution and exponentially distributed load-dependent interarrival times. His approach is based on the notion of the conditional throughput $v_i(k)$ (adjusted service rate) which is obtained using a recursive formula. Marie's model is extended to multiple servers by Stewart and Marie [37] and Marie, Snyder, and Stewart [25] using numerical techniques. Perros [28] gives exact closed-form expression of the probability distribution of the number of jobs in an $M/C_n/1$ station.

In this section we derive the formulas for conditional throughputs $v_i(k)$ (for $k = 1, \cdots, K$) (adjusted service rates) for load-dependent networks in detail, since the classical method of Marie [22]–[24] differs from the technique presented only in the computation of $v_i(k)$.

A state of a station is denoted by a pair:

$$\mathbf{k} = (k, j)$$

where

k is the number of jobs, $k = 0, 1, 2, \cdots, K$
j is the number of phases, $j = 1, 2$.

A transition from one state to another takes place either when a new job arrives or when a job leaves the station through either phase. We will represent arrival rates by $\lambda_i(k)$, where the arrivals rates are dependent on the number of jobs in the station as computed in Section II-A. A job leaving the station after phase one is denoted by $b_i\mu_{i1}(k)$. $\mu_{i2}(k)$ is the departure rate of a job leaving the station after phase two. A job enters into phase two at a rate $a_i\mu_{i1}(k)$ and may do so only after receiving service in phase one. The state (0) denotes that no job is in the ith station.

To compute the probability of being in a state as shown in the state transition diagram, Fig. 3, we use the Chapman-Kolmogorov equations. Consider the probability $p_i(k, j)$ as the probability of being in the ith station of the network given k jobs and the current job in phase j. There are six cases to consider.

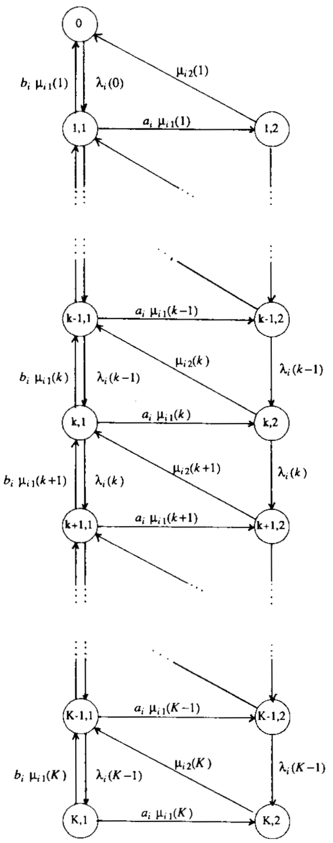


Fig. 3.

Case 1: $k = 0$.

$$p_i(0)\,\lambda_i(0) = p_i(1,1)\,b_i\mu_{i1}(1) + p_i(1,2)\,\mu_{i2}(1) \quad (4)$$

Case 2: $j = 1$ and $k \geq 1$.

$$p_i(k,1)[\mu_{i1}(k) + \lambda_i(k)] = p_i(k-1,1)\,\lambda_i(k-1) + p_i(k+1,1) \cdot b_i\mu_{i1}(k+1) + p_i(k+1,2)\,\mu_{i2}(k+1) \quad (5)$$

Case 3: $j = 2$ and $k = 1$.

$$p_i(1,2)[\lambda_i(1) + \mu_{i2}(1)] = p_i(1,1)\,a_i\mu_{i1}(1) \quad (6)$$

Case 4: $j = 2$ and $k > 1$.

$$p_i(k,2)[\lambda_i(k) + \mu_{i2}(k)] = p_i(k,1)\,a_i\mu_{i1}(k) + p_i(k-1,2)\,\lambda_i(k-1) \quad (7)$$

Case 5: $j = 1$ and $k = K$.

$$p_i(K,1)[\mu_{i1}(K)] = p_i(K-1,1)\,\lambda_i(K-1) \quad (8)$$

Case 6: $j = 2$ and $k = K$.

$$p_i(K,2)[\mu_{i2}(K)] = p_i(K-1,2)\,\lambda_i(K-1) + p_i(K,1)\,a_i\mu_{i1}(K) \quad (9)$$

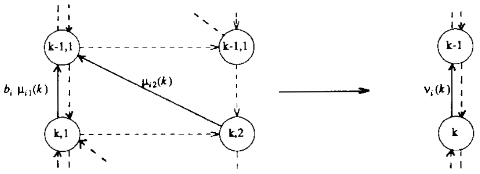


Fig. 4.

Let the probability of k jobs in a station, independent of which phase a job is executing in, be denoted as

$$p_i(k) = p_i(k,1) + p_i(k,2) \qquad \text{for } k = 1, \cdots, K. \quad (10)$$

We can express the conditional throughputs $v_i(k)$ (adjusted service rates) in terms of the equilibrium state probabilities and the service rates of the Cox phases. Note that the service rates of the Cox phases (3) do not vary in the entire execution of the algorithm:

$$v_i(k) = \frac{1}{p_i(k)}\,[p_i(k,1)\,b_i\mu_{i1}(k) + p_i(k,2)\,\mu_{i2}(k)]. \quad (11)$$

The derivation of (11) is obvious in Fig. 4.

The following theorem has been proven by Marie [22]:

$$p_i(k-1)\,\lambda_i(k-1) = p_i(k)\,v_i(k). \quad (12)$$

Simply stated, the probability that a job leaves a station in which there are k jobs, is equal to the probability that a job arrives at the same station when there are $(k-1)$ jobs.

To determine $v_i(k)$ we modify (7) and obtain:

$$p_i(k,1) = \frac{\lambda_i(k) + \mu_{i2}(k)}{a_i\mu_{i1}(k)}\,p_i(k,2) - \frac{\lambda_i(k-1)}{a_i\mu_{i1}(k)}\,p_i(k-1,2) \quad (13)$$

dividing by $p_i(k,2)$ gives

$$\frac{p_i(k,1)}{p_i(k,2)} = \frac{\lambda_i(k) + \mu_{i2}(k)}{a_i\mu_{i1}(k)} - \frac{\lambda_i(k-1)}{a_i\mu_{i1}(k)} \cdot \frac{p_i(k-1,2)}{p_i(k,2)}. \quad (14)$$

From (10) and (11) we obtain

$$v_i(k) = \frac{p_i(k,1)\,b_i\mu_{i1}(k) + p_i(k,2)\,\mu_{i2}(k)}{p_i(k,1) + p_i(k,2)} \quad (15)$$

which can be rewritten as

$$\frac{p_i(k,1)}{p_i(k,2)} = \frac{\mu_{i2}(k) - v_i(k)}{v_i(k) - b_i\mu_{i1}(k)}. \quad (16)$$

Substituting (16) into (14):

$$\frac{\mu_{i2}(k) - v_i(k)}{v_i(k) - b_i\mu_{i1}(k)} = \frac{\lambda_i(k) + \mu_{i2}(k)}{a_i\mu_{i1}(k)} - \frac{\lambda_i(k-1)}{a_i\mu_{i1}(k)} \cdot \frac{p_i(k-1, 2)}{p_i(k, 2)}. \quad (17)$$

To solve the (17) in terms of $v_i(k)$, without determining the state probabilities $p_i(k, j)$ the term

$$\left[\frac{p_i(k-1, 2)}{p_i(k, 2)}\right]$$

must be replaced in (17).

Observe that

$$\frac{p_i(k-1, 2)}{p_i(k, 2)} = \frac{p_i(k-1, 2)}{p_i(k-1)} \cdot \frac{p_i(k-1)}{p_i(k)} \cdot \left[1 + \frac{p_i(k, 1)}{p_i(k, 2)}\right]. \quad (18)$$

From (11) we obtain

$$p_i(k-1)\, v_i(k-1) = p_i(k-1, 1)\, b_i\mu_{i1}(k-1) + p_i(k-1, 2)\, \mu_{i2}(k-1). \quad (19)$$

Rewriting

$$p_i(k-1)\, v_i(k-1) + p_i(k-1, 2)\, b_i\mu_{i1}(k-1) = p_i(k-1)\, b_i\mu_{i1}(k-1) + p_i(k-1, 2)\, \mu_{i2}(k-1). \quad (20)$$

From (20) the following equation can be derived:

$$\frac{p_i(k-1, 2)}{p_i(k-1)} = \frac{v_i(k-1) - b_i\mu_{i1}(k-1)}{\mu_{i2}(k-1) - b_i\mu_{i1}(k-1)}. \quad (21)$$

Substituting (12), (16), and (21) in (18) the term for equilibrium state probabilities in (17) can be expressed as follows:

$$\frac{\mu_{i2}(k) - v_i(k)}{v_i(k) - b_i\mu_{i1}(k)} = \frac{\lambda_i(k) + \mu_{i2}(k)}{a_i\mu_{i1}(k)} - \frac{\lambda_i(k-1)}{a_i\mu_{i1}(k)} \cdot \frac{v_i(k-1) - b_i\mu_{i1}(k-1)}{\mu_{i2}(k-1) - b_i\mu_{i1}(k-1)} \cdot \frac{v_i(k)}{\lambda_i(k-1)} \cdot \left[1 + \frac{\mu_{i2}(k) - v_i(k)}{v_i(k) - b_i\mu_{i1}(k)}\right]. \quad (22)$$

The solution of (22) after $v_i(k)$ provides the desired formula for the recursive computation of the conditional throughputs $v_i(k)$ (adjusted service rates) from the parameters of the Cox-2-distribution and the arrival rates $\lambda_i(k)$ for $k > 1$.

$$v_i(k) = \frac{[\mu_{i2}(k-1) - b_i\mu_{i1}(k-1)][\mu_{i1}(k)\,\mu_{i2}(k) + b_i\mu_{i1}(k)\,\lambda_i(k)]}{\{[\lambda_i(k) + \mu_{i2}(k) + a_i\mu_{i1}(k)][\mu_{i2}(k-1) - b_i\mu_{i1}(k-1)]\} - \{[v_i(k-1) - b_i\mu_{i1}(k-1)][\mu_{i2}(k) - b_i\mu_{i1}(k)]\}}. \quad (23)$$

The computation for $v_i(k)$ is an iteration process based on previous $v_i(k)$ values. Analogous to (6) we obtain:

$$\frac{p_i(1, 1)}{p_i(1, 2)} = \frac{\lambda_i(1) + \mu_{i2}(1)}{a_i\mu_{i1}(1)}. \quad (24)$$

Equation (16) is also valid for $k = 1$. Substituting (16) in (24) we obtain:

$$\frac{\lambda_i(1) + \mu_{i2}(1)}{a_i\mu_{i1}(1)} = \frac{\mu_{i2}(1) - v_i(1)}{v_i(1) - b_i\mu_{i1}(1)}. \quad (25)$$

Solving (25) yields:

$$v_i(1) = \frac{b_i\lambda_i(1)\,\mu_{i1}(1) + \mu_{i1}(1)\,\mu_{i2}(1)}{\lambda_i(1) + \mu_{i2}(1) + a_i\mu_{i1}(1)}. \quad (26)$$

Note that in case of load-independent stations, it is valid that

$$\mu_{ij}(k) = \mu_{ij} \qquad \text{for } i = 1, \cdots, N;\ k = 1, \cdots, K;\ j = 1, 2. \quad (27)$$

Substituting (27) in (23) the following formula for conditional throughputs $v_i(k)$ for load-independent networks is obtained, which is also given by Marie [24].

$$v_i(k) = \frac{b_i\lambda_i(k)\mu_{i1} + \mu_{i1}\mu_{i2}}{\lambda_i(k) + \mu_{i1} + \mu_{i2} - v_i(k-1)} \quad (28)$$

C. Equilibrium State Probabilities

From (12) and considering that

$$\sum_{k=0}^{K} p_i(k) = 1$$

the equilibrium state probabilities $p_i(k)$ for $k = 0, 1, \cdots, K$ are obtained as follows [23]:

$$p_i(k) = p_i(0) \prod_{n=0}^{k-1} \frac{\lambda_i(n)}{v_i(n+1)} \qquad \text{for } k = 1, \cdots, K \quad (29)$$

where

$$p_i(0) = \left[1 + \sum_{k=1}^{K} \prod_{n=0}^{k-1} \frac{\lambda_i(n)}{v_i(n+1)}\right]^{-1}. \quad (30)$$

D. Termination Test

After each iteration we check to see if the sum of the mean number of jobs is equal to the total number of jobs in the given network within a tolerance level ϵ:

$$\frac{\left| K - \sum_{i=1}^{N} \sum_{k=1}^{K} k \cdot p_i(k) \right|}{K} < \epsilon. \qquad (31)$$

Additional check is made to see if the throughput rates of each station are consistent with the topology of the network:

$$\frac{\left| \frac{1}{e_j} \sum_{k=1}^{K} p_j(k)\, v_j(k) - \frac{1}{N} \sum_{i=1}^{N} \frac{1}{e_i} \sum_{k=1}^{K} p_i(k)\, v_i(k) \right|}{\left| \frac{1}{N} \sum_{i=1}^{N} \frac{1}{e_i} \sum_{k=1}^{K} p_i(k)\, v_i(k) \right|} < \epsilon$$

$$\text{for } j = 1, \cdots, N \qquad (32)$$

where e_i is the mean number of visits that a job makes to station i and is compared by:

$$e_i = \sum_{j=1}^{N} e_i p_{ji}$$

and ϵ is a tolerance level. Usual value is $\epsilon = 10^{-4}$.

If one or both of these conditions are violated, the service rates are adjusted

$$\mu_i(k) := v_i(k) \qquad \text{for } k = 1, \cdots, K, \qquad (33)$$

i.e., the conditional throughputs [(25) and (28)] are assumed to be the new service rates, and the next iteration is carried out. Iterations continue until acceptable tolerances are obtained. The performance measures can then be computed by the following formulas.

- Throughput of each station:

$$\lambda_i(K) = \sum_{k=1}^{K} p_i(k)\, \mu_i(k). \qquad (34)$$

- Mean number of jobs in each station:

$$\bar{k}_i(K) = \sum_{k=1}^{K} k \cdot p_i(k). \qquad (35)$$

- Mean residence time:

$$\bar{t}_i(K) = \frac{\bar{k}_i(K)}{\lambda_i(K)}. \qquad (36)$$

III. Algorithm Summary

The following is the complete algorithm for calculating performance measures for the load-dependent general networks:

1) Determine the parameters for the Cox distribution using (3).

2) Iterative Part:

a) Compute the throughput values $\lambda_c^i(k)$ for each subnetwork without the ith station (for $i = 1, \cdots, N$) using the convolution algorithm [5], [6], [33] or mean value analysis for load dependent stations [31]. Assign the throughput values $\lambda_c^i(k)$ with k jobs to the arrival rates of the ith station with $(K - k)$ jobs (1).

b) Compute the conditional throughputs $v_i(k)$ using (23) and (26) for $i = 1, \cdots, N$ and $k = 1, \cdots, K$.

c) Compute the equilibrium state probabilities for stations $i = 1, \cdots, N$ using (29) and (30).

d) Check the termination conditions (31) and (32). If the test is not successful, then apply (33) and goto a). Otherwise compute the performance measures from (34)–(36).

IV. Evaluation

The algorithm has been implemented on a VAX 11/780 system. The computation of load-dependent arrival rates, in Section II-A has been realized by both the convolution algorithm [5], [6], [33] and also mean value analysis for load dependent networks [31]. Our tests have shown that mean value analysis for load-dependent networks has the advantage of handling a greater number of jobs. Several different networks containing two to ten stations were analyzed, with the number of jobs ranging from ten to eighty in each network. The termination value ϵ ranged from 10^{-4} to 10^{-6}. The vast majority of the variations beneath 5 percent. It is clearly evident that this approximation method is capable of accurate modeling of load dependency. It should be noted that in all instances where the algorithm showed a relatively high deviation from the actual results (over 8 percent), the numbers involved were quite small. In such cases the relative error might appear large even though the difference in the two numbers is insignificant. Under these circumstances, the relative error cannot be considered a reliable indicator of the accuracy of the method.

The number of iterations in the method is not predictable. It depends on the number of stations, the number of jobs, the complementary subnetworks and the epsilon value used. In most of the cases the method converges in 6–10 iterations. Although no mathematical proof for convergence is given here or also in [22]–[24] we were unable to find any model with load dependent stations in which the method did not converge.

When analyzing the complexity of the algorithm we find that the space complexity of the algorithm is $O(3NK)$. The majority of this space was required for the computation of the equilibrium state probabilities (29), (30). Analyzing the time complexity of the method shows that it increases drastically as the number of jobs K in the network increases.

In the following we give nine examples with different input parameters. In Examples 1–5 networks with single, multiple, and infinite servers are analyzed. In Examples 6 and 7 we treat networks with load-dependent service rates. Approximate, exact, or simulation results for performance measures such as throughput and the mean number of jobs are given in tables. The approximate results are compared with exact values obtained by numerical analysis [36] in case of small number of jobs. However, numerical analysis cannot be applied for large number of jobs such as in Examples 8 and 9. The approximation is validated by simulation in those cases. The simulation results are obtained within 95 percent confidence interval. The tables also contain relative deviations which are computed by:

$$\delta = \left\{ \frac{\left| \text{Exact (or Simulation)} - \text{Approximation} \right|}{\text{Exact (or Simulation)}} \right\} * 100.$$

In Example 3 we plot the marginal probabilities as a function of the number of jobs in each station. In the graphs, the solid lines show approximate results, the dashed lines show exact results and the dotted lines show the case where the service time distributions are assumed to be exponential (i.e., $c_i = 1$ for all i).

In Example 4 we calculate the performance measures for different c_i^2 parameters. We plot total throughput results for different squared coefficient of variation values in Example 5.

Example 1: (Tandem Network); $K = 5$ jobs.

Station	μ_i	m_i	c_i^2
1	4.0	1	0.5
2	2.0	3	10
3	2.0	2	5
4	1.0	3	1

	Approximation	Exact	δ (%)
$\bar{k}_1$	0.6907	0.67777	1.92
$\bar{k}_2$	1.0322	1.0340	0.2
$\bar{k}_3$	1.2438	1.2669	1.82
$\bar{k}_4$	2.0333	2.0214	0.59
λ	1.8108	1.7803	1.71

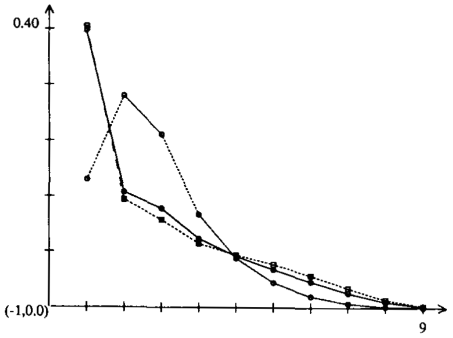
Example 2: $K = 9$ jobs.

Station	μ_i	m_i	c_i^2	p_{i1}	p_{i2}	p_{i3}
1	1.0	9	100	0.0	1.0	0.0
2	5.0	1	0.5	0.3	0.0	0.7
3	2.0	2	25	0.5	0.5	0.0

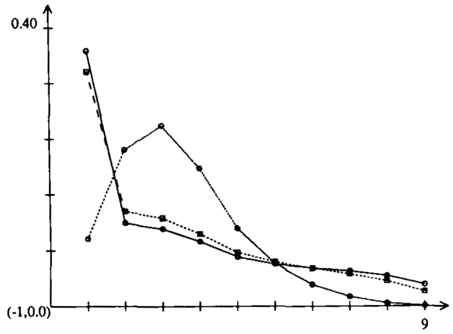
	Approximation	Exact	δ (%)
$\bar{k}_1$	2.1694	2.1578	0.54
$\bar{k}_2$	3.7252	3.7788	1.42
k_3	3.1054	3.0634	1.37
λ	2.1694	2.1578	0.54

Example 3:

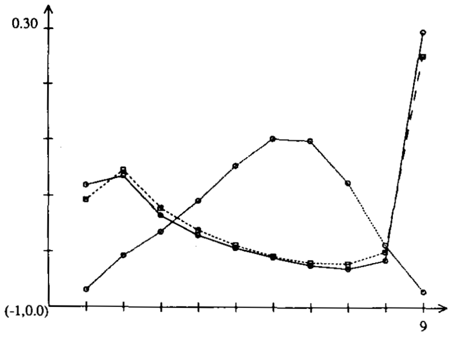
Station	μ_i	m_i	c_i^2	p_{i1}	p_{i2}	p_{i3}
1	2	3	0.8	0.0	0.7	0.3
2	1	4	4	1.0	0.0	0.0
3	0.5	2	25	1.0	0.0	0.0



Plot of marginal probabilities of station 1 as a function of jobs



Plot of marginal probabilities of station 2 as a function of jobs



Plot of marginal probabilities of station 3 as a function of jobs

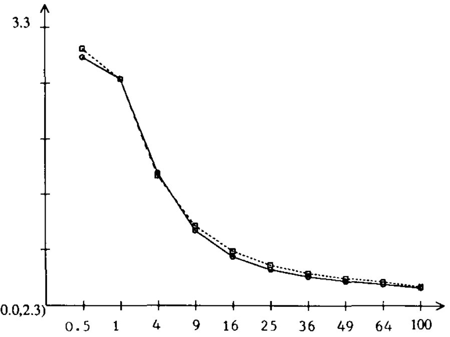
Example 4: (Tandem Network); $K = 10$ jobs; c_i^2 are variable.

Station	μ_i	m_i
1	1	6
2	2	4
3	5	1

c_1^2	c_2^2	c_3^2	λ_{ap}	λ_{ex}	δ(%)	$\bar{k}_{1,ap}$	$\bar{k}_{1,ex}$	δ(%)	$\bar{k}_{2,ap}$	$\bar{k}_{2,ex}$	δ(%)	$\bar{k}_{3,ap}$	$\bar{k}_{3,ex}$	δ(%)
100	100	100	2.806	2.825	0.67	3.820	3.826	0.16	2.457	2.448	0.38	3.723	3.726	0.08
100	100	0.5	2.994	2.970	0.81	3.952	3.917	0.89	2.541	2.521	0.79	3.506	3.562	1.57
100	0.5	100	3.134	3.147	0.41	4.169	4.187	0.43	1.998	2.033	1.72	3.833	3.780	1.40
100	0.5	0.5	3.561	3.554	0.20	4.676	4.695	0.40	1.899	1.901	0.11	3.424	3.403	0.62
0.5	100	100	3.081	3.069	0.39	3.826	3.823	0.08	2.598	2.588	0.39	3.577	3.589	0.33
0.5	100	0.5	3.754	3.718	0.97	4.041	3.960	2.05	3.123	3.138	0.48	2.836	2.901	2.24
0.5	0.5	100	3.594	3.595	0.03	4.017	4.031	0.35	1.915	1.884	1.65	4.067	4.084	0.42
0.5	0.5	0.5	4.553	4.628	1.62	4.771	4.820	1.02	2.361	2.363	0.08	2.868	2.817	1.81

Example 5: (Tandem Network); $K = 6$ jobs; $c_1^2 = c_2^2 = c_3^2$.

Station	μ_i	m_i
1	3.0	2
2	1.0	5
3	2.0	3



Plot of throughput as a function of the squared coefficient of variation c_i^2

Example 6: (Tandem Network); $K = 7$ jobs; $c_1^2 = 0.5$; $c_2^2 = 16$; $c_3^2 = 4$.

k	$\mu_1(k)$	$\mu_2(k)$	$\mu_3(k)$
1	1.0	1.5	2.0
2	1.0	1.8	2.0
3	1.0	1.9	2.2
4	1.2	2.0	2.5
5	1.4	2.0	3.0
6	1.5	2.0	3.0
7	4.0	2.0	3.5

	Approximation	Exact	δ (%)
$\bar{k}_1$	3.5202	3.5031	0.49
$\bar{k}_2$	2.2260	2.1979	1.28
$\bar{k}_3$	1.2538	1.2991	3.49
λ	1.1659	1.1530	1.12

Example 7: (Central Server Model); $K = 8$ jobs.

Station	c_i^2	p_{i1}	p_{i2}	p_{i3}
1	16	0	0.7	0.3
2	0.9	1.0	0	0
3	9	0.8	0.2	0

k	$\mu_1(k)$	$\mu_2(k)$	$\mu_3(k)$
1	1.0	0.5	2.0
2	1.5	0.5	2.0
3	1.8	1.0	2.2
4	2.0	2.0	2.5
5	2.3	3.0	3.0
6	2.5	3.0	3.5
7	3.0	3.0	4.0
8	4.0	3.0	10.0

	Approximation	Exact	δ (%)
$\bar{k}_1$	2.6312	2.6556	0.92
$\bar{k}_2$	3.0868	3.0815	0.17
$\bar{k}_3$	2.2820	2.2629	0.84
λ	1.4529	1.4477	0.36

Example 8: $K = 50$ jobs.

Station	μ_i	m_i	c_i^2	p_{i1}	p_{i2}	p_{i3}	p_{i4}
1	2	7	25	0	0	0.8	0.2
2	5	1	0.5	0.7	0	0.3	0
3	2	10	1	0	0.2	0	0.8
4	1	20	9	0	0.5	0	0.5

	Approximation	Simulation	*Sim–delta* (%)	*dev.* δ (%)
$\bar{k}_1$	2.4253	2.4593	3.7	1.38
$\bar{k}_2$	36.8822	37.0321	0.3	0.40
$\bar{k}_3$	2.1485	2.1547	0.4	0.29
$\bar{k}_4$	8.5440	8.3539	0.8	2.23
λ	3.4980	3.4989	0.75	0.03

Example 9: (Tandem Network); $K = 50$ jobs.

Station	μ_i	m_i	c_i^2
1	5	3	1
2	6	2	0.5
3	4	2	10
4	5	1	25
5	2	6	0.8
6	3	2	16
7	1	5	4

	Approximation	Simulation	*Sim–delta* (%)	*dev.* δ (%)
$\bar{k}_1$	0.7920	0.8324	0.6	4.85
$\bar{k}_2$	0.7024	0.7970	0.6	11.87
$\bar{k}_3$	4.0239	4.0773	4.2	1.31
$\bar{k}_4$	17.1263	15.9290	6.0	7.52
$\bar{k}_5$	1.9291	2.0310	0.8	5.02
$\bar{k}_6$	10.7728	10.5129	2.5	2.47
$\bar{k}_7$	14.6536	15.8204	1.9	7.38
λ	3.8373	3.6656	0.7	4.68

References

[1] S. C. Agrawal, J. P. Buzen, and A. W. Shum, "Response time preservation: A general technique for developing approximate algorithms for queueing networks," in *Proc. ACM Sigmetrics Conf.*, Cambridge, MA, Aug. 21-24, 1984, pp. 63-77.

[2] I. F. Akyildiz, "Mean value analysis of closed queueing networks with Erlang service time distributions," *Comput. J.*, vol. 39, pp. 219-232, Oct. 1987.

[3] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacios, "Open, closed, and mixed networks of queues with different classes of customers," *J. ACM*, vol. 22, no. 2, pp. 248-260, Apr. 1975.

[4] W. Bux and U. Herzog, "The phase concept: Approximation of measured data and performance analysis," in *Computer Performance Proceedings*, K. M. Chandy and M. Reiser, Eds. Amsterdam, The Netherlands: North-Holland, 1977.

[5] J. P. Buzen, "Computational algorithms for closed queueing networks with exponential servers," *Commun. ACM*, vol. 16, no. 9, pp. 527-531, Sept. 1973.

[6] K. M. Chandy, U. Herzog, and L. Woo, "Parametric analysis of queueing network models," *IBM J. Res. Develop.*, pp. 36-42, Jan. 1975.

[7] —, "Approximate analysis of general queueing networks," *IBM J. Res. Develop.*, vol. 19, no. 1, pp. 43-49, Apr. 1975.

[8] K. M. Chandy, J. H. Howard, and D. Towsley, "Product form and local balance in queueing networks," in *Modelling and Performance Evaluation of Computer Systems*, Workshop Preprints, 1976, pp. 89-103.

[9] K. M. Chandy and J. Martin, "A characterization of product form queueing networks," *J. ACM*, vol. 30, no. 2, pp. 286-299, 1983.

[10] D. R. Cox, "A use of complex probabilities in the theory of stochastic processes," *J. Proc. Cambr. Phil. Soc.*, vol. 51, pp. 313-319, 1955.

[11] P. J. Courtois, *Decomposability: Queueing and Computer System Applications.* New York: Academic, 1977.

[12] D. Ferrari, G. Serazzi, and A. Zeigner, *Measurement and Tuning of Computer Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1983.

[13] E. Gelenbe, "On approximate computer system models," *J. ACM*, vol. 22, no. 2, pp. 261-269, Apr. 1975.

[14] E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer System Models.* London: Academic, 1980.

[15] W. J. Gordon and G. F. Newell, "Closed queueing systems with exponential servers," *Oper. Res.*, vol. 15, pp. 254-265, 1967.

[16] U. Herzog, L. Woo, and K. Chandy, "Solution of queueing problems by a recursive technique," *IBM J. Res. Develop.*, pp. 295-300, May 1975.

[17] P. Jackson, "Job shop like queueing systems," *Management Sci.*, vol. 10, no. 1, 1963.

[18] H. Kobayashi, "Application of the diffusion approximation to queueing networks, Part I: Equilibrium queue distributions," *J. ACM*, vol. 21, no. 2, pp. 316-328, Apr. 1974.

[19] —, "Application of the diffusion approximation to queueing networks, Part II: Nonequilibrium distributions and computer modelling," *J. ACM*, vol. 21, no. 3, pp. 459-469, July 1974.

[20] D. D. Kouvatsos, "A maximum entropy queue length distribution for a G/G/1 finite capacity queue," *ACM Sigmetrics Perform. Eval. Rev.*, vol. 14, no. 1, pp. 224-237, May 1986.

[21] P. J. Kuehn, "Approximate analysis of general queueing networks by decomposition," *IEEE Trans. Commun.*, vol. 27, no. 1, pp. 113-126, Jan. 1979.

[22] R. Marie, "Methodes iteratives de resolution de modeles mathematiques de systemes informatiques," *R.A.I.R.O. Informatique/Comput. Sci.*, vol. 12, pp. 107-122, 1978.

[23] ——, "An approximate analytical method for general queueing networks," *IEEE Trans. Software Eng.*, vol. SE-5, no. 5, pp. 530-538, Sept. 1979.

[24] ——, "Calculating equilibrium probabilities for $\lambda(n)/C/1 - N$ queues," *Perform. Eval. Rev.*, (*ACM Sigmetrics Conference*), vol. 9, no. 2, pp. 117-125, Summer 1980.

[25] R. Marie, P. M. Snyder, and W. J. Stewart, "Extensions and computational aspects of an iterative method," in *Proc. ACM Sigmetrics Conf.*, vol. 11, no. 4, Seattle, WA, Aug. 1982, pp. 186-195.

[26] R. R. Muntz, "Poisson departure process and queueing networks," in *Sciences and Systems, Proc. 7th Annu. Princeton Conf. Information*, Princeton Univ., Princeton, NJ, Mar. 1973, pp. 435-440.

[27] A. S. Noetzel, "A generalized queueing discipline for product form network solutions," *J. ACM*, vol. 26, no. 4, pp. 779-793, Oct. 1979.

[28] H. G. Perros, "The queue length distribution of the $M/C_k/1$ queue," *ACM Sigmetrics Perform. Eval.*, pp. 21-24, 1983.

[29] M. Reiser, "Mean value analysis of queueing networks: A new look at an old problem," in *Proc. Fourth Int. Symp. Modeling and Performance Evaluation of Computer Systems*, vol. I, Feb. 1979.

[30] M. Reiser and S. S. Lavenberg, "Mean value analysis of closed multichain queueing networks," *J. ACM*, vol. 27, no. 2, pp. 313-322, Apr. 1980.

[31] M. Reiser, "Mean value analysis and convolution method for queueing dependent servers in closed queueing networks," *Perform. Eval.*, vol. 1, no. 1, pp. 7-18, Jan. 1981.

[32] C. H. Sauer and K. M. Chandy, *Computer Systems Performance Modeling*. Englewood Cliffs, NJ: Prentice-Hall, 1981.

[33] C. H. Sauer, "Computational algorithms for state dependent queueing networks," *ACM Trans. Comput. Syst.*, vol. 1, no. 1, pp. 67-92, Feb. 1983.

[34] A. W. C. Shum, "Queueing models for computer systems with general service time distributions," Ph.D. dissertation, Div. Eng. Appl. Phys., Univ. Massachusetts, Dec. 1976.

[35] J. P. Spirn, "Queueing networks with random selection for service," *IEEE Trans. Software Eng.*, vol. SE-5, pp. 287-289, 1979.

[36] W. J. Stewart, "A comparison of numerical techniques in Markov modeling," *Commun. ACM*, vol. 21, no. 2, pp. 144-152, Feb. 1978.

[37] W. L. Stewart and R. Marie, "A numerical solution for the $\lambda(n)/C_k/r/N$ queue," *European J. Oper. Res.*, vol. 5, pp. 56-68, 1980.

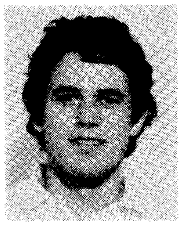
[38] D. Towsley, "Queueing network models with state-dependent routing," *J. ACM*, vol. 27, no. 2, pp. 323-337, Apr. 1980.

[39] R. H. Walstra, "Non-exponential networks of queues: A maximum entropy analysis," *ACM Sigmetrics Perform. Eval. Rev.*, vol. 13, no. 2, Aug. 1985.

I. F. Akyildiz (M'86) was born in Istanbul, Turkey, in 1954. He received the Vordiplom, Diplom Informatiker, and Doctor of Engineering degrees in computer science from the University of Erlangen—Nuernberg, West Germany, in 1978, 1981, and 1984, respectively.

From 1981 through 1985 he served as a Scientific Employee with the Department of Informatik IV (Operating Systems) at the University of Erlangen—Nuernberg. During that time he coauthored a textbook entitled *Analysis of Computer Systems* (in German) published by Teubner-Verlag in the Fall of 1982. From 1985 through 1987 he was an Assistant Professor with the Department of Computer Science at Louisiana State University, Baton Rouge. In the Fall of 1987 he joined the faculty in the School of Information and Computer Science at Georgia Institute of Technology, Atlanta, as an Assistant Professor. His research interests are performance evaluation of computer networks and distributed systems, in particular, analytical modeling and simulation, and operating systems, in particular, protection and security.

Dr. Akyildiz is a member of the Association for Computing Machinery (Sigops and Sigmetrics), GI (Gesellschaft fuer Informatik), and MMB (German Interest Group in Measurement, Modeling, and Evaluation of Computer Systems).



Albrecht Sieber was born in Erlangen, West Germany, in 1961. He received the Diplom Informatiker degree in computer science from the University of Erlangen-Nuernberg in 1987.

From 1986 through 1987 he spent a year at Louisiana State University, Baton Rouge, as a Research Assistant. Currently he is a Research Assistant with the Department of Informatik VII (Computer Architecture and Traffic Theory) at the University of Erlangen—Nuernberg. His doctoral research is focused on performance evaluation of multiprocessors and communication networks.