Numerical Methods for Modeling Computer Networks Under Nonstationary Conditions

DAVID TIPPER, MEMBER, IEEE, AND MALUR K. SUNDARESHAN, MEMBER, IEEE

Abstract—**Computer communication networks are frequently subjected to a variety of nonstationary phenomena during operation, resulting in considerable periods when nonstationary conditions prevail. However, the majority of available techniques for network modeling in order to conduct studies of performance evaluation or the design of network control strategies have been developed under simplifying approximations of steady-state conditions. In this paper, we discuss numerical techniques for modeling computer networks under nonstationary conditions, and two distinct approaches are presented. The first approach employs a queueing theory formulation to develop differential equation models which describe the behavior of the network by time-varying probability distributions. In the second approach a nonlinear differential equation model is developed for representing the dynamics of the network in terms of time-varying mean quantities. This approach allows multiple classes of traffic to be modeled and establishes a framework for the use of optimal control techniques in the design of network control strategies. Numerical techniques for determining the queue behavior as a function of time for both approaches are discussed and their computational advantages are contrasted with simulation.**

I. Introduction

FUNDAMENTAL to the problems of performance evaluation and the design of efficient control strategies (viz. routing, flow control, buffer management, etc.) for computer networks is the development of an appropriate model for representing the network. Although a wide variety of techniques and mathematical tools have been used in handling the modeling problem, most of these focus on the principal feature of a computer network, viz. the contention for shared resources, and the resultant queueing for these resources. Quite naturally, the most widely used models are based on queueing theory and these models represent the computer network as a network of interconnected queues. There has been a considerable effort in analyzing queueing networks with particular emphasis on their applications to computer communication systems [1], [2].

Although very significant advances have been made in the development of queueing network models, which have served as the basis for various studies aimed at performance evaluation and control design procedures, these models often make several oversimplifying assumptions not always justifiable in practice. Specifically, a major shortcoming of a vast majority of the currently available queueing models is that they describe only the steady-state or stationary (i.e., long-term) behavior of the queueing system; consequently, any control algorithm tailored on the basis of such models can ensure optimal performance only under steady-state conditions.

Recently, it has been increasingly noted that computer communication networks not only must have good steady-state performance but also must deliver acceptable performance under nonstationary and transient conditions [3]–[10]. Transient or nonstationary conditions occur in computer networks when the statistics of the packet arrival processes to the various network queues or the service rates of the queues vary with time. A simple evaluation of the time constant (relaxation time) of the widely used $M/M/1$ queueing model of a network link [3] indicates that the time taken by the queue at the link to reach steady-state after an event that generates transient conditions will be quite long, particularly when the link is heavily loaded; hence periods of nonstationary or transient behavior prevail during much of the time.

Typical events that give rise to nonstationary or transient conditions in computer networks are load sharing, changes in routing and flow control parameters (i.e., adaptive routing and flow control), failures of links, nodes, or other network resources, topological changes, network start-up and shutdown, and, most importantly, nonstationary input loads. It is well known [11] that in many packet switched computer networks, the user demand for data communication varies so rapidly that the load is essentially nonstationary for large time periods. In fact, it is in recognition of the nonstationary conditions that exist in most packet switched wide area networks (WAN's) that there has been such a considerable effort to develop adaptive routing and flow control methods. Furthermore, it has recently been noted [4] that a class of data networks called rapidly reconfigurable networks (RRN's) exists. These networks are subject to frequent, if not continuous, changes in the combination of network geometry, user demand for data communication, and transmission link capacity, such that nonstationary conditions always exist and the nonstationary behavior is the only meaningful measure of performance. Thus there exists a need for techniques to analyze the time-varying behavior of computer communication networks.

Manuscript received September 27, 1989; revised April 10, 1990, and June 30, 1990. An earlier version of this paper was presented at the IEEE Conference INFOCOM'89, Ottawa, Canada, April 24-27, 1989.

D. Tipper is with the Electrical and Computer Engineering Department, Clemson University, Clemson, SC 29634-0915.

M. K. Sundareshan is with the Electrical and Computer Engineering Department, University of Arizona, Tucson, AZ 85721.

IEEE Log Number 9039272.

In this paper, the problem of modeling computer networks under nonstationary conditions is considered with the focus on techniques that can be used in the performance evaluation and the design of control procedures. It should be emphasized that the techniques developed herein are not limited to computer networks, but are applicable to the general class of nonstationary queueing systems. The organization of the paper is as follows. In Section II, a Markov-process-based queueing theory approach is employed to develop differential equation models which describe the network queues by time-varying probability distributions. Numerical methods for solving the differential equations to determine the queue behavior in a nonstationary environment are discussed and numerical examples presented. In Section III, a nonlinear differential equation model for representing the queue dynamics at the various transmission links in the network in terms of time-varying mean quantities is presented. The numerical solution of this state variable model in order to conduct nonstationary performance evaluations of queueing systems is discussed and a framework for using this model to design control strategies which ensure optimum network performance under both nonstationary and steady-state conditions is described. Lastly, Section IV summarizes the paper.

II. A Numerical Queueing Theory Approach

The most widely known analytical models of computer networks for use in the performance evaluation and design of routing and flow control algorithms are queueing network models. Most of the literature on queueing networks focuses on the steady-state conditions under which an analysis may readily be carried out. Tractable analysis of a queueing network is possible when the network lends itself to the so-called product form solution [1], [2], [12].

The traditional approach using queueing theory methods for designing ''optimal'' routing and flow control strategies is to formulate a steady-state product form queueing model for the network or a portion of the network under consideration (i.e., a node or a virtual circuit) and to then derive an expression for a suitable performance measure in terms of the queueing model. One can then pose an optimization problem to be solved by mathematical programming techniques to determine the optimal steady-state control parameters. To provide a certain degree of adaptivity to changes in the incoming traffic demand and/or network topology, an updating of the routing and flow control parameters is commonly employed by monitoring the network conditions at periodic intervals and recalculating the optimum steady-state control at each period. This is the so-called quasi-static or quasi-stationary approach [2]. It may be noted that this approach assumes static loading conditions during each updating period allowing the network to attain steady state. Thus the network may be assumed to go through a series of steady-state periods. Although extensively referred to in the literature, there has been very little effort to determine the conditions under which the quasi-static assumption may be justified, further emphasizing the desirability of developing models which can capture the more general time-varying behavior of computer networks.

A. Evaluation of Nonstationary Queue Dynamics

There have been few studies concerning the nonstationary characteristics of data networks, largely because there are few time-dependent solutions currently available for the queueing models of these networks [12], [13]. Furthermore, the queueing models that do exist are often complex from a computational perspective and are awkward to manipulate. In this section, we describe a numerical scheme for evaluating the nonstationary behavior of computer networks. For a concise description of the procedure that will be followed, we begin with the well-known Chapman-Kolmogorov differential equations describing the time-dependent state probabilities of a finite capacity $M/M/1$ queue with time-varying average arrival and service rates [12]:

$$\begin{aligned}\frac{dp^0(t)}{dt} &= -\lambda^0(t)p^0(t) + \mu^1(t)p^1(t)\\ \frac{dp^n(t)}{dt} &= \lambda^{n-1}(t)p^{n-1}(t) - \left[\lambda^n(t) + \mu^n(t)\right]p^n(t)\\ &\quad + \mu^{n+1}(t)p^{n+1}(t), \qquad 0 < n < K\\ \frac{dp^K(t)}{dt} &= \lambda^{K-1}(t)p^{K-1}(t) - \mu^K(t)p^K(t). \qquad (1)\end{aligned}$$

Here $p^n(t)$ is the probability of n customers in the system (i.e., queue + service) at time t, $\lambda^n(t)$ is the average arrival rate to the queue if there are n customers in the system, $\mu^n(t)$ is the average service rate if there are n customers in the system, and K is the capacity of the system.

This set of differential equations is notoriously difficult to solve analytically due to the time-varying coefficients. Even in the simplest case, when the arrival rate and service rate are constant (i.e., $\lambda^n(t) = \lambda$, $\mu^n(t) = \mu$) and a steady-state equilibrium will eventually be reached, the solution of (1) for the transient behavior of $p^n(t)$ involves an infinite sum of Bessel functions [12]. This is computationally difficult and there has been a considerable effort [14]-[16] to develop efficient computational methods to accurately determine $p^n(t)$ for this simplest case. It is well recognized that in the general nonstationary case, the solution of (1) by analytical means to obtain a compact expression for $p^n(t)$ is very difficult even if $\lambda^n(t)$ and $\mu^n(t)$ are smooth well-behaved functions [12]. This problem can be circumvented by applying numerical techniques using an approach similar to that proposed by Odoni and Roth [17] and in a separate study by Van As [5]. The basic idea is to approximate the time-varying average arrival and service rates by constants over small time intervals. This allows one to numerically solve (1) for the state probabilities over one time interval and to repeat the procedure for all time intervals of interest. The

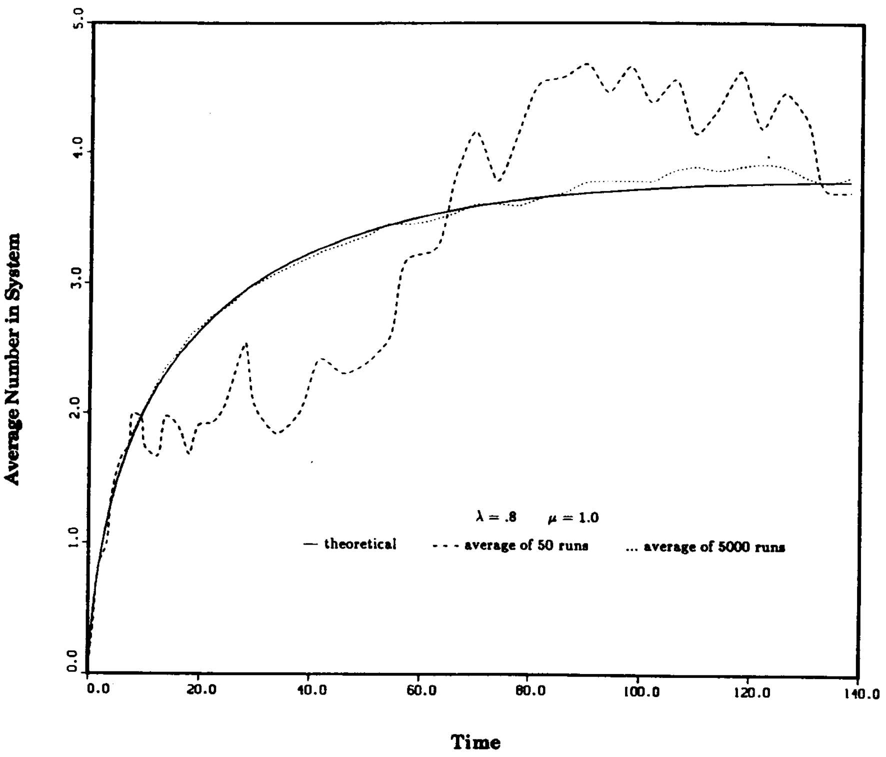


Fig. 1. Time-varying behavior of the average number in the system for an $M/M/1/20$ queue.

exact steps in the solution technique proposed here are described in the following.

Begin with some known boundary condition $p^n(0)$, such as having no customers in the queue at time zero. Over the first interval $[0, t_1]$, assume a constant arrival rate $\lambda^n(t) = \lambda^n(t_1/2)$ and service rate $\mu^n(t) = \mu^n(t_1/2)$ for each state n. Then use a numerical technique to solve the set of differential equations (1) for $p^n(t)$ over the interval $[0, t_1]$. Numerical studies have shown that the fourth- or fifth-order Runge-Kutta [5], [18] method provides a good balance between accuracy and computing time. The state of the system at the end of the first time interval is given by the probability distribution $p^n(t_1)$ and this becomes the boundary condition for the next time interval $[t_1, t_2]$. One then selects new constant arrival and service rates for the new time interval and solves the differential equations for $p^n(t)$ again. This procedure is repeated for each time interval in the time horizon.

From the solution of (1) for the time-dependent state probabilities $p^n(t)$, one can study the nonstationary performance of the system for a given time-varying arrival and service pattern. However, one of the general difficulties in analyzing the time-varying behavior of queueing systems is formulating meaningful performance measures; the standard time-averaged performance measures such as average delay and power cannot be directly used in the nonstationary case. Furthermore, during nonstationary periods many of the standard queueing relationships (e.g., output rate of queue = input rate) do not hold and performance measures must be determined from the state probabilities $p^n(t)$ using basic probability principles. Some examples of possible performance measures are $L(t)$, the expected number of customers in the system at time t, which is given by $L(t) = \Sigma_{n=1}^{K} np^n(t)$, and $D(t)$, the departure rate from the queue at time t, which is determined by $D(t) = \Sigma_{n=1}^{K} \mu^n p^n(t)$.

To illustrate some of the issues discussed above, consider the problem of determining the time varying response of an $M/M/1/20$ queue with ($\lambda^n(t) = \lambda = 0.8$ $\forall n$, $\mu^n(t) = \mu = 1.0$ $\forall n$) and the queue in an initial empty state. Note that after an initial transient the system will attain a steady state. A possible performance measure for this system is $L(t)$, which can be determined using the procedure presented above. In Fig. 1, the average number in the system $L(t)$ is plotted along with the average number in the system as determined from an ensemble average of 50 and 5000 independent simulation runs. One can clearly see that a large number of independent simulation runs must be generated to get an accurate portrayal of the system behavior; hence large amounts of computer run time are required. A detailed treatment of simulation methods for estimating the nonstationary behavior of computer networks can be found in [26].

In the following, we develop a numerical procedure similar to the one given above for a single queue to approximately model a computer network, allowing one to incorporate such features as adaptive routing and finite buffering into the time-varying model. As in the case of the product form queueing network models, we consider a network consisting of M queues connected in an arbitrary topology. We assume that customers arrive at the network according to nonstationary Poisson processes with mean rate $\gamma_i(t)$ at queue i at time t. We denote the finite buffer existing at each queue i by b_i. Focusing on an arbitrary queue i in the network, we assume that it can be approximately modeled as an $M/M/1/b_i$ queue with state-dependent arrival rate $\lambda_i^n(t)$ and service rate $\mu_i^n(t)$, where the state n is the number of packets at the link. Typically, the arrival rate is independent of the state n (i.e., $\lambda_i^n(t) = \lambda_i(t)\ \forall n$); likewise the service rate is independent of the state n and proportional to the link capacity C_i (i.e., $\mu_i^n(t) = \mu C_i\ \forall n$, where $1/\mu$ is the average packet length). Defining $p_i^n(t)$ as the probability of n packets being in the ith queueing system at time t, the differential equation model (1) for queue i becomes

$$\frac{dp_i^0(t)}{dt} = -\lambda_i(t)\,p_i^0(t) + \mu C_i p_i^1(t)$$

$$\frac{dp_i^n(t)}{dt} = \lambda_i(t)\,p_i^{n-1}(t) - [\lambda_i(t) + \mu C_i]\,p_i^n(t) + \mu C_i p_i^{n+1}(t), \qquad 0 < n < b_i$$

$$\frac{dp_i^{b_i}(t)}{dt} = \lambda_i(t)\,p_i^{b_i-1}(t) - \mu C_i p_i^{b_i}(t). \tag{2}$$

Given the time-varying link load $\lambda_i(t)$, one can solve (2) using the numerical technique discussed earlier with the number of equations K fixed at the finite buffer size (i.e., $K = b_i$). To evaluate the performance of the network as a whole, the arrival rate at each link must be known and this can be found in a fashion similar to that for product form queueing networks. Letting $r_{ij}(t)$ represent the time-varying routing probability that a packet at queue i is routed to queue j, then the total arrival rate at queue i at time t is given by

$$\lambda_i(t) = \gamma_i(t) + \sum_{l=1}^{M} \mu C_l(1 - p_l^0(t))\,r_{li}(t) \tag{3}$$

where the first term on the right-hand side represents the arrival rate of external packets to queue i and the second term represents the arrival rate of traffic from other network queues. Note that the output from a queue l is given by the service rate μC_l times the probability that the server is busy $(1 - p_l^0(t))$ since, as previously noted, during nonstationary periods the output rate $\mu C_l(1 - p_l^0(t))$ is not equal to the input rate of the queue $\lambda_l(t)\,(1 - p_l^{b_l}(t))$. In order for the $M/M/1/b_i$ assumption to hold, the link arrival rate $\lambda_i(t)$ must be a nonstationary Poisson process. This assumption can be justified by noting that the superposition of a number of point processes approaches a Poisson process in the limit as the number of point processes increases [19]. Alternately one can justify this assumption based upon a more stringent condition, which is to assume that the departure process of each queue l is approximately a nonstationary Poisson process with mean rate $\mu C_l(1 - p_l^0(t))$. In [20], Taafe has studied the output process of nonstationary finite Markovian systems in detail and shown that the nonstationary Poisson process approximation is reasonably accurate. The assumption made here is not as strong as an independence assumption, as there is clearly a dependence between the queues through (3).

In the same fashion as the method discussed previously for a single queue, one can apply standard numerical integration techniques in an iterative fashion over the time intervals of interest to solve (2) together with (3) for the state probabilities $p_i^n(t)$ at each queue. When steady-state conditions occur, the solution of (2) and (3) by numerical techniques allows the steady-state behavior to be evaluated and the settling time to be determined empirically.

It is interesting to compare the approach proposed above with an exact Markovian analysis. We can define a Markov process $\langle N_1(t), N_2(t), \cdots, N_M(t)\rangle$, where $N_i(t)$ is the number in the system at queue i at time t. In theory, one can develop the Chapman-Kolmogorov equations for the Markov process and then apply numerical integration techniques to solve the equations. The practicality of such an approach is extremely limited owing to the dimensionality of the state space. In fact, an exact analysis will require the solution of $\Pi_{i=1}^{M}(b_i + 1)$ differential equations, whereas for the approximate approach given above one need solve only $\Sigma_{i=1}^{M}(b_i + 1)$ equations. As an example of the computational savings consider the tandem queueing system of Fig. 2(a) with an equal buffer space of 50 at each queue (i.e., $b_i = 50\ \forall i$). An exact analysis will require the solution of 132 651 differential equations versus 153 equations for the approximate approach.

As previously noted, the standard performance measures of average network delay, throughput, and power are not directly useful in evaluating the nonstationary behavior of the network, and alternative performance measures must be considered. Some possible network performance measures are $LN(t)$, the expected number of packets in the network at time t, which is given by

$$LN(t) = \sum_{i=1}^{M}\sum_{n=1}^{b_i} np_i^n(t),$$

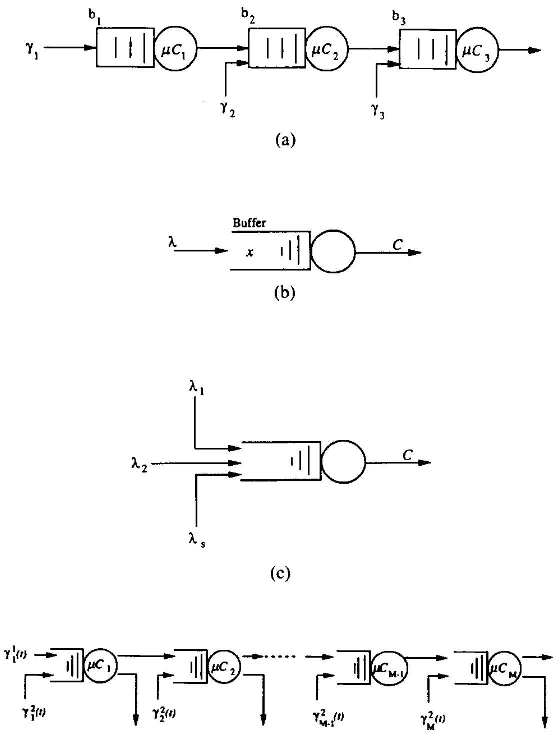
$RR(t)$, the rate at which packets are rejected by the network at time t, which is given by

$$RR(t) = \sum_{i=1}^{M} \lambda_i(t)\,p_i^{b_i}(t),$$

and $TN(t)$, the total network link flow at time t, which can be determined by

$$TN(t) = \sum_{i=1}^{M} \mu C_i(1 - p_i^0(t)).$$

This total network link flow is not the same as the network throughput since packets that are transmitted through one link may be dropped before reaching the destination.



(a)

(b)

(c)

(d)

Fig. 2. Queueing models studied.

These performance measures can be used to evaluate the performance of data networks under nonstationary conditions.

As an illustration of the numerical method and its accuracy, consider the system of three tandem queues shown in Fig. 2(a) with all the queues having the same buffer size of 7 (i.e., $b_i = 7 \; \forall i$) and the same service rate of 1 (i.e., $\mu C_i = 1 \; \forall i$). In order to demonstrate the accuracy of the departure process approximation, the rate of externally arriving packets to queues 2 and 3 was set to 0 (i.e., $\gamma_2(t) = \gamma_3(t) = 0$) and several numerical studies were conducted and contrasted with simulation results. The results of a typical study are shown in Fig. 3, where the average number in the system for each queue (i.e., $L_1(t)$, $L_2(t)$, $L_3(t)$) is plotted along with an equivalent quantity estimated from an ensemble of 5000 independent simulation runs. The input process at the first queue was chosen to be a function which varies between light and heavy load conditions, specifically $\gamma_1(t) = 0.5 + 0.4 \sin(0.2(t + 20))$. Fig. 3 clearly shows that the numerical approach given above provides accurate results. Furthermore, the numerical approach presented here is considerably more computationally efficient than the comparable simulation. For example, typical run times for a simulation study like the one in Fig. 3 coded in SLAM required approximately 66 min and 40 s of CPU time on a Sun IV workstation, whereas the numerical integration approach implemented using the fourth-order Runge–Kutta routine in MATLAB required only 20 min on a personal computer equivalent to an IBM PC AT. Additional performance evaluation studies have been conducted which follow the present approach and these may be found in [6], [21]. In these works, a comparative nonstationary performance evaluation of different buffer management schemes implemented in a network subjected to dynamically varying load conditions is reported.

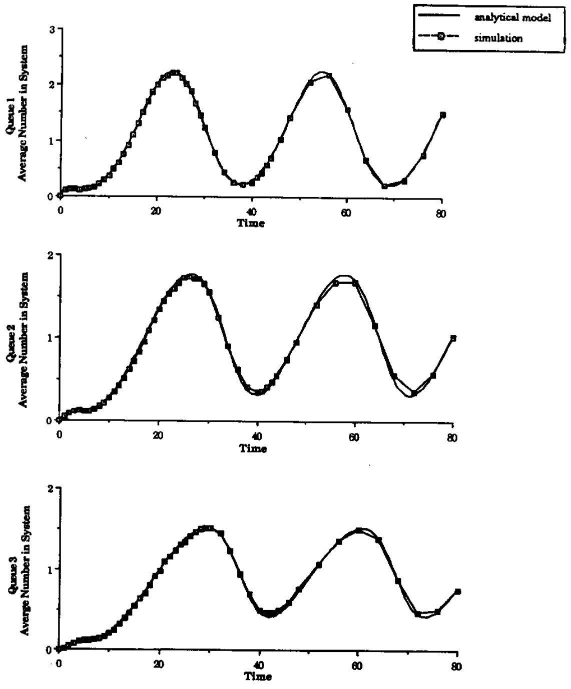


Fig. 3. Nonstationary behavior of tandem queueing system.

III. A State Model Representation of Queue Dynamics

As described in Section II, the time-varying behavior of a computer network can be analyzed by solving, for each queue, the set of differential equations describing the queue length probability distributions. While this approach, as shown earlier, is particularly useful for a numerical evaluation of the network performance during both nonstationary and steady-state periods, one is limited by the computational complexity to considering small systems. Also, obtaining analytical expressions for network performance measures that may be used in an optimization problem to design efficient control algorithms is difficult. Here we note that routing and flow control procedures are commonly based on optimization and feedback of average quantities, such as the average number of packets in the queues or the average delay on the links, and it is difficult to determine the time-varying behavior of such mean quantities from queueing models [12]. In this section, we develop an alternative approach that characterizes the dynamics of the network by a set of nonlinear differential equations describing the time-varying behavior of the mean queue lengths at the various network queues. This approach offers the advantage of a considerable reduction of computational time in performance evaluations. It also establishes a framework for the formulation of precise control problems with different performance objectives, allowing the rich theory of optimal control to be employed in the determination of network control algorithms. The model is developed by focusing on the dynamics of the packet queues at the transmission links in the network. Because of the similarity of this model to the state variable models popularly used in modern control theory, we shall refer to it as a state model.

A. A State Model for a Single Communication Link

Consider an arbitrary transmission link in a computer network of the type shown in Fig. 2(b). We define C as the capacity of the link, $N(t)$ as the number in the system (i.e., queue + server) at time t, and $x(t)$ as the state variable representing the average number in the system at time t. Note that the state variable is the *ensemble average* of the number in the system at time t (i.e., $x(t) = E\{N(t)\}$). Let $d(t)$ represent the flow out of the system at time t, and $a(t)$ the flow into the system at time t. Defining $f_{\text{out}}(t)$ and $f_{\text{in}}(t)$ as the ensemble average of flow out and flow in of the queue, respectively (i.e., $f_{\text{out}}(t) = E\{d(t)\}$ and $f_{\text{in}}(t) = E\{a(t)\}$), then from the flow conservation principle, the rate of change of the state variable can be related to the ensemble average flow in and flow out by a differential equation of the form

$$\dot{x}(t) = -f_{\text{out}}(t) + f_{\text{in}}(t). \tag{4}$$

This equation is intuitive in nature and can be found in several places in the literature [7], [8], [22], [23] in various forms and is often called a fluid flow equation. Note that the approach taken here is different from the commonly used *fluid flow approximations*, which are developed to approximate the time-varying behavior of $N(t)$ [24]. It should be emphasized that (4) is quite general in nature and can be used to model a wide range of queueing and contention systems.

Assuming that the queue storage capacity is unlimited (i.e., $x \in [0, \infty)$, and that customers arrive at the queue according to a nonstationary Poisson process with rate $\lambda(t)$, then $f_{\text{in}}(t)$ is just the offered load $\lambda(t)$ since no packets are dropped. The flow out of the system, $f_{\text{out}}(t)$, can be related to the ensemble average utilization of the link $\rho(t)$ by $f_{\text{out}}(t) = C\rho(t)$. Note that $\rho(t) = P(N(t) > 0) = \Sigma_{i \geq 1} P(N(t) = i) = (1 - P(N(t) = 0))$. We assume that $\rho(t)$ can be approximated by a function $G(x(t))$, which represents the ensemble average utilization of the link at time t as a function of the state variable. Since $x(t) = 0 \Rightarrow f_{\text{out}}(t) = 0$, $G(0) = 0$ and since $x(t) = \infty \Rightarrow f_{\text{out}}(t) = C$, $G(\infty) = 1$. Thus, in order to model the physical system, $G(x(t))$ must range over $x(t) \in [0, \infty)$ with values belonging to $[0, 1)$ and pass through the origin, i.e., $G(0) = 0$. Furthermore, to show the effects of congestion, $G(x(t))$ must be a nonnegative, strictly concave function. Thus, the dynamics of the link queue can be represented by a nonlinear differential equation:

$$\dot{x}(t) = -CG(x(t)) + \lambda(t) \tag{5}$$

with initial condition $x(0) = x_0$.

This type of model has been proposed by several researchers [7], [23] to describe the dynamic behavior of queueing systems in terms of time-varying mean quantities. The exact form of the utilization function $G(x(t))$ which will accurately model the system will depend on the queue under study and the data available. If experimental data from an existing system can be obtained, then the function can be determined statistically [22]. However, such data are normally unavailable and one must determine $G(x(t))$ by other means, such as the technique suggested by Agnew [22], which requires matching the steady-state equilibrium point of (5) with that of the equivalent queueing theory model representation. Making the standard product from queueing network assumption that the packet transmission time is proportional to the packet length and that the packets are exponentially distributed in length with mean length $1/\mu$ [1], then the link is modeled as an $M/M/1$ queue. Note that when the arrival rate to the queue is constant (i.e., $\lambda(t) = \lambda \ \forall t$) the average number in the system at steady state is given by $\lambda/(\mu C - \lambda)$ from the $M/M/1$ queueing formulas [1]. Thus requiring that $x(t) = \lambda/(\mu C - \lambda)$ when $\dot{x}(t) = 0 \Rightarrow G(x(t)) = \lambda(t)/C$ results in $G(x(t)) = \mu[x(t)/(1 + x(t))]$ and the state model becomes

$$\dot{x}(t) = -\mu C\left(\frac{x(t)}{1 + x(t)}\right) + \lambda(t) \tag{6}$$

with initial condition $x(0) = x_0$. The validity of matching the steady-state equilibrium point of the state model with that of the queueing model has been checked using simulation by Filipiak [7] and Rider [23] and shown to lead to an accurate approximation to the time-varying mean number in the system.

Obviously one can use the numerical integration approach given in the previous section for a single queue to solve the state model for the average number in the system (i.e., $x(t)$) as a function of time. In order to check the accuracy of the state model we use the technique proposed in [17] of approximating the $M/M/1$ queue by a $M/M/1/K$ queue with K large enough that the probability of blocking is negligible. One can then use the method of Section II to numerically solve the Chapman–Kolmogorov equation (1) for the nonstationary behavior of the $M/M/1/K$ queue and use this as a benchmark. For example, consider the queueing system of Fig. 2(b) when the link capacity is 1 (i.e., $C = 1$), the average packet length is 1 (i.e., $\mu = 1$), and the arrival rate is given by $\lambda(t) = 0.6$ for $t < 40$ and $\lambda(t) = 0.2$ for $t \geq 40$. In Fig. 4 the results of integrating the state model for $x(t)$ are shown along with the average number in the system $L(t)$ as determined from an $M/M/1/40$ queue. One can see that the state model is reasonably accurate, and the

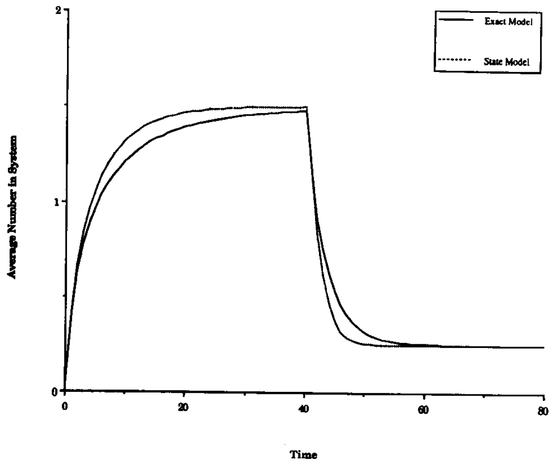


Fig. 4. Comparison of the $M/M/1$ state model with solution of Chapman-Kolmogorov equations.

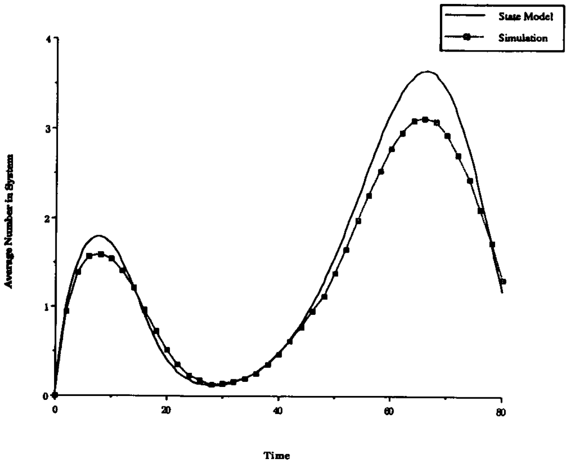


Fig. 5. Comparison of $M/M/1$ state model with simulation.

computational advantages of solving a single differential equation are obvious. Several additional numerical studies were conducted on the same system to compare the accuracy of the state model with a benchmark solution and the results of a typical study given in Fig. 5. In generating Fig. 5, the arrival rate was determined by $\lambda(t) = 0.5 + 0.4 \sin(0.1(t + 20))$ and the comparison curve is the result of an ensemble average of 5000 simulation runs. Simulation was used here to determine the benchmark since at heavy loads K must be very large for the $M/M/1/K$ approximation technique to be accurate. From our numerical studies we conclude that the state model always produces the same form of response as the benchmark solution (i.e., curves have the same shape) but consistently overshoots the magnitude of peaks and valleys in the response.

Note that, in deriving the state model (5), the only stochastic model assumption made is that of Poisson arrivals and thus (5) holds for queues of the $M/G/1$ type. In order to obtain a closed expression for $G(x(t))$ from the technique of matching the steady-state equilibrium points, the steady-state average number in the system for the queueing model must only be a function ρ, the steady-state utilization of the queue. As a second example of the approach, consider an $M/D/1$ queue. Under steady-state conditions (i.e., $\dot{x}(t) = 0$) requiring that $x(t) = \rho + \rho^2/[2(1 - \rho)]$, where $\rho = \lambda(t)/\mu C$, results in the state model

$$\dot{x}(t) = -\mu C\left(x(t) + 1 - \sqrt{x(t)^2 + 1}\right) + \lambda(t). \quad (7)$$

As a simple illustration of the accuracy of the approximation technique, consider the queueing model of Fig. 2(b) when the capacity of the link is 1 (i.e., $C = 1$) and the packets have a constant length of 1 (i.e., $\mu = 1$). The results of a typical numerical study are shown in Fig. 6, where the arrival process was allowed to vary between light and heavy load conditions, specifically $\lambda(t) = 0.5 + 0.4 \sin(0.2(t + 20))$. In Fig. 6, the curve labeled state model was determined by integrating (7) for the state variable using the Runge-Kutta routine in MATLAB and is plotted along with the average number in the system determined from an ensemble average of 5000 simulation runs. Clearly the model is reasonably accurate and the computational advantages of solving a single differential equation over simulation are evident.

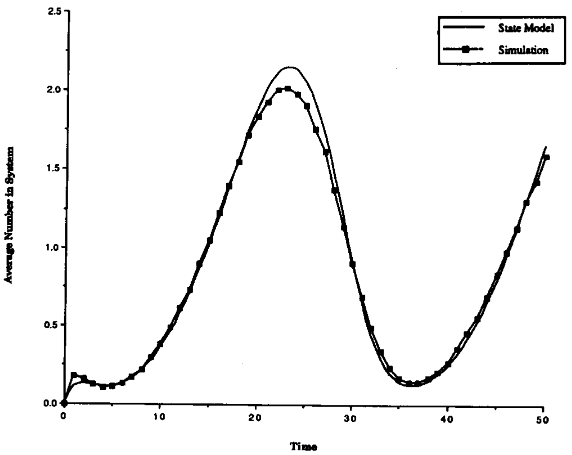


Fig. 6. Comparison of the $M/D/1$ state model with simulation.

In computer networks, the traffic in the network is normally divided into a number of classes, and the control actions (i.e., routing and flow control) are based on the class type. Since the traffic is already grouped into classes, it would be advantageous for performance evaluation and control purposes if the state model could be modified to represent the dynamic behavior of each class separately. Here we show that this can be accomplished by modeling the link by a set of coupled state equations where there is a state defined for each class of traffic.

Consider an arbitrary transmission link of a network shown in Fig. 2(c). There are S different classes of packets arriving at the link with the average arrival rates $\lambda_1(t)$, $\lambda_2(t), \cdots, \lambda_S(t)$. Let $x_l(t)$ represent the ensemble average number of packets of class l in the system at time t. Making the standard assumptions [2] so that the link can be modeled as an $M/M/1$ queue, then the model (6) describing the average total number of packets in the system $x_T(t) = \Sigma_{l=1}^{S} x_l(t)$ will become

$$\dot{x}_T(t) = -\mu C\left(\frac{x_T(t)}{1 + x_T(t)}\right) + (\lambda_1(t) + \lambda_2(t) + \cdots + \lambda_S(t)). \tag{8}$$

Now, since the flow conservation will hold for each class of packets, a state model of the form of (5) can be developed for each class as

$$\dot{x}_l(t) = -CG_l(x_l(t), x_T(t)) + \lambda_l(t) \qquad \forall l = 1, 2, \cdots, S \tag{9}$$

where $G_l(x_l(t), x_T(t))$ represents the average utilization of the link by the class l traffic. Note that if there are only class l packets present in the link (i.e., $x_j(t) = 0$, $\lambda_j(t) = 0$, $\forall j, j \neq l$, $x_T(t) = x_l(t)$), then G_l will be a function of the class l packets $x_l(t)$ only and must have the form of the utilization function in state model (6) (i.e., $G_l = \mu C x_l(t)/(1 + x_l(t))$) since it will represent the dynamics of an $M/M/1$ queue with only one class of traffic. However, if additional classes of traffic are also present in the link, they will use part of the transmission capacity of the link, and the portion of link capacity seen by the class l packets will depend on the total amount of link capacity being used. Thus G_l will be a function of both the average number of class l packets in the system and the average total number of packets in the system. Determination of G_l can be done by using the technique of matching the steady-state equilibrium points of the state model and the equivalent stochastic model. Then from the $M/M/1$ queueing model of the link with S classes of customers, when the arrival rates are constant (i.e., $\lambda_j(t) = \lambda_j \; \forall t$ and $\forall j$) a steady state will result and we require that

$$x_l(t) = \frac{\lambda_l}{\mu C - \sum_{j=1}^{S} \lambda_j} \qquad \forall l. \tag{10}$$

Now, at steady state $\dot{x}_l(t) = 0 \Rightarrow CG_l(x_l(t), x_T(t)) = \lambda_l \; \forall l$ and $\dot{x}_T(t) = 0 \Rightarrow \mu C(x_T(t)/1 + x_T(t)) = \Sigma_{j=1}^{S} \lambda_j(t)$. Hence, substituting in (10) and solving for G_l results in

$$G_l(x_l(t), x_T(t)) = \mu\left(\frac{x_l(t)}{1 + x_T(t)}\right) = \mu\left(\frac{x_l(t)}{1 + \sum_{j=1}^{S} x_j(t)}\right) \qquad \forall l$$

and the state model for the transmission link now becomes

$$\dot{x}_l(t) = -\mu C\left(\frac{x_l(t)}{1 + \sum_{j=1}^{S} x_j(t)}\right) + \lambda_l(t), \qquad l = 1, 2, \cdots, S. \tag{11}$$

Note that since $x_T(t) = \Sigma_{l=1}^{S} x_l(t)$, we have $\dot{x}_T(t) = \Sigma_{l=1}^{S} \dot{x}_l(t)$ and substitution of (11) into this expression results in (8), as expected. Thus the link can be described by the set of S state equations of the form (11) representing the behavior of each class of traffic separately. One can use an approach similar to developing (11) to derive a state model for the $M/D/1$ queue with multiple classes of traffic.

In order to illustrate the advantages and quantify the accuracy of the state model developed here, several numerical studies were conducted to compare the state model (11) with a simulation of the same system. A typical comparison for a queue with two classes of traffic (i.e., $S = 2$) is shown in Fig. 7, where step loads of $\lambda_1 = 0.5$ and $\lambda_2 = 0.2$ are applied to an initially empty queue with service rate $\mu C = 1$. The evolution of the queue depicted by the state model was determined by numerical integration of (11), whereas the simulation curves were generated by averaging 5000 independent simulation runs. Obviously, the state model is sufficiently accurate and saves considerable computation. A relative idea of the magnitude of the computational savings can be seen by noting that the simulation which was conducted in SLAM required 5 h, 35 min, and 42 s of actual CPU execution time on a VAX 11/750 system, whereas the numerical solution of the state model using fifth-order Runge–Kutta technique coded in Fortran required only 13 s of CPU execution time on the same system.

In comparison with the analytical methods described in Section II, which are based on the Chapman–Kolmogorov differential equations, it is to be noted that the state model offers several advantages. The most obvious are the considerable reduction in computational complexity and the ability to model a wider range of queueing systems. Another strong feature is the simplicity in modeling queueing systems with multiple classes of traffic when the behavior of each class is to be examined separately.

B. Development of State Models for Networks

The state model (6) represents the dynamics of a single link and can be extended to describe the time-varying behavior of a computer network. Consider a network consisting of M queues interconnected in an arbitrary topology. The traffic is assumed to arrive from outside the network at each queue i according to a nonstationary Poisson process with rate $\gamma_i(t)$. Let us define $r_{ij}(t)$ as the routing probability that the traffic at queue i is routed to queue j. Also, let μC_i denote the exponential service rate of queue i, $\lambda_i(t)$ represent the average arrival rate to queue i, and $x_i(t)$ be the state variable denoting the average number in the system at queue i. Under these assump-

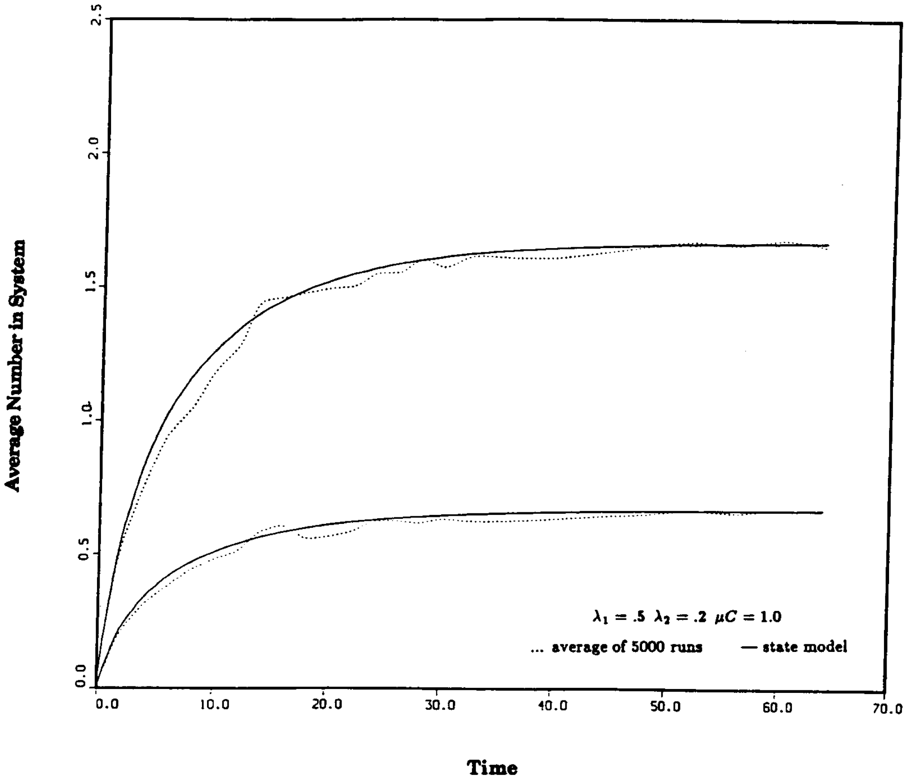


Fig. 7. Comparison of the state model with multiple classes with simulation.

tions, one can readily obtain the state variable model for each queue i in the network as

$$\dot{x}_i(t) = -\mu C_i\left(\frac{x_i(t)}{1 + x_i(t)}\right) + \lambda_i(t). \qquad (12)$$

The queue arrival rate, $\lambda_i(t)$, can be determined as

$$\lambda_i(t) = \gamma_i(t) + \sum_{l=1}^{M} \mu C_l\left(\frac{x_l(t)}{1 + x_l(t)}\right) r_{li}(t). \qquad (13)$$

Note that the term summed represents the arrival rate of traffic from other queues to link i. Unlike the finite queue case discussed in Section II, the output process of an $M/M/1$ queue is a Poisson process and the only approximation here is the one used in determining $G(x(t))$. One can easily show that under steady-state conditions, the state model defined by (12) and (13) yields exactly the same average number in the system at the network queues as the corresponding steady-state product form queueing network. The fact that the state model above, (12), approximately describes the time-varying behavior of the average number in the system at the various queues in the network is of considerable interest for performance evaluation. One can easily conduct quick network performance studies by numerically integrating the state model (12) along with (13) over the time interval of interest utilizing the approach discussed in Section II. Note that the number of differential equations to be solved is only M, allowing reasonably large networks to be studied. As a simple illustration of the use of the model, consider the tandem queueing model shown in Fig. 2(a) for the case where the buffer space is unlimited, each link has a capacity of 1 (i.e., $C_i = 1 \; \forall i$), the packet length is 1 (i.e., $\mu = 1$), and the arrival pattern of external traffic is $\gamma_1 = 0.5$, $\gamma_2 = \gamma_3 = 0$. In Fig. 8 the time-varying behavior of the state variable for each queue is plotted along with the average number in the system at each queue as estimated from an ensemble average of 5000 simulation runs. One can see that the state model is reasonably accurate. The computational advantages of the state model approach are significant, as the simulation coded in SLAM requires of the order of 66.67 min of CPU time on a SUN IV workstation, whereas the state model can be solved using the Runge–Kutta routines in MATLAB on a personal computer equivalent to an IBM PC AT in approximately 1 min of CPU time. Note that one can approximately estimate the settling time of the network (time until last queue attains steady state) from the numerical solution of the state model.

In a fashion similar to that above, one can use the state model defined by (11) to model a computer network where the traffic is split into distinct classes. Consider a network consisting of M queues interconnected in an arbitrary topology in which S classes of traffic exist. Each class of

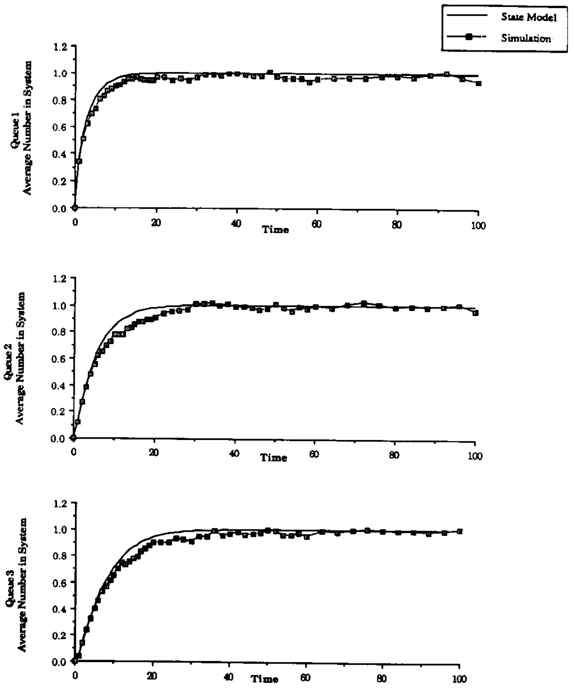


Fig. 8. Comparison of state model with simulation for tandem queueing model.

traffic, v, is assumed to arrive from outside the network at each queue i according to a nonstationary Poisson process with rate $\gamma_i^v(t)$. We define $x_i^v(t)$ as the state variable denoting the average number of class v in the system at queue i at time t, and $r_{ij}^v(t)$ as the routing probability that the v traffic at queue i is routed to queue j at time t. Also, let μC_i denote the exponential service rate of queue i, and $\lambda_i^v(t)$ represent the average arrival rate of class v traffic to queue i at time t. Under these assumptions, one can readily obtain the state variable model for each queue i in the network as

$$\dot{x}_i^v(t) = -\mu C_i \left(\frac{x_i^v(t)}{1 + \sum_{v=1}^{S} x_i^v(t)} \right) + \lambda_i^v(t),$$

$$v = 1, 2, \cdots, S. \tag{14}$$

The queue arrival rate, $\lambda_i^v(t)$, can be determined as

$$\lambda_i^v(t) = \gamma_i^v(t) + \sum_{l=1}^{M} \mu C_l \left(\frac{x_l^v(t)}{1 + \sum_{v=1}^{S} x_l(t)} \right) r_{li}^v(t),$$

$$v = 1, 2, \cdots, S. \tag{15}$$

Note that the term summed the arrival rate of class v traffic from other queues to link i. As before, one can conduct network performance studies by numerically integrating the state model (14) along with (15) over the time interval of interest utilizing the approach discussed in Section II. Note that the number of differential equations to be solved is only SM. Also notice that the routing variables can be defined to represent source-destination pair routing variables, in contrast to the routing variables of model (12) and (13), which represent the portion of total link flow routed to a particular queue.

As an example of the use of the system consider the tandem queueing model shown in Fig. 2(d), which is often used to represent a single source-destination path in a computer network. The traffic traversing all of the queues could represent a virtual circuit session and the traffic entering and exiting at each queue would represent the background traffic from other sessions in the network. Consider the case where there are three queues in the model (i.e., $M = 3$), each link has a capacity of 1 (i.e., $C_i = 1 \; \forall i$) and the packet length is 1 (i.e., $\mu = 1$). We denote the traffic which is traversing all three queues as class 1 traffic, and the background traffic at each queue is denoted as class 2 traffic. From Fig. 2(d) we have that $r_{12} = r_{23} = 1$ and all other routing variables are 0; and thus $\lambda_i^v(t)$ can be determined from (15) for a given arrival pattern. Several different numerical studies of this system have been conducted and compared with simulation studies. The results of two typical studies are given in Figs. 9 and 10. For Fig. 9 the arrival pattern of external traffic is $\gamma_1^1(t) = 0.2$, $\gamma_1^2(t) = 0.4$, $\gamma_2^1(t) = 0$, $\gamma_2^2(t) = 0.5$, $\gamma_3^1(t) = 0$, $\gamma_3^2(t) = 0.6$. For Fig. 10 the external arrival pattern is given by $\gamma_1^1(t) = [29 \cos (0.1(t + 20)) + 31]/60$, $\gamma_1^2(t) = [29 \sin (0.05(t + 20) + \pi) + 31]/60$, $\gamma_2^1(t) = 0$, $\gamma_2^2(t) = [29 \cos (0.05(t + 20)) + 31]/60$, $\gamma_3^1(t) = 0$, $\gamma_3^2(t) = [29 \cos (0.1(t + 20)) + 31]/60$. The simulation results shown in Figs. 9 and 10 are the result of averaging the number of each type of traffic in the system over 5000 simulation runs. From these figures it can be seen that the shape of the response is accurately reproduced, but the state model overshoots and undershoots peaks and valleys in the response. However the state model is accurate enough to allow one to determine possible bottlenecks and its computational simplicity makes it a valuable tool in obtaining a rough idea of the nonstationary behavior of large networks. A detailed treatment of the simulation methods employed here can be found in [26].

Another possible application of the state model is as a predictor of the queue lengths in a fashion similar to that proposed by Stern [9] and can be explained as follows. Adaptive routing algorithms typically use a periodic monitoring and collection of network status information to adaptively select the best routes. This normally involves each node measuring its status information (e.g., queue lengths or link delays) and disseminating this information through the network on a node by node exchange basis or transmitting the information to a network control center. This exchange of status information can introduce considerable communication and processing overhead and it is desirable to transmit this information infrequently. Here we note that the state model allows us to reduce the frequency of exchange of status information and to make intelligent use of the status information between update pe-

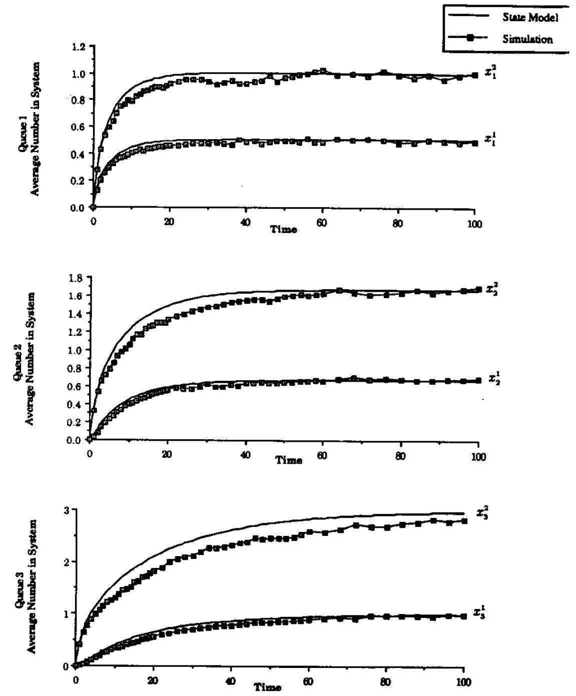


Fig. 9. Comparison of the state model with multiple classes with simulation for a tandem queueing system.

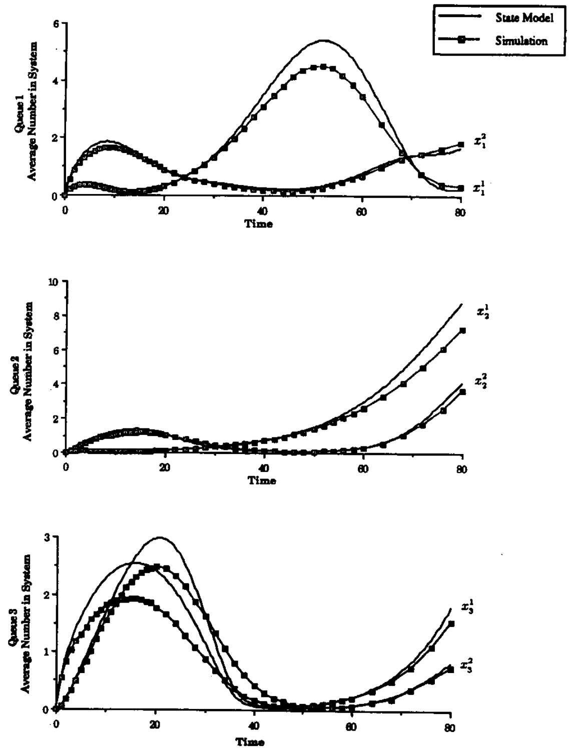


Fig. 10. Typical nonstationary behavior of the state model with multiple classes.

riods. Specifically, the nodes can periodically measure the number in the system at the queues and the arrival rates on their links and exchange this information in the proper fashion. Between updates the nodes can predict the behavior of the network queues by numerically integrating (12) or (14) to solve for the state (average number in system) using the most recent status information as initial values for the state variables and the link arrival rates (i.e., the forcing functions).

C. A Framework for the Application of Optimal Control Methods

In this subsection, we discuss certain properties that establish a framework for the use of the model along with optimal control theory in the design of optimal control strategies. A more detailed discussion of such issues as stability, settling time, and step response can be found in [21], and specific examples of routing strategies designed on the basis of (5) are given in [7], [21], and [25].

1) Establishment of Performance Measures: A principal advantage of the state model approach is the flexibility it affords in establishing various performance measures that can be used further in the design process for an optimal selection of network parameters. Nonstationary performance measures of delay and throughput can be easily established in a form appropriate to use in conjunction with the state model (6) for formulating optimal control problems. The total mean throughput of a queue during a time interval $[t_0, t_f]$ can be defined in two ways—either as the total mean flow into the queue during the time interval or as the total mean flow out. Since no traffic is dropped, the total mean throughput into the queue is just the total amount of traffic arriving at the link over the time interval $[t_0, t_f]$ and can be represented by

$$J_{T_{\text{in}}} = \int_{t0}^{tf} \lambda(t)\, dt. \tag{16}$$

On the other hand the total mean flow out of the queue during $[t_0, t_f]$ is given by

$$J_{T_{\text{out}}} = \int_{t0}^{tf} \mu C\left(\frac{x(t)}{1 + x(t)}\right). \tag{17}$$

The delay over the time interval $[t_0, t_f]$ can be approximately estimated from Little's formula [1], which in terms of the state variable $x(t)$ becomes $x(t) = \lambda(t)D$, where D is the average packet delay on the link. Thus the total average delay seen by packets on the link is given by

$$J_D = \int_{t0}^{tf} \frac{x(t)}{\lambda(t)}\, dt. \tag{18}$$

Note that Little's law only holds under steady-state conditions and (18) may be inappropriate for nonstationary periods. However an alternative measure of the delay can be formulated by assuming that the cost of waiting in the system is proportional to the number of customers in the system, as in [7]. Hence, minimizing the number of customers is equivalent to minimizing the delay, and the performance measure is determined as

$$J_{D2} = \int_{t0}^{tf} x(t)\, dt. \tag{19}$$

Normally, minimizing the delay is chosen as the objective in designing routing algorithms, while flow control is selected to maximize the throughput [1], [2]. In recognition of the conflicting nature of delay and throughput and the realization that the consideration of a single performance measure to the exclusion of the other can lead to poor performance, alternative performance measures have been proposed. The most widely recognized of these alternative indices is Kleinrock's combined performance measure of power [28], which is defined as the ratio of throughput to delay. In terms of the present state variable, the power, P, can be estimated by $P = \lambda / D = \lambda^2 / x$ and the total power of the link is represented by

$$J_P = \int_{t0}^{tf} \frac{\lambda^2(t)}{x(t)}\, dt. \tag{20}$$

The control objective in this case is to maximize the power J_P. Another important performance measure which has been used extensively in the operations research literature [29] is to optimize the average net benefit or cost of operating the system. If we associate a cost or loss with holding customers in the system and a reward or profit for each customer served by the system, then a linear net benefit performance criterion can be formulated as the difference between the total holding cost and the total reward. The optimization of the linear net benefit criterion has been shown to result in social optimization of the queueing system rather than individual optimization [29]. In terms of the state variable used here, the linear net benefit can be represented by

$$J_{NB} = \int_{t0}^{tf} (\omega x(t) - \tilde{\omega}\lambda(t))\, dt \tag{21}$$

where $\omega (0 < \omega \leq 1)$ is a weighting constant representing the holding cost per unit time per packet in the system, and similarly $\tilde{\omega}(0 < \tilde{\omega} \leq 1)$ is a weighting constant reflecting the reward for each packet successfully transmitted on the link. In the context of computer networks, the performance criterion (21) can be interpreted as simultaneously optimizing delay and throughput, with ω and $\tilde{\omega}$ being weighting constants reflecting the relative importance of delay and throughput.

Several other performance indexes similar to the ones discussed can be proposed [21]. Suitable network performance measures can be formulated by summing one of the performance indexes for a single link discussed above over all the queues in the network. Note that the performance measures discussed above are time-varying quantities and with the proper specifications of the limits of integration, they can be used to measure the performance of the queue over any time period of interest. Lastly, the integrands in the performance measures above represent instantaneous performance measures similar to those discussed in Section II-B, and under stationary conditions these quantities will correspond to the appropriate time average measures.

2) An Illustration of the Application of Optimal Control: As mentioned earlier, an advantage of the state model representation of the queue dynamics is the possibility of employing the mathematical framework of optimal control theory in the design of network control strategies. To provide specific examples to illustrate this approach, let us consider the following problem of controlling $\lambda(t)$, the arrival rate of packets to a transmission link, in order to optimize a specified performance measure. The problem can be formulated in terms of the state model (6) as follows.

Problem P: Optimize J such that

$$\text{(C1)} \qquad \dot{x}(t) = -\mu C\left(\frac{x(t)}{1 + x(t)}\right) + \lambda(t)$$

$$\text{(C2)} \qquad \lambda(t) \geq 0.$$

In this problem, (C1) and (C2) are the constraints under which optimization with respect to $\lambda(t)$ is to be performed, J can be any of the performance measures defined earlier, and by *optimize* we mean either minimization or maximization, depending on the specific performance measure considered. Note that in contrast to the available literature on the optimal control of queueing systems [29], which are based on steady-state queueing models, an assumption of steady-state conditions is not necessary.

As a simple illustration of the general solution technique for Problem P, consider the problem of determining the optimal arrival rate, $\lambda^*(t)$, that maximizes the linear net benefit J_{NB} given by (21). For the solution, one may employ the standard Hamilton-Jacobi technique [27], forming the Hamiltonian $\mathcal{H}$, which is given by

$$\mathcal{H} = \omega x(t) - \tilde{\omega}\lambda(t) + p(t)\left[-\mu C\left(\frac{x(t)}{1 + x(t)}\right) + \lambda(t)\right] \tag{22}$$

where $p(t)$ is the costate variable and is determined by

$$\dot{p}(t) = -\frac{\partial \mathcal{H}}{\partial x} = -\omega + p(t)\mu C\left(\frac{1}{1 + x(t)}\right)^2. \tag{23}$$

Using Pontryagin's minimum principle, the optimal control $\lambda^*(t)$ is determined by minimizing $\mathcal{H}$ with respect to $\lambda(t)$. Thus, setting $\partial\mathcal{H}/\partial\lambda = 0$, one obtains $p(t) = \tilde{\omega}$. This is the optimal value of the costate variable $p^*(t)$, which is a constant. Hence, under the conditions of optimality, $\dot{p}(t) = 0$. Now setting the right-hand side of (23) to 0 and solving for $x(t)$, the optimum value of the state variable can be determined as

$$x^*(t) = \frac{\sqrt{\tilde{\omega}\mu C}}{\omega} - 1. \tag{24}$$

Since $x^*(t)$ is a constant, $\dot{x}^*(t) = 0$; hence setting the right-hand side of constraint equation (C1) to zero, one can readily determine the optimal arrival rate, $\lambda^*(t)$, as

$$\lambda^*(t) = \mu C\left(\frac{x^*(t)}{1 + x^*(t)}\right) = \mu C - \frac{\sqrt{\omega \mu C}}{\tilde{\omega}}. \quad (25)$$

It may be noted that $\lambda^*(t) < \mu C \Rightarrow \rho^* < 1$; hence, the queue is stable and will attain a steady-state equilibrium condition. Also, to satisfy constraint (C2), we note from (25) that $\lambda^*(t) \geq 0$ if $\tilde{\omega}\mu C \geq \omega$. It is easy to see that when $\tilde{\omega}\mu C < \omega$, the system is unprofitable at any positive arrival rate since the holding cost ω is greater than the average reward $\tilde{\omega}\mu C$. Thus we require that $\lambda^*(t) = 0$ if $\tilde{\omega}\mu C < \omega$ and $\lambda^*(t)$ is given by (25) if $\tilde{\omega}\mu C \geq \omega$. It is interesting to note that the optimal arrival rate is time-invariant, resulting in steady-state conditions after an initial transient period and the optimal control in this case is an open-loop control. Furthermore, the value of $\lambda^*(t)$ determined here agrees with the socially optimal arrival rate for an $M/M/1$ queue determined by Stidham [29] using queuing theory methods.

As a second illustration, the solution of the problem of determining the optimal arrival rate, $\lambda^*(t)$, for maximizing the power J_P, given by (20), can be obtained following the same procedure as

$$\lambda^*(t) = \frac{p^*(t)x^*(t)}{2} \quad (26)$$

where the evolutions of $p^*(t)$, the optimal costate, and $x^*(t)$, the optimal state, are governed by

$$\dot{x}^*(t) = -\mu C\left(\frac{x^*(t)}{1 + x^*(t)}\right) + \frac{p^*(t)x^*(t)}{2}$$

$$\dot{p}^*(t) = -\frac{p^{*2}(t)}{4} + p^*(t)\mu C\left(\frac{1}{1 + x^*(t)}\right)^2. \quad (27)$$

Details of these computations may be found in [21]. Note that the optimal arrival rate defined by (26) is a feedback control law and can be determined by numerically solving (27) with appropriate boundary conditions. Actual implementation of the resulting control would entail estimation of the state variable $x(t)$, which can be determined by a number of Kalman filtering techniques [30], [31]. For comparison with certain known steady-state results for this problem, observe that setting $\dot{x}(t) = 0$ and $\dot{p}(t) = 0$ will yield

$$x^*(t) = 1 \quad \text{and} \quad \lambda^*(t) = \mu C/2, \quad (28)$$

i.e., the optimal arrival rate should equal half the service rate, and power is maximized when the average number of customers in the system is 1. These values precisely agree with those computed by Kleinrock [28] using a queueing theory model under stationary conditions.

Following the solution technique outlined above, the optimal arrival rate, $\lambda^*(t)$, can be determined for the other performance measures in a similar fashion [21]. A specific example of using the network state model for multiple classes of traffic (14) in the design of a virtual circuit routing algorithm can be found [21] and [25].

IV. Conclusions

In this paper we have developed modeling techniques appropriate for conducting performance evaluation studies and designing control strategies for computer networks under nonstationary conditions. This work has been motivated by the observation that the majority of currently available analytical and simulation techniques for studying computer networks are valid only under steady-state operating conditions. However, owing to a variety of nonstationary phenomena in operating networks, nonstationary or transient conditions prevail during considerable periods of time. This paper offers two distinct approaches, with complementary capabilities, for accurately modeling the network behavior under both nonstationary and steady-state conditions. The first approach, based on a queueing theory formulation, uses the fundamental Chapman–Kolmogorov equations of Markov processes for determining the probability distribution of the number of packets at a queue and develops a method for numerical evaluation of performance measures of the nonstationary queue behavior. In the second approach, a nonlinear state model for representing the dynamics of the packet queues at the various transmission links in terms of time-varying mean quantities is developed. The principal advantages offered by this approach are a considerable reduction in the computational burden, the ability to obtain simple expressions for different network performance measures, and ease in formulation of precise optimal control problems for designing routing and flow control strategies that ensure optimal performance under both nonstationary and steady-state operating conditions. The two approaches taken together allow the modeling of nonstationary queue behavior in computer networks for purposes of performance evaluation and controller design with different degrees of accuracy and detail.

Acknowledgment

The authors are very grateful to C. Bagwell for her help in conducting the simulation experiments and preparing the plots. Also, they are indebted to the anonymous reviewers for their valuable comments.

References

[1] M. Schwartz, *Telecommunication Networks: Protocols, Modeling and Analysis.* Reading, MA: Addison-Wesley, 1987.

[2] D. Bertsekas and R. Gallager, *Data Networks.* Englewood Cliffs, NJ: Prentice-Hall, 1987.

[3] T. Stern, "Approximations of queue dynamics and their applications to adaptive routing in computer communication networks," *IEEE Trans. Commun.*, vol. COM-27, pp. 1331–1335, Sept. 1979.

[4] J. Hammond and J. Spragins, "Rapidly reconfiguring computer communication networks: Definition and major issues," in *Proc. IEEE INFOCOM 87*, San Francisco, CA, 1987, pp. 202–206.

[5] H. Van As, "Transient analysis of Markovian queueing systems and its application to congestion control modeling," *IEEE J. Select. Areas Commun.*, vol. SAC-4, no. 6, pp. 891–904, Sept. 1986.

[6] D. Tipper and M. K. Sundareshan, "Adaptive policies for optimal buffer management in dynamic load environments," in *Proc. IEEE INFOCOM'88*, New Orleans, LA, Mar. 1988, pp. 535–544.

[7] J. Filipiak, *Modeling and Control of Dynamic Flows in Communication Networks.* New York: Springer-Verlag, 1988.

[8] A. Weiss and D. Mitra, "A transient analysis of a data network with a processor sharing switch," *AT&T Tech. J.*, pp. 4-16, Sept./Oct. 1988.

[9] D. Mitra and A. Weiss, "The transient behavior in Erlang's model for large trunk groups and various traffic conditions," in *Proc. 12th Int. Teletraffic Congress*, Turin, Italy, June 1988, pp. 1364-1374.

[10] J. Zhang and E. Coyle, "Matrix recursive solutions for the transient behavior of QBD-processes and its application to random access networks," in *Proc. Inform. Sci. Syst. Conf.*, Princeton, NJ, 1988.

[11] J. McQuillan and D. Walden, "The Arpanet design decision," *Computer Networks*, vol. 1, pp. 243-289, 1977.

[12] S. Tripathi and A. Duda, "Time-dependent analysis of queueing systems," *INFOR*, vol. 24, no. 3, pp. 199-219, 1986.

[13] T. C. Kotiah, "Approximate transient analysis of some queueing systems," *Oper. Res.*, vol. 26, pp. 334-346, 1978.

[14] P. E. Cantrell, "Computation of the transient $M/M/1$ queue Cdf, Pdf, and mean with generalized Q functions," *IEEE Trans. Commun.*, vol. COM-34, pp. 814-817, Aug. 1986.

[15] M. H. Ackroyd, "$M/M/1$ transient state occupancy probabilities via the discrete Fourier transform," *IEEE Trans. Commun.*, vol. COM-30, pp. 557-559, Mar. 1982.

[16] S. K. Jones, R. K. Cavin, and D. A. Johnson, "An efficient computational procedure for the evaluation of the $M/M/1$ transient state occupancy probabilities," *IEEE Trans. Commun.*, vol. COM-28, pp. 2019-2020, Dec. 1980.

[17] A. R. Odoni and E. Roth, "An empirical investigation of the transient behavior of stationary queueing systems," *Oper. Res.*, vol. 31, no. 3, pp. 432-455, May-June, 1983.

[18] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes.* Cambridge, England: Cambridge University Press, 1986.

[19] S. Ross, *Stochastic Processes.* New York: Wiley, 1983.

[20] M. R. Taafe, "Approximating nonstationary queueing systems," Ph.D. dissertation, Ohio State Univ., Columbus, OH, 1982.

[21] D. Tipper, "Adaptive routing, flow control and buffer management in computer communication networks," Ph.D. dissertation, Univ. of Arizona, 1988.

[22] C. Agnew, "Dynamic modeling and control of congestion-prone systems," *Oper. Res.*, vol. 24, no. 3, pp. 400-419, 1976.

[23] K. L. Rider, "A simple approximation to the average queue size in the time-dependent $M/M/1$ queue," *J. Ass. Comput. Mach.*, vol. 23, no. 2, pp. 361-367, 1976.

[24] G. Newell, *Applications of Queueing Theory.* London: Chapman and Hall, 1971.

[25] D. Tipper and M. K. Sundareshan, "An optimal control approach to decentralized dynamic virtual circuit routing in computer networks," in *Proc. IEEE INFOCOM '90*, San Francisco, CA, June 5-7, 1990.

[26] W. Lovegrove, "A methodology for simulation of the nonstationary behavior of computer networks," Ph.D. dissertation, Clemson Univ., Clemson, SC, 1990.

[27] A. P. Sage and C. White, *Optimum Systems Control.* Englewood Cliffs, NJ: Prentice-Hall, 1977.

[28] L. Kleinrock, "Power and deterministic rules of thumb for probabilistic problems in computer communications," in *Proc. Int. Conf. Commun.*, vol. 1, June 1979, pp. 43.1.1-43.1.10.

[29] S. Stidham, "Optimal control of admission to a queueing system," *IEEE Trans. Automat. Contr.*, vol. AC-30, no. 8, pp. 705-713, 1985.

[30] P. Chemouil and J. Filipiak, "Kalman filtering of traffic fluctuations for real-time network management," in *Proc. 4th IFAC/IFORS Symp. Large Scale Syst.*, Zurich, Switzerland, 1986.

[31] S. Hantler and Z. Rosberg, "Optimal estimation for an $M/M/c$ queue with time varying parameters," *Stochastic Models*, vol. 5, no. 2, pp. 295-313, 1989.

David Tipper (S'78-M'88) obtained the B.S. degree in electrical engineering from Virginia Tech, Blacksburg, in 1980 and the M.S. degree in systems engineering in 1984 and the Ph.D. degree in electrical engineering in 1988, both from the University of Arizona, Tucson.

From 1980 to 1982 he was employed as a Systems Engineer on NASA's space shuttle mission simulator by Singer-Link in Houston, TX. Currently he is an Assistant Professor in the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC. His current research interests are the performance analysis of communication networks and the control ol queueing systems.

Malur K. Sundareshan (M'77) received the B.E. degree in electrical engineering from Bangalore University, Bangalore, India, in 1966, and the M.E. and Ph.D. degrees in electrical engineering from the Indian Institute of Science, Bangalore, India, in 1969 and 1973, respectively.

Between 1973 and 1976, he held various visiting faculty positions at the Indian Institute of Science, Bangalore, at the University of Santa Clara, Santa Clara, CA, and at Concordia University, Montreal, Quebec, Canada. From 1976 to 1981, he was on the faculty of the Department of Electrical Engineering, University of Minnesota, Minneapolis. Since 1981, he has been on the faculty of the Department of Electrical and Computer Engineering, University of Arizona, Tucson, where he is a Professor. His current research interests are in large-scale systems, communication networks, adaptive control and estimation, and statistical signal processing.