

# Transient Analysis of Manufacturing Systems Performance

Y. Narahari and N. Viswanadham, *Fellow, IEEE*

**Abstract**— Studies in performance evaluation of automated manufacturing systems, using simulation or analytical models, have always emphasized *steady-state* or *equilibrium* performance in preference to *transient* performance. In this study, we present several situations in manufacturing systems where transient analysis is very important. Manufacturing systems and models in which such situations arise include: systems with failure states and deadlocks, unstable queueing systems, and systems with fluctuating or non-stationary workloads. Even in systems where equilibrium exists, transient analysis is important in studying issues such as accumulated performance rewards over finite intervals, first passage times, sensitivity analysis, settling time computation, and deriving the behavior of queueing models as they approach equilibrium. In certain systems, convergence to steady-state is so slow that only transient analysis can throw light on the system performance. In this paper, we focus on transient analysis of *Markovian* models of manufacturing systems. After presenting several illustrative manufacturing situations where transient analysis has significance, we discuss two problems for demonstrating the importance of transient analysis. The first problem is concerned with the computation of distribution of time to absorption in Markov models of manufacturing systems with deadlocks or failures, and the second problem shows the relevance of transient analysis to a multiclass manufacturing system with significant setup times. We also briefly discuss computational aspects of transient analysis.

## I. INTRODUCTION

**S**TUDIES in performance analysis of discrete manufacturing systems and in general, discrete event dynamical systems have traditionally emphasized *steady-state* or *equilibrium* performance over *transient* or *time-dependent* performance. This paper is concerned with transient analysis of manufacturing systems performance. Transient analysis is very important in manufacturing system models that do not attain a steady state or equilibrium. Examples of such systems include, systems with failure states, unstable queueing systems, and systems with fluctuating or non-stationary workloads. Even in systems where equilibrium does exist, transient analysis is important for studying performance over finite intervals, sensitivity analysis, first passage time computation, settling time computation, and for deriving the behavior of models as they approach equilibrium.

In this paper, we view a manufacturing system as a *discrete event dynamical system* [1,2] and consider that the evolution of a manufacturing system constitutes a discrete state space

Manuscript received November 20, 1992; revised June 29, 1993. This work was sponsored by a Department of Science and Technology, Government of India, research grant in the area of manufacturing systems.

The authors are with the Indian Institute of Science, Department of Computer Science and Automation, Bangalore, India.

IEEE Log Number 9214692.

stochastic process. In particular, we focus on Markov chain models. Such a model could be generated directly or using higher level models such as queueing networks, stochastic Petri nets, or discrete event simulation [2].

### A. Steady-State Analysis

Steady-state analysis has been the focus of most performance studies in the area of discrete manufacturing systems. The two recent textbooks in this area, by Viswanadham and Narahari [2], and by Buzacott and Shantikumar [3] are concerned mostly with steady-state analysis. There are also many survey articles that discuss steady-state analysis of manufacturing systems using simulation modeling [4], Markov chain models [5], queues and queueing network models [6], [7], [8], and stochastic Petri net models [9], [10].

Steady-state analysis deals mainly with customer average measures or time average measures. Performance measures such as steady-state waiting time belong to the first category whereas measures such as steady-state number of jobs in system are time average measures. In the literature, much of the analysis deals with only mean values of these performance measures. Higher moments and distributions are only occasionally computed, for special classes of systems.

There are three main reasons for the popularity of steady-state analysis:

- 1) There are computationally efficient and simple methods for steady-state analysis. For example, the computation of steady-state probabilities in a Markov chain is carried out by solving a system of linear equations; the computation of performance measures in product form queueing networks is accomplished through efficient polynomial-time algorithms; and so on. Availability of a wide variety of efficient linear equation solvers, including parallelized algorithms, has made possible the solution of Markov chains with several hundred thousand states.
- 2) Major results in queueing theory, such as Burke's result [11], Little's law [12], Jackson's theorem [13], product form of closed queueing networks [14], the BCMP formulation [15], and the arrival theorem [16] are all concerned with steady-state analysis.
- 3) Developments in aggregation and decomposition methods for solving large Markov chain models or large queueing models have also focused on steady-state analysis (see, for example, the paper by Curtois [17]).

Often, manufacturing system models do not have a steady state or do not reach a steady state in the observation period

of interest. Transient analysis becomes important in such situations. In Section II, we will be looking at several such situations.

**B. Transient Analysis**

Let us assume that a manufacturing system evolves in time as a homogeneous continuous time Markov chain (CTMC)  $\{X(t) : t \geq 0\}$  with state space  $S = \{0, 1, \dots\}$  and infinitesimal generator  $Q$ . Let  $i, j \in S$  and

$$p_{ij}(t) = P\{X(t) = j | X(0) = i\}$$

$$H(t) = [p_{ij}(t)]$$

The forward and backward differential equations that govern the behavior of this CTMC are respectively given by [18], [19], [2],

$$\frac{d}{dt}(H(t)) = H(t)Q \tag{1}$$

$$\frac{d}{dt}(H(t)) = QH(t) \tag{2}$$

with initial conditions  $H(0) = I$  in both the cases. Note that these are first order, linear, ordinary differential equations. In terms of the individual matrix elements, the above equations become

$$\frac{d}{dt}(p_{ij}(t)) = q_{jj}p_{ij}(t) + \sum_{k \neq j} q_{kj}p_{ik}(t) \tag{3}$$

$$\frac{d}{dt}(p_{ij}(t)) = q_{ii}p_{ij}(t) + \sum_{k \neq i} q_{ik}p_{kj}(t) \tag{4}$$

The forward and backward equations have the same unique solution given by

$$H(t) = e^{Qt} \tag{5}$$

where  $e^{Qt}$  is the matrix exponential defined by the Taylor series

$$e^{Qt} = \sum_{k=0}^{\infty} \frac{(Qt)^k}{k!} \tag{6}$$

If we are interested in the state probabilities

$$\Pi(t) = [p_0(t), p_1(t), \dots]$$

where  $p_j(t) = P\{X(t) = j\}$ ,  $j \in S$ , then we need to solve the differential equation

$$\frac{d}{dt}(\Pi(t)) = \Pi(t)Q \tag{7}$$

The solution of the above is given by

$$\Pi(t) = \Pi(0)e^{Qt} \tag{8}$$

*1) An Example:* To get a feel for the equations above, let us consider a simple example [19], [2]. Consider a manufacturing system comprising a single machine that fails with failure time exponentially distributed with rate  $\lambda$  and gets repaired, once failed, with repair time exponentially distributed with rate  $\mu$ . Assuming that the failure and repair times are independent, the system can be formulated as a CTMC with state space  $S = \{0, 1\}$  where state 0 indicates, say, "machine in the up condition" and state 1 denotes "machine undergoing repair." Figure 1 depicts the state diagram of this Markov chain. For

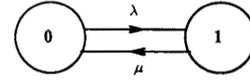


Fig. 1. Markov chain model of a single machine system

this example, we have

$$H(t) = \begin{bmatrix} p_{00}(t) & p_{01}(t) \\ p_{10}(t) & p_{11}(t) \end{bmatrix}$$

$$Q = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}$$

The forward equations (1) in this case are given by

$$\frac{d}{dt}(p_{00}(t)) = p_{00}(t)q_{00} + p_{01}(t)q_{10}$$

$$\frac{d}{dt}(p_{01}(t)) = p_{01}(t)q_{11} + p_{00}(t)q_{01}$$

$$\frac{d}{dt}(p_{10}(t)) = p_{10}(t)q_{00} + p_{11}(t)q_{10}$$

$$\frac{d}{dt}(p_{11}(t)) = p_{11}(t)q_{11} + p_{10}(t)q_{01}$$

The backward equations are given by

$$\frac{d}{dt}(p_{00}(t)) = q_{00}p_{00}(t) + q_{01}p_{10}(t)$$

$$\frac{d}{dt}(p_{01}(t)) = q_{01}p_{11}(t) + q_{00}p_{01}(t)$$

$$\frac{d}{dt}(p_{10}(t)) = q_{10}p_{00}(t) + q_{11}p_{10}(t)$$

$$\frac{d}{dt}(p_{11}(t)) = q_{11}p_{11}(t) + q_{10}p_{01}(t)$$

The solution of the coupled differential equations above is straightforward and it can be shown that the transition probabilities are given by

$$p_{00}(t) = \frac{\mu}{\lambda + \mu} + \left(\frac{\lambda}{\lambda + \mu}\right)e^{-(\lambda + \mu)t} \tag{9}$$

$$p_{01}(t) = \frac{\lambda}{\lambda + \mu} - \left(\frac{\lambda}{\lambda + \mu}\right)e^{-(\lambda + \mu)t} \tag{10}$$

$$p_{10}(t) = \frac{\mu}{\lambda + \mu} - \left(\frac{\mu}{\lambda + \mu}\right)e^{-(\lambda + \mu)t} \tag{11}$$

$$p_{11}(t) = \frac{\lambda}{\lambda + \mu} + \left(\frac{\mu}{\lambda + \mu}\right)e^{-(\lambda + \mu)t} \tag{12}$$

Figure 2 illustrates the evolution of these state probabilities. Note that

$$\lim_{t \rightarrow \infty} p_{00}(t) = \lim_{t \rightarrow \infty} p_{10}(t) = \frac{\mu}{\lambda + \mu}$$

$$\lim_{t \rightarrow \infty} p_{11}(t) = \lim_{t \rightarrow \infty} p_{01}(t) = \frac{\lambda}{\lambda + \mu}$$

The above limiting probabilities are precisely the steady-state probabilities  $\pi_0$  and  $\pi_1$  of the states 0 and 1, respectively. For  $j = 0, 1$ , the state probabilities  $p_j(t)$  are given by

$$p_j(t) = p_{0j}(t)p_0(0) + p_{1j}(t)p_1(0) \tag{13}$$

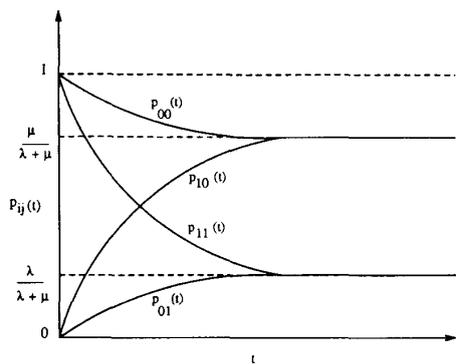


Fig. 2. Evolution of transition probabilities

2) *Relevant Literature*: Literature on transient analysis of Markov chain models is vast and is scattered across several inter-disciplinary areas. We shall only mention here some papers that are of direct interest.

Grassman's article [20] is an authentic survey on transient analysis whereas the paper by Stewart [21] discusses numerical techniques for transient analysis. The edited volume by Stewart [22] has several papers that touch upon transient analysis of Markovian models. The recent survey paper by Philippe, Saad, and Stewart [23] on numerical methods in Markov chain modeling also has some relevant discussion on transient analysis. In a highly relevant survey, Reibman and Trivedi [24], [25] have provided an overview of the various numerical techniques for transient analysis while Marie *et al.* [26] have discussed the transient analysis of acyclic Markov chains. Bobbio and Trivedi [27], [28] have discussed an aggregation method for transient analysis of Markov chains.

Reliability and availability modeling has been a major motivating factor for conducting transient analysis. For example, see the papers by Reibman *et al.* [29], Bavuso *et al.* [30], Dyer [31], and de Souza de Silva and Gail [32,33]. The paper by Dyer [31] directly deals with transient analysis of Markovian models that arise in reliability, availability, and repairability modeling, and develops an efficient approximate method for transient analysis, that exploits the special structure of the transition rate matrix in such models. Analysis of fault-tolerant computer systems and performability modeling have also spurred several research efforts in transient analysis. For example, see the works by de Souza de Silva and Gail [33], Gerber [34], Meyer [35], and Trivedi *et al.* [36].

Transient analysis of queueing models arising in computer and communication systems is the subject of the works by Baiocchi *et al.* [37], Kotiah [38], Konstantopoulos and Baccelli [39], Tripathi and Duda [40], Upton and Tripathi [41], Weiss and Mitra [42], and Kobayashi [43].

In the manufacturing context, some work on transient analysis has been reported in the works of Ram [44], Gopalakrishna [45], Viswanadham and Ram [46], Ram and Viswanadham [47], Viswanadham and Narahari [2], and Viswanadham *et al.* [48,49]. In these works, transient analysis is applied to manufacturing situations from a *performability* viewpoint. The main objective of these works is to compute the complete

distributions of throughput and cycle time in the presence of unreliable machines and components which may fail randomly. The papers by Viswanadham *et al.* [48], Viswanadham and Ram [46], and Ram and Viswanadham [47] comprise results to compute the performability distributions for single part type manufacturing systems. The recent works of Viswanadham, Pattipati, and Gopalkrishna [49] and Gopalakrishna [45] treat the multiple part type case and show that the performability distributions can be obtained by solving a set of forward or adjoint linear hyperbolic partial differential equations. These papers also develop efficient computational methods for solving such equations. Other notable contributions towards transient analysis of manufacturing systems are the papers by Miltenburg [50], Gershwin [51], and Malhame and Boukas [52]. The papers by Miltenburg and Gershwin contain transient analysis results for tandem or serial production lines and both address the problem of computing the variance of throughput in those systems. Malhame and Boukas look at the statistical evolution of a manufacturing system producing a single product, under hedging point control policies. They formulate partial differential equations that describe this evolution and show that transient analysis is very important here since the convergence to steady state is very slow.

The aim of this paper is to spell out clearly the need for transient analysis of manufacturing system models and to explore the major issues of relevance.

### C. Organization of the Paper

In this section, we have introduced the transient analysis problem in performance modeling. In the next section, we discuss several situations in manufacturing systems analysis where transient analysis is relevant. We discuss these under four categories:

- 1) Systems where steady state does not exist.
- 2) Models with absorbing states.
- 3) Performance computation over finite time durations.
- 4) Other important transient phenomena.

In Section III, we present two illustrative examples. The first is concerned with the computation of *time to absorption* in Markov models with absorbing states. This analysis can be used to study manufacturing systems with deadlocks and systems with total failure states. The second example is that of a machine center that produces two types of products with substantial setup times to switch over from one product type to another. For this system, we show that transient analysis can yield performance values that are often significantly different from those obtained using steady-state analysis.

In Section IV, we briefly touch upon important computational issues in transient analysis. In Section V, we provide a summary of the paper.

## II. WHY TRANSIENT ANALYSIS?

The aim of this section is to provide various situations in manufacturing system analysis where transient analysis assumes much significance.

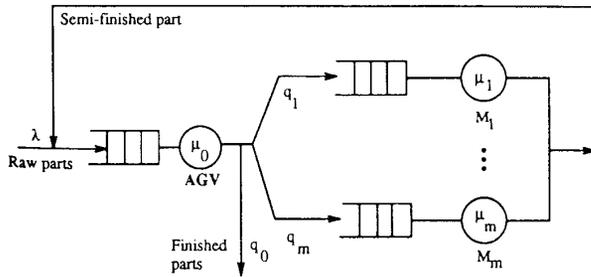


Fig. 3. Open central server queueing network model

A. Systems with No Steady State

It is only in special classes of Markov chain models, such as *ergodic* Markov chains, that a unique steady state or equilibrium exists. We now give some examples where a steady state does not exist.

Example 1: An Unstable Queue.

Consider an M/M/1 queue with arrival rate  $\lambda$  and service rate  $\mu$ . The queue is stable if and only if  $\lambda < \mu$  and steady-state performance measures will be meaningful only in this case. When  $\lambda = \mu$ , it is known that the underlying Markov chain states are all *null recurrent* [53] and the number of customers in the system grows to infinity in the long term. If  $\lambda > \mu$ , all the states are *transient* and the system is again unstable. Similar arguments hold for any single or multiple server queueing system. The operation of machine centers that are flooded with a large number of demands or crippled machines with reduced service capacities can be faithfully represented by such unstable queues.

Example 2: An Unstable Queueing Network.

Consider the *open central server queueing network* model shown in Fig. 3. This is a very popular model of flexible manufacturing systems [54], [7], [3], [2]. This network is a special class of a *Jackson network* [13]. If  $\lambda$  is the external arrival rate of jobs and  $\mu_i (i = 0, 1, \dots, m)$  are the service rates (see Fig. 3), it is known that the above network is stable if and only if  $\rho_j < 1$  for all  $j = 0, 1, \dots, m$ , where

$$\rho_0 = \frac{\lambda}{q_0 \mu_0}$$

$$\rho_j = \frac{\lambda q_j}{q_0 \mu_j} \quad j = 1, \dots, m$$

If even one of these conditions is not satisfied, the network is unstable and steady-state analysis loses significance. Such an unstable queueing network could be the model of a heavily loaded job shop or a manufacturing system whose service capacity is reduced by machine or subsystem failures.

Example 3: A Kanban Cell with Non-Stationary Demands.

Mitra and Mitrani [55] have studied the performance of a linear network of Kanban cells, subjected to stochastic demands. Figure 4 depicts a single Kanban cell subjected to external demands. The input to the machines is modulated by the arrival processes of demands and raw parts.

Mitra and Mitrani [55] assume that the demands for finished parts arrive according to a Poisson process. However, in the

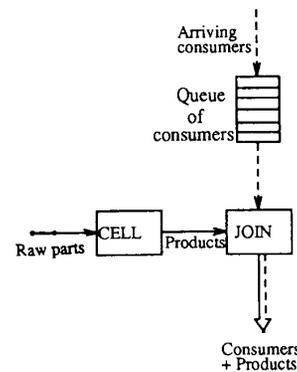


Fig. 4. A Kanban cell subjected to external demands

real-world context, the demands arrive in very complex fashion and the workload to the system is highly *non-stationary*. For example, during rush hours, the demands arrive rapidly and during other times, their arrival follows some stochastic pattern. The underlying queueing system belongs to the realm of non-stationary queues and the system here may be unstable or stable depending on the maximum rate of arrivals of demands and raw parts. There is a rich body of literature in the area of non-stationary queueing systems [56], where the issue of stability has been resolved for a very limited class of models.

Example 4: Re-Entrant Lines.

Re-entrant lines [57] constitute a class of manufacturing systems models where the flows are *non-acyclic* since the parts visit the same machines several times. These are characteristic of semiconductor and thin film manufacturing. Scheduling is an important problem in these systems and several distributed policies based on buffer priorities and due dates have been formulated for these systems (see, for example, the papers by Kumar [57] and by Lu and Kumar [58]). Stability is an important issue in evaluating these scheduling policies. Not all the policies suggested in the above papers are stable [57], [58] and performance analysis of re-entrant lines under such unstable policies can only be carried out via transient analysis.

B. Models with Absorbing States

Markov models with absorbing states have a trivial steady-state, namely that the chain ends up in some absorbing state, remaining there forever; therefore, transient analysis alone throws any light on the system performance. We consider two examples below.

Example 5: Reliability Analysis.

Manufacturing systems with no or limited repair of failed elements will lead to models with absorbing states. In such systems, reliability is an important performance index. Consider, for instance, a manufacturing system with  $m$  identical machines and an automated guided vehicle (AGV). Both the machines and the AGV are failure-prone and let us assume that repair is not possible. If the failure times are all independent exponential random variables, then the model that describes the failure-repair behavior of this system is a Markov chain. It is reasonable to assume that the system is *operational* only

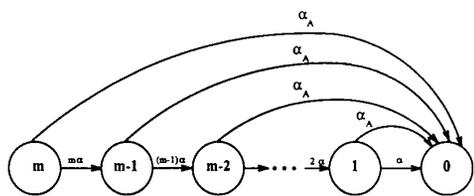


Fig. 5. Markov chain model for failure-repair behavior.

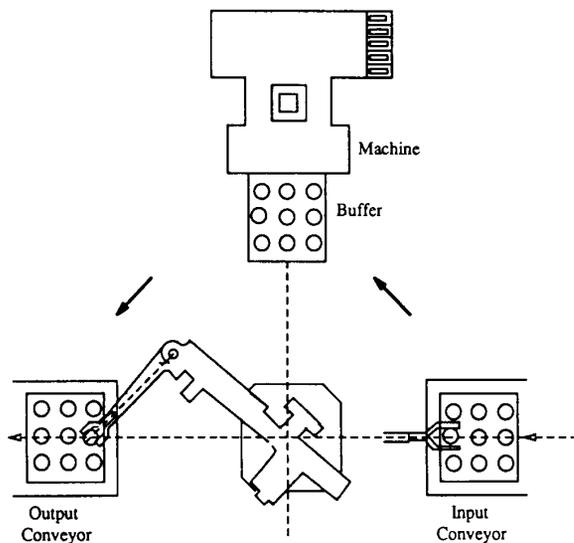


Fig. 6. A robotic cell to illustrate deadlock.

when the AGV is "up" and at least one machine is "up" (this is because the AGV is involved in the successful completion of processing of every job). In such a case, the Markov chain model has state space  $S = \{0, 1, \dots, m\}$ , where state 0 corresponds to the *failed state* (all machines are down or AGV is down or both) and state  $i$  ( $i = 1, \dots, m$ ) indicates AGV "up" and exactly  $i$  machines "up." Figure 5 shows this Markov chain model, assuming  $\lambda_A$  as the AGV failure rate and  $\lambda$  as the failure rate of each machine. This same model is discussed in depth in [48]. State 0 is an absorbing state and the *reliability* of this system at time  $t$  is the probability that the system is not in state 0 at time  $t$ , given some initial condition. The reliability in this case can only be computed through transient analysis.

#### Example 6: A Manufacturing System with Deadlocks.

This example is taken from [2]. Consider the robotic cell shown in Fig. 6, where there is a single machine that produces parts, with processing time exponentially distributed with rate  $\mu$ . Raw parts arrive onto an input conveyor according to a Poisson process having rate  $\lambda$ . A robot picks up a raw part from the input conveyor and loads it onto the machine if the machine is free or to its buffer if the machine is busy. The robot picks up the finished part and puts it on the output conveyor. Assume that arrival of raw parts into the system is inhibited whenever the machine is busy, the buffer is full, and the robot is holding a raw part. Hence, if the buffer capacity is  $n$ , the

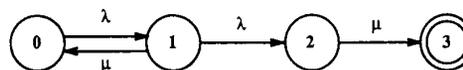


Fig. 7. Markov chain model of the robotic cell.

maximum number of jobs inside the system is  $n + 2$ . Let us assume that the robot takes negligible time to load and unload parts.

First, consider the case where there is no buffer. Here, the states of the system are 0, 1, 2, 3, with the following interpretation:

- 0: no raw parts; machine idle.
- 1: machine processing a part, no raw parts waiting.
- 2: machine processing a part, robot holding a raw part.
- 3: machine waiting for the robot to transfer the finished part and the robot waiting for the machine to release the finished part.

The CTMC model of the above system is shown in Fig. 7. In state 3, the waiting is indefinite if we assume that the robot controller and the machine controller are not programmed to react to such mutual or circular waiting. Such a state is called a *deadlock*, which stalls further activity and production in the system. In this simple example, it is easy to see how the deadlock may be prevented, but in a real-world manufacturing system having a large number of resources and concurrent interactions, deadlocks can occur commonly. Deadlock prevention or deadlock avoidance policies can be used to eliminate such deadlock situations, but such policies often lead to poor resource utilization [2]. For this reason, resource allocation policies that might result in deadlocks are preferred to avoidance or prevention strategies, in order to maintain an acceptable level of resource utilization.

State 3 is an absorbing state in Fig. 7. If we need to compute the distribution of time before the deadlock is reached or the number of parts produced before deadlock, transient analysis becomes important.

In the above example, if there is a buffer in front of the machine, the number of states will increase; in fact, if the buffer capacity is  $n$ , there will be exactly  $n + 4$  states in the model and state  $n + 3$  will be the absorbing state.

#### C. Performance in Finite Intervals

In a manufacturing system, we would often be interested in computing the cumulative performance in a finite duration of time, for example in a shift period. It is not realistic to expect the system to reach a steady state during this finite observation period. We consider three examples below.

##### Example 7: A Wafer Fabrication Line.

In a typical semiconductor wafer fabrication line [59], [60], each *lot* of wafers goes through a large number of operations and spends several days, inside clean rooms, repeatedly visiting many workcenters. The typical cycle time and queueing time of a *lot* of wafers is much larger compared to a shift duration. Therefore, if we are interested in the production or congestion levels at the end of a shift duration, we cannot rely on steady-state performance estimates. Furthermore, some scheduling policies in such *re-entrant lines* are known to be

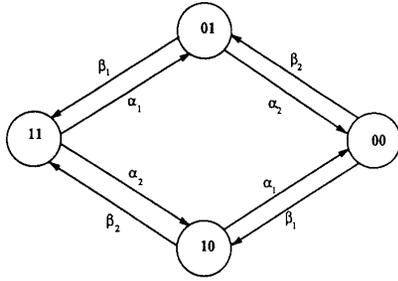


Fig. 8 Failure-repair model of a two-machine system

unstable (see Example 4) and transient analysis becomes even more important.

*Example 8: Interval Dependability Measures.*

Fault-tolerance and flexibility are the prime attributes of advanced manufacturing systems. The degree of fault-tolerance of a manufacturing system is characterized by *dependability* measures such as reliability and availability. To define these measures, we partition the system states into *operational* states (states in which the system produces useful output) and *failed* states. Given an interval  $[0, t]$ , the *reliability* of the system is the probability that the system never reaches a failed state during that interval. The *point availability* at time  $u \in [0, t]$  is the probability that, at time  $u$ , the system is in an operational state. The *interval availability* is the fraction of time during  $[0, t]$ , the system is in operational states. To compute these measures, one needs to do transient analysis.

As an illustrative example, we consider a manufacturing system comprising two machines  $M_1$  and  $M_2$  (this example is taken from [48]). Let the failure times of  $M_i$  ( $i = 1, 2$ ) be exponentially distributed with rate  $\alpha_i$  and be independent. When a machine fails, assume that repair starts immediately, with repair time for machine  $M_i$  being an exponential random variable having rate  $\beta_i$ . The failure-repair behavior of this system is a Markov chain with four states given by

$$S = \{(11), (10), (01), (00)\}$$

where each state is a pair  $(x_1, x_2)$ , with  $x_i = 1$  when  $M_i$  is “up” and  $x_i = 0$  when  $M_i$  is “down.” Figure 8 shows this Markov chain.

Obviously, the set of operational states is given by

$$S_o = \{(11), (10), (01)\}$$

and the set of failed states is given by

$$S_f = \{(00)\}$$

Let  $\{Z(u) : u \geq 0\}$  be this Markov chain. Given an interval  $[0, t]$ , the reliability  $R(t)$  is given by

$$R(t) = P\{Z(u) \in S_o \forall u \in [0, t]\}$$

The point availability is given by

$$PA(u) = P\{Z(u) \in S_o\}$$

The interval availability is given by

$$IA(t) = \frac{1}{t} \int_0^t P\{Z(u) \in S_o\}$$

The above failure-repair process is often referred to as the *structure state process* [48].

*Example 9: Performability Measures.*

Performability is a generic, composite measure of performance and dependability. There is a vast literature on performability of computer and communication systems [33]. More recently, performability has been investigated in the manufacturing systems context also [48].

We shall give a simple example, based on the system in Example 8. Assume that raw parts are always available and that parts undergo exactly one operation, either on  $M_1$  or on  $M_2$ , and leave the system. Also, assume that machine  $M_i$  processes parts at rate  $\mu_i$ . Then in state (11), the total production rate is  $\mu_1 + \mu_2$ . The production rates in states (10), (01), (00) are respectively,  $\mu_1$ ,  $\mu_2$ , and zero. During the interval  $[0, t]$ , let  $\tau_{11}, \tau_{10}, \tau_{01}, \tau_{00}$  be the total times spent in the corresponding states. Note that these are random variables. The total accumulated production in the interval is then given by

$$Y(t) = \tau_{11}(\mu_1 + \mu_2) + \tau_{10}\mu_1 + \tau_{01}\mu_2$$

$Y(t)$  is called the throughput-related performability. In general, performability could be with respect to any performance measure such as throughput, lead time, queueing time, etc. To compute the distribution of  $Y(t)$ , one needs to do transient analysis.

*D. Other Transient Phenomena*

There are many other aspects of manufacturing system performance that can be effectively addressed only by transient analysis.

1) *Performance under Real-Time Control Policies:* When real-time control decisions are taken, for example, in the dynamic scheduling of manufacturing system operations, it is of intrinsic interest to look at the transient performance, especially if the evolution is such that it takes a long time before a steady state is reached. For instance, Malhame and Boukas [52] have considered the operation of a failure-prone, single-product manufacturing system under dynamic hedging point control policies. They characterize the transient performance using a system of coupled partial differential equations.

2) *Settling Time of Queueing Systems:* The settling time of a queueing system with a given initial number of customers in the system is the total time until the number in the system is zero. There have been a few efforts at computing the distribution of settling time of multiserver queues and open queueing networks [61,62,63].

The notion of settling time is analogous to the *makespan* of a manufacturing network, which is the total amount of time required to complete the processing of a given number of workpieces. Makespan computation is quite important in stochastic manufacturing systems.

3) *Sensitivity Analysis:* It is often required to determine the performance or reliability *bottleneck* of a system. In this context, it is necessary to evaluate the derivative of the desired performance measure with respect to important system parameters. The parameter with the largest derivative deserves

the attention of the designers to improve the characteristics of the designed system. Such derivatives can also be used in a system optimization effort based on gradient search techniques. Sensitivity analysis often relies on transient analysis of performance. For example, Heidelberger and Goyal [64] have shown how transient analysis techniques can be effectively utilized for sensitivity analysis in continuous time Markov chains. The SPNP (Stochastic Petri Net Package) tool [65], in fact, includes procedures for analyzing sensitivity of various performance measures to changes in system parameters.

4) *Cut-Off Phenomenon*: An interesting quantity to study in the evolution of a stochastic manufacturing system is the rate at which the steady state is approached. This depends on the time constants (eigen values) of the system [42]. There is a class of Markov chain models and queueing systems (for example, see the articles by Konstantopoulos and Baccelli [39] and Anantharam [66]) which exhibit a *cut-off phenomenon* namely, the existence of a time such that before this time, the system is far from steady state, while, after this time, the system is very close to steady state. The existence of cut-off phenomenon is a good indicator to whether a transient or a steady-state analysis is appropriate in a given setting. For example, if the cut-off time is known and the duration of observation is less than the cut-off time, then transient analysis is more meaningful than steady-state analysis.

### III. DETAILED EXAMPLES

In this section, we illustrate transient analysis of manufacturing systems using two examples. In the first, we show the computation of distributions of *time to absorption* in a Markov model with absorbing states. In the second, we show how performance estimates, obtained using transient analysis, may be significantly different from those of steady-state analysis.

#### A. Time to Absorption

We have observed in Section II-B that absorbing states occur in manufacturing system models that capture non-repairable behavior and phenomena such as deadlocks. An important quantity of interest in such systems is the time until an absorbing state is reached. Let  $\{X(u) : u \geq 0\}$  be the Markov chain under consideration. Let the state space be finite and given by

$$S = \{0, 1, \dots, m, m+1, \dots, m+n\}$$

where  $m \geq 0$ ,  $n > 0$ , the first  $(m+1)$  states are transient states, and the rest of the states are absorbing states. Let 0 be the initial state and  $T$ , the time to reach any absorbing state. Define

$$p_{ij}(t) = P\{X(t) = j | X(0) = i\}$$

Then, we have, for any  $t > 0$ ,

$$P\{T > t\} = P\{X(t) \notin \{m+1, \dots, m+n\}\}$$

In other words, we have

$$P\{T > t\} = 1 - \sum_{j=1}^n p_{0,m+j}(t)$$

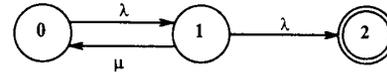


Fig. 9. A Markov chain with an absorbing state

Hence the cumulative distribution function of  $T$  is given by

$$F_T(t) = \sum_{j=1}^n p_{0,m+j}(t) \quad (14)$$

The individual probabilities  $p_{0,m+j}(t)$  have to be computed by solving the differential equations shown in (1) or (2).

We now show the computation of the distribution of time to absorption for a simple Markov chain. Consider the Markov chain of Fig. 9.

There are two possible interpretations for the above model. In the first interpretation, we have a single machine system which is in state 0 when there is no part being processed, in state 1 when there is a part being processed, and in state 2 when there is a deadlock. The arrival rate of parts is  $\lambda$  and the service rate of each part is  $\mu$ . This interpretation is similar to Example 6. The time to absorption here is the time elapsed before a deadlock is reached.

In the second interpretation, we consider a two-machine system with exponential failures and repairs. In state 0, both machines are "up" but only one of them is chosen to process parts. When this chosen machine fails, the system reaches state 1, in which the non-failed machine starts processing parts and the repair of the failed machine is in progress. If the non-failed machine now fails before completion of repair of the already failed machine, we reach state 2 and we abandon any further repair. On the other hand, if the failed machine in state 1 is repaired before the non-failed machine fails, we return to state 0. State 2 corresponds to a total failure state and the time to absorption corresponds to the time to total failure.

We know in this case that  $F_T(t) = p_{02}(t)$ . To compute  $p_{02}(t)$ , we first write down the infinitesimal generator  $Q$  of this Markov chain:

$$Q = \begin{bmatrix} -\lambda & \lambda & 0 \\ \mu & -(\lambda + \mu) & \lambda \\ 0 & 0 & 0 \end{bmatrix}$$

First consider the backward equation (4) for  $p_{02}(t)$ :

$$\frac{d}{dt}(p_{02}(t)) = q_{00}p_{02}(t) + q_{01}p_{12}(t) + q_{02}p_{22}(t)$$

Since  $q_{02} = 0$ , the above becomes

$$\frac{d}{dt}(p_{02}(t)) = -\lambda p_{02}(t) + \lambda p_{12}(t)$$

The backward equation for  $p_{12}(t)$  is given by

$$\frac{d}{dt}(p_{12}(t)) = q_{10}p_{02}(t) + q_{11}p_{12}(t) + q_{12}p_{22}(t)$$

Since  $p_{22}(t) = 1$ , the above becomes

$$\frac{d}{dt}(p_{12}(t)) = \mu p_{02}(t) - (\lambda + \mu)p_{12}(t) + \lambda$$

We shall solve for  $p_{02}(t)$  by the Laplace transform method. Let  $P_{ij}(s)$  denote the Laplace transform of  $p_{ij}(t)$ . Taking the transform on either side of the equations above, we get

$$sP_{02}(s) = -\lambda P_{02}(s) + \lambda P_{12}(s) \quad (15)$$

$$sP_{12}(s) = \mu P_{02}(s) - (\lambda + \mu)P_{12}(s) + \frac{\lambda}{s} \quad (16)$$

Simplifying using (15) and (16), we get

$$P_{02}(s) = \frac{\lambda^2}{s(s^2 + s(2\lambda + \mu) + \lambda^2)} \quad (17)$$

Now,  $p_{02}(t)$  can be obtained from equation (17) by inverse Laplace transformation. It is a simple matter to show that

$$p_{02}(t) = A + Be^{-at} + Ce^{-bt}$$

where the constants are given by

$$a = \frac{2\lambda + \mu + \sqrt{\mu^2 + 4\lambda\mu}}{2}; \quad b = \frac{2\lambda + \mu - \sqrt{\mu^2 + 4\lambda\mu}}{2}$$

$$A = \frac{\lambda}{ab}; \quad B = \frac{\lambda(b - 2a)}{ab(b - a)}; \quad C = \frac{\lambda}{b(b - a)}$$

In the above case, we were able to give a closed form expression for the cumulative distribution function of time to absorption, only because of the small number of states and simple structure. In general, this computation is a formidable task and in fact, is the subject of several research efforts. The problem is identical to computation of first passage times in Markov chains [67], [68], [69]. Marie, Reibman, and Trivedi have given a general way of obtaining such distributions efficiently for acyclic Markov chains [26]. There are several software tools that have been developed in this context and we will be briefly covering those in Section 4.

The early works of Kemeny [68] and Buzacott [70], [71] contain a discussion similar to the one presented in the above example. The recent book by Buzacott and Shantikumar [3] also has a brief discussion on computing the mean time to absorption.

### B. Transient Analysis of a Multiproduct Manufacturing Facility

Here, we consider a versatile machine center that is operated to produce two different classes of products, say A and B. The machine center switches production between the two product types based on the *exhaustive* service policy. That is, once set up for a particular product type, say A, processing is done on all class A parts until no more of them are waiting in queue. The machine will then switch over to produce class B products provided raw parts are available. Otherwise, it becomes idle. The switchover (setup) times are assumed to be quite substantial and this makes it interesting to study the transient characteristics of the system. We shall assume that there are two buffers, buffer1 and buffer2, of capacities  $N_1$  and  $N_2$ , respectively. See Fig. 10 for a schematic of the above

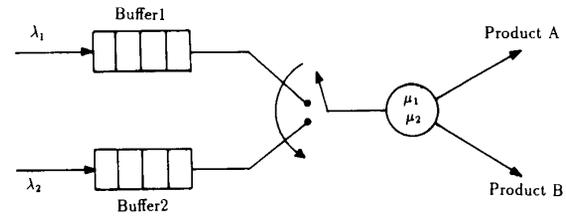


Fig. 10. A multiproduct manufacturing facility.

system. The items in buffer1 (buffer2) could correspond to any of the following:

- 1) Raw parts of class A (class B) waiting for their turn to get processed by the machine. In this case, the exogeneous arrivals into buffer1 (buffer2) correspond to externally arriving raw parts of class A (class B).
- 2) External demands for class A (class B) products. In this case, the exogeneous arrivals into buffer1 (buffer2) correspond to arriving external demands for class A (class B) products.

In the discussion that follows, we shall assume the first interpretation. The discussion is equally valid and relevant for the other interpretation. We make the following assumptions about the operation of the system.

- 1) Raw parts of class A (class B) arrive into the system according to a Poisson process with rate  $\lambda_1$  ( $\lambda_2$ ). Arriving raw parts of type A (type B) that find buffer1 (buffer2) full leave the system without undergoing service.
- 2) The setup time for product A (product B), which is also the time to switch over from product B to product A (product A to product B) is a stochastic variable, distributed exponentially with rate  $s_1$  ( $s_2$ ). We assume that  $s_1 = 0.5/h$  and  $s_2 = 0.4/h$ . That is, the average setup time for class A (class B) is 2 hours (2.5 hours).
- 3) The processing time for class A (class B) jobs is exponentially distributed with rate  $\mu_1$  ( $\mu_2$ ). In the numerical experimentation on the system, we have assumed  $\mu_1 = 4/h$  and  $\mu_2 = 6/h$ .
- 4) The *exhaustive* service policy [2] is used for switching over from one product type to another. That is, for example, if the machine is currently set up for product A, it will process class A parts as long as class A raw parts can be found in buffer1. When no more class A raw parts are available, the machine will switch over to product B if class B raw parts are available, otherwise the machine becomes idle with a setup for producing product A. When the machine is idle with a setup for product A and the next raw part to arrive is of class A, the machine will start processing that part without having to go through a setup; if the next raw part to arrive is of class B, the machine is set up for product B and then the processing starts.
- 5) FCFS (first come first serve) policy is used for dispatching parts in the individual buffers.
- 6) The machine does not fail during the interval of observation.
- 7) All the random variables involved are mutually independent.

8) The initial state of the system is: machine idle with setup for product A; buffer1 empty; and buffer2 empty. If the first arrival corresponds to class A, the machine will start directly processing the part. If the first arrival is of class B, the machine switches over to class B and starts processing.

Under the assumptions above, the model corresponds to a continuous time Markov chain. Using the SPNP (Stochastic Petri Net Package) [65] tool, the above Markov model was studied, to gain insight into the transient and steady-state performance of the system. The performance measures considered were:

- 1) Average cumulative throughput of class A (class B) parts during an interval  $[0, t]$ . Here,  $t$  can be any period of observation, for example, a shift of 8 hours duration or a full day's operation, etc.
- 2) Average manufacturing lead time (MLT) of class A (class B) jobs during an observation period  $[0, t]$ . The MLT of a job is the total time the job spends waiting and getting serviced in the system.
- 3) Mean steady-state throughput and mean steady-state MLT.

The performance results obtained for this system are shown in the graphs given in Figs. 11–16. The following convention is followed in these graphs: solid lines represent transient performance whereas dotted lines indicate steady-state performance; individual values for product A are shown by unfilled circles, while filled circles (i.e., black dots) indicate values for product B.

1) *Performance Over Finite Observation Periods:* For different observation intervals  $[0, t]$ , where  $t$  is varied from 1 hour to 12 hours in steps of 1 hour, the transient performance of the system is shown in Figs. 11 and 12. It is assumed that  $\lambda_1 = \lambda_2 = 4/h$ , and  $N_1 = N_2 = 4$ . Recall that  $s_1 = 0.5/h$ ,  $s_2 = 0.4/h$ ,  $\mu_1 = 4/h$ , and  $\mu_2 = 6/h$ . In Fig. 11, the average accumulated throughput during  $[0, t]$  (the average number of parts produced during  $[0, t]$ ) for each class is shown. As can be observed, both transient and steady-state values are shown. Note that the transient values are appreciably different from steady-state values. The effect of switchover times can be seen in the transient values. In the steady-state case, the effect of switchover times is averaged out and the throughput rate for the two classes is in the ratio of their processing rates.

Figure 12 shows the average MLT for the two product classes, for different observation intervals. Note that the transient values for class A reach a peak value around  $t = 4$  and the values slowly converge towards the steady-state values. The effect of the initial state is most appreciable in the case of class B, as can be seen from the steep decline in the beginning. If the interval of observation is  $[0, 4]$ , it can be observed that the average MLT of class A jobs reaches a maximum value whereas that for class B reaches a minimum value. Such interesting trends in system behavior can only be captured via transient analysis.

2) *Effect of Buffer Size:* Fixing the interval of observation as 8 hours and  $\lambda_1 = \lambda_2 = 4/h$ , we study the behavior of accumulated throughput in 8 hours and average MLT, as a

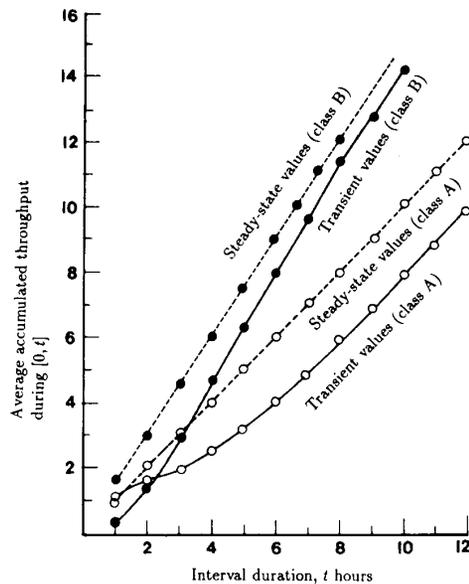


Fig. 11. Variation of average accumulated throughput.

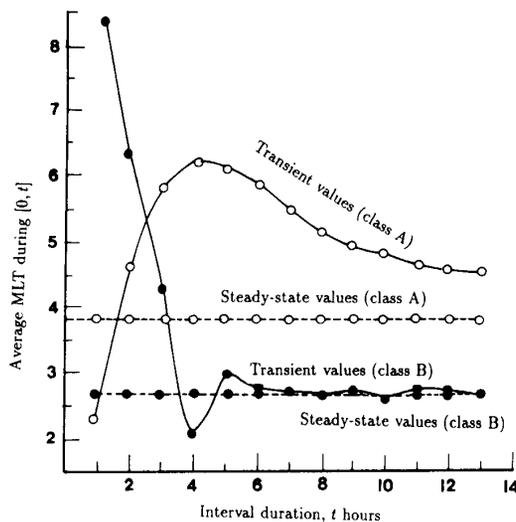


Fig. 12. Variation of average MLT with interval duration.

function of the size of the buffers. We assume that  $N_1 = N_2$  and vary this size from 1 to 12 in unit steps. With increase in buffer size, less number of arrivals leave the system without service leading to enhanced throughput and increased delay. This trend is exhibited in Figs. 13 and 14, except in some ranges of buffer sizes. We observe that:

- 1) Throughput of class B dominates over that of class A and the average MLT of class B is also relatively less. The throughput of class A jobs is found to decrease in certain ranges of buffer sizes since in those ranges, class B jobs are processed much more in a given setup due to their lower processing times. Consequently, the

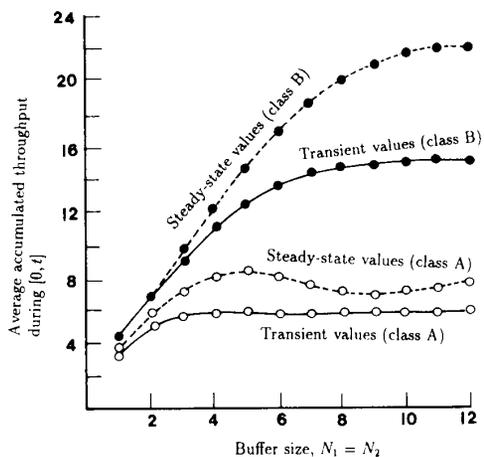


Fig. 13. Effect of buffer size on average accumulated throughput.

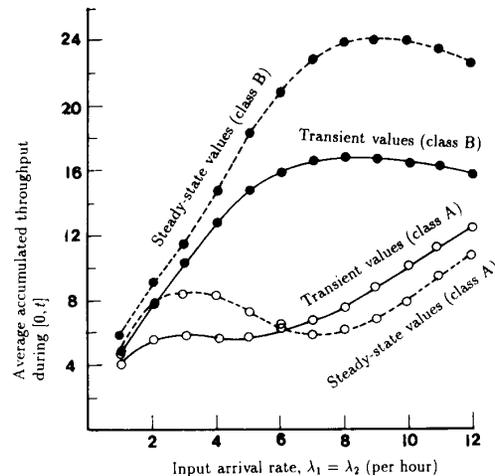


Fig. 15. Effect of arrival rate on average accumulated throughput.

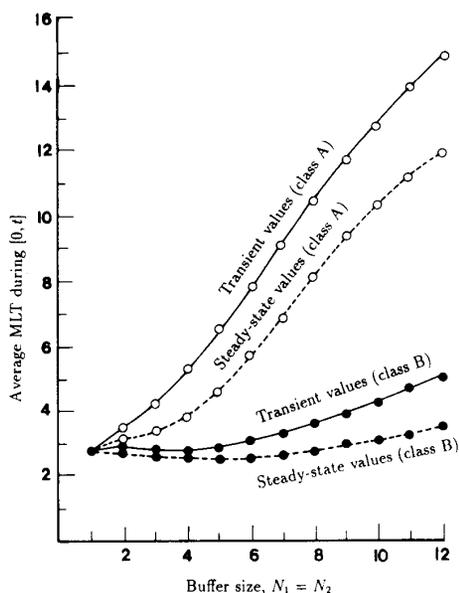


Fig. 14. Effect of buffer size on average MLT.

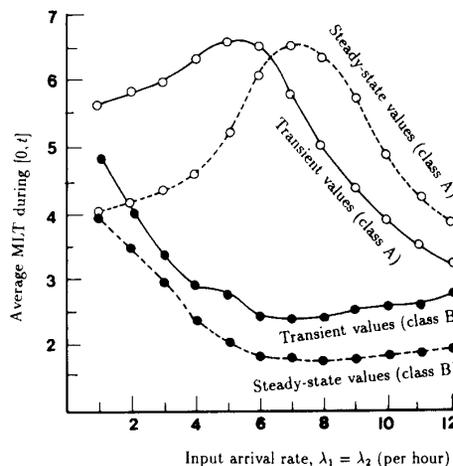


Fig. 16. Effect of arrival rate on average MLT.

average MLT of class B jobs in those ranges shows a slight decreasing trend.

- 2) Steady-state values of throughput are higher than the corresponding transient values whereas the reverse happens in the case of average MLT. The reason for this is that the effect of setup times is averaged out in the case of steady-state values.
- 3) The difference between transient values and steady-state values is quite appreciable and increases with buffer size. This happens because the size of the underlying Markov chain increases with buffer size and the time to attain steady state correspondingly increases.

3) *Effect of Arrival Rate:* Assuming an 8-hour observation period and fixing  $N_1 = N_2 = 5$ , we now study the variation of average accumulated throughput in 8 hours and the average

MLT during 8 hours of operation, with change in input arrival rate. We assume  $\lambda_1 = \lambda_2$  and vary this parameter from 1 per hour (slow arrivals) to 12 per hour (rapid arrivals) in unit steps. The resulting graphs are shown in Figs. 15 and 16. The behavior in these cases is quite interesting. For example, the average accumulated throughput for class A parts reaches a minimum around  $\lambda_1 = \lambda_2 = 7$ , whereas that for class B parts reaches a peak around the same point (Fig. 15). The throughput of class A jobs is found to decrease in the initial ranges of values of arrival rates since in those ranges, the machine tends to produce a long sequence of class B jobs once a switch-over from class A to class B takes place. Again there is appreciable difference between transient and steady-state values, and this difference increases with increase in input arrival rate. This happens since the time to reach steady-state increases when the arrival rate increases.

In Fig. 16, the difference in the transient and steady-state values for class B shows a rather interesting behavior and indicates that steady-state analysis can sometimes lead to wildly inaccurate performance estimates. An unusual behavior observed is that the MLT of class A parts decreases with increase in the arrival rate in certain ranges. This is because of the long waiting times incurred by class A parts while the machine produces a long sequence of class B parts. Some of these trends would change if the initial state of the system is varied.

#### IV. COMPUTATIONAL ISSUES

In transient analysis, we are interested in computing the transition probabilities  $p_{ij}(t)$  or state probabilities  $p_i(t)$  or cumulative performance measures over finite time intervals. To obtain the transition probabilities, we need to solve equations (1) or (2), and to obtain state probabilities, we need to solve equations given by (3). These are coupled, linear, first order, ordinary differential equations. The computation of cumulative measures also involves solving linear differential equations [25]. There are three basic ways in which the above differential equations may be solved:

- 1) Obtain a general solution by deriving and symbolically inverting Laplace transforms. Analytic Laplace transform inversion requires that the eigen values of the infinitesimal generator of the Markov model be accurately determined. If the size of the state space is  $N$ , this would have a worst-case computational complexity of  $O(N^5)$ .
- 2) Evaluate the matrix exponential series (6) directly. This approach is however beset with numerical instabilities, such as severe round off errors [24].
- 3) Numerically solve the differential equations using well developed techniques such as the fourth fifth order Runge-Kutta method, or the TR-BDF2 method (Trapezoid Rule—Second order Backward Difference) [24].

The above methods are not always tractable and other numerical methods have been proposed for transient analysis. Among these, *Uniformization* or *randomization* [72] has assumed prominence as an excellent numerical tool. There are also approximate techniques based on, for example, aggregation and decomposition [27], [28] and diffusion approximations [43]. In some special cases, exact closed form expressions can be obtained for transient measures, such as in acyclic Markov chains [26].

There are excellent review articles dealing with computational aspects of transient analysis. The papers by Grassman [20] and Stewart [21] are two of the earliest ones. More recently, Reibman and Trivedi [24,25] have done a neat survey of numerical transient analysis techniques for transition probabilities, state probabilities, and cumulative measures. The article by Johnson and Malek [73] is a detailed survey on software packages for reliability and availability evaluation; many of these packages, in fact, carry out transient analysis. Much of the following discussion is based on these survey articles.

#### A. Computational Difficulties

There are mainly three problems that one is confronted with in transient analysis: *largeness*, *stiffness*, and *ill-conditioning* [24].

1) *State Space Explosion*: Markov models of real-world manufacturing systems will have a large number of states, often exceeding tens of thousands. So, even an algorithm of low polynomial complexity can become intractable. Also, this will call for a large amount of storage, though, often the matrices are sparse. If the algorithms preserve the sparsity of the matrices involved, savings in storage can be obtained.

2) *Stiffness*: In a manufacturing system, the activities fall into different time scales. For example, operation times are typically small compared to mean time to failure or mean time to repair. Set-up times, depending on the specific system, may be much larger or much smaller than other activity durations. The result is, the transition rates in the Markov chain model will exhibit several orders of magnitude difference. This causes the problem of stiffness. In general, we say a system of differential equations is stiff on the interval  $[0, t]$  if there exists a solution component that has variation on that interval that is large compared to  $\frac{1}{t}$  [24]. A component with large variation changes rapidly relative to the length of the interval. Stiffness makes many integration methods, such as uniformization and Runge-Kutta method, inefficient [74].

3) *Ill-Conditioning*: Manufacturing system models often lead to transition rate matrices that are ill-conditioned. That is, small changes in the matrix elements can produce large changes in the solution. This will lead to inaccurate estimation of transient performance.

#### B. Computational Methods

We shall discuss the computational methods under various heads.

1) *Analysis of Special Classes*: Acyclic Markov chains arise frequently in reliability and performability modeling. Marie, Reibman, and Trivedi [26] have proposed a method for automatically deriving transient solutions that are symbolic in the time duration  $t$ , for acyclic chains. Their method is applicable to cumulative measures of performance and sensitivity analysis of the transient solution. Donatiello and Iyer [75] have proposed a double transform-based procedure for computing performability distributions of systems whose failure-repair behavior is described by acyclic Markov chains. Goyal and Tantawi [76] have proposed a different numerical method for the same problem. In all these cases, the acyclic structure of the Markov model plays a crucial role in the solution procedure.

2) *Laplace Transform Inversion*: This technique was illustrated in Section III-A. This method is good for hand computation on small or special case models. It has a worst-case computational complexity of  $O(N^5)$  where  $N$  is the number of states and requires that the eigen values of the transition rate matrix be accurately determined. For acyclic Markov chains, this technique is adequate, as shown in [26], [75]. Laplace transform inversion using Fourier series [77] is a promising

technique but both analytic and numerical Laplace transform inversions are unstable, in general.

3) *Computation of Matrix Exponential*: For small values of  $t$ , the matrix exponential method gives accurate and efficient solutions for transient analysis. For large values of  $t$ , the exponential series has poor numerical properties even for small problems. Round-off error is a common problem with these computations [20]. There are many alternative ways of evaluating the matrix exponential [78], [79], but they are not efficient for large sized problems and for large values of  $t$ .

4) *Numerical Solution of Differential Equations*: The classical techniques for numerical solution of the differential equations (1), (2), or (6), first find the eigen values and the eigen vectors of the transition rate matrix  $Q$ . The solution is then obtained using the Lagrange-Sylvester formula [80]. This method has complexity of  $O(N^4)$  when all the eigen values are distinct and  $O(N^5)$  otherwise. Thus this approach is impractical for solving large models. Furthermore, for large matrices, it is difficult to accurately generate the entire eigen system.

Numerical differential equation solvers fall into two classes: explicit methods and implicit methods. Explicit methods require only function evaluations, whereas implicit methods require the solution of a linear algebraic system at each time step [29]. The Runge-Kutta method [81] is the most popular explicit method for solving differential equations. This method is widely available and is satisfactory for nonstiff problems with normal accuracy requirements. It is however not suitable for the solution of stiff equations. Popular implicit methods include, the Backwards Euler and the Trapezoid Rule [82]. These methods are very good for handling stiffness, however they are less accurate and incur substantial performance penalties on nonstiff problems.

5) *Uniformization*: Uniformization or randomization [72] is probably the most popular numerical method for transient analysis. In this method, a continuous time Markov chain is reduced to a discrete time Markov chain subordinated to a Poisson process [20], [72]. Uniformization first transforms the transition rate matrix  $Q$  to the matrix  $Q^*$  given by

$$Q^* = \frac{Q}{q} + I$$

where  $q$  is the largest magnitude of a diagonal element of  $Q$ . The solution is then given in the form of an infinite series. The series can be truncated at a desired stage and the error bounds are immediately known. Uniformization is not subject to the round-off errors encountered while directly evaluating the matrix exponential series. It is quite accurate and efficient, and allows accurate error control. It is however not very good for stiff problems.

Uniformization has now emerged as a method of choice for many typical problems in transient analysis. It is extensively used in performability evaluation [32], [33] and sensitivity analysis [64]. It has been implemented in several software packages [73], [29], [65]. Dyer [31] describes an efficient method, based on uniformization, to carry out transient analysis of large Markov chains that arise in reliability, availability, and

repairability modeling. The method uses the special structure of the transition rate matrices arising in such models.

*Aggregation Methods* These methods are approximate and are intended to transform a stiff Markov chain into a nonstiff chain having a smaller state space. Bobbio and Trivedi [27], [28] have proposed an aggregation technique that exploits the stiffness of the chain. In their method, the states are classified into fast and slow states. Fast states are further classified into fast recurrent subsets and a fast transient subset. A separate analysis of each of these fast subsets is done and each fast recurrent subset is replaced by a single slow state while the fast transient subset is replaced by a probabilistic switch. The resulting smaller and nonstiff chain is then analyzed using any suitable method.

*Other Methods* Other methods for transient analysis include, using diffusion approximations [43], fluid approximations [42], and approximate techniques for transform inversion [38].

### C. Software Packages

Johnson and Malek [73] have surveyed several software packages for evaluating reliability, availability, and serviceability. Several of these are useful for transient analysis.

CARE (Computer Aided Reliability Estimator program) [83] is a general purpose reliability estimation tool for large, highly reliable digital fault-tolerant avionics systems. For transient analysis, this package uses the method of convolution integral.

HARP (Hybrid Automated Reliability Predictor) [30] provides a hybrid model for evaluation of reliability and availability of large complex systems. This uses an extended stochastic Petri net model for specifying fault handling and employs the Runge-Kutta method for solving the differential equations.

METASAN (Michigan Evaluation Tool for the Analysis of Stochastic Activity Networks) [84] evaluates performability for non-repairable and repairable systems, over finite intervals of time, by analyzing or simulating a stochastic activity network model, which is an extension of stochastic Petri nets.

SHARPE (Symbolic Hierarchical Reliability and Performance Evaluator) [85] provides a hierarchical modeling framework for evaluating reliability and availability of non-repairable and repairable systems. This uses the technique of Laplace transform inversion for transient analysis.

SAVE (System Availability Estimator) [86] computes reliability and availability of all classes of systems, by doing a transient analysis using the technique of uniformization.

Marie, Reibman, and Trivedi [26] describe an algorithm called ACE (Acyclic Markov chain Evaluator) for evaluating the transition probabilities in symbolic form, for acyclic chains. Reibman, Trivedi, Sanjayakumar, and Ciardo [29] describe a software package for the specification and solution of stiff Markov chains, using the technique proposed by Bobbio and Trivedi [27], [28]. The package ESP (Evaluation Package for Stochastic Petri Nets) [87] is a stochastic Petri net-based package for transient and steady-state analysis. The tool SPNP [65] is a powerful package, developed by Ciardo, Trivedi, and Muppala, that uses stochastic Petri nets as

a specification language and carries out both transient and steady-state analyses. This package uses uniformization for transient analysis and also implements sensitivity analysis.

#### V. SUMMARY

In this article, we have made a case for enhancing research efforts in analyzing the transient performance of discrete manufacturing systems. There are available several computational methods and software tools for conducting transient analysis of Markov models. Application of these methods and tools can facilitate a better understanding of the manufacturing system dynamics and an improved methodology for design. In addition to the issues discussed in this paper, there are certain others that deserve attention of researchers in this area:

- 1) Performance optimization studies using transient analysis.
- 2) Transient analysis of semi-Markov models, M/G/1 type of models, and renewal processes.
- 3) Improved algorithms and numerical techniques for transient analysis, including methods based on aggregation.

#### ACKNOWLEDGMENT

The first author gratefully acknowledges the excellent academic and computing facilities at the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, where he spent seven months, visiting on a INDO-US Science and Technology Fellowship Program.

#### REFERENCES

- [1] Y. C. Ho, "Performance evaluation and perturbation analysis of discrete event dynamical systems," *IEEE Transactions on Automatic Control*, vol. 32, pp. 563-572, 1987.
- [2] N. Viswanadham and Y. Narahari, *Performance Modeling of Automated Manufacturing Systems*. Englewood Cliffs, NJ: Prentice Hall, 1992.
- [3] J. A. Buzacott and J. G. Shantikumar, *Stochastic Models of Manufacturing Systems*. Englewood Cliffs, NJ: Prentice Hall, 1993.
- [4] A. M. Law, "Introduction to simulation: A powerful tool for analyzing complex manufacturing systems," *Industrial Engineering*, vol. 28, no. 5, pp. 46-61, May 1986.
- [5] Y. Dallery and S. B. Gershwin, "Manufacturing flow line systems: A review of models and analytical results," Technical Report 91-002, Laboratory for Manufacturing and Productivity, MIT, April 1992.
- [6] J. A. Buzacott and D. D. Yao, "Flexible manufacturing systems: A review of analytical models," *Management Science*, vol. 32, pp. 890-905, 1986.
- [7] J. A. Buzacott and D. D. Yao, "On queueing network models of flexible manufacturing systems," *Queueing Systems: Theory and Applications*, vol. 1, pp. 5-27, 1986.
- [8] P. Kouvelis and D. Tirupati, "Approximate performance modeling and decision making for manufacturing systems: A queueing network optimization framework," *Journal of Intelligent Manufacturing*, vol. 2, pp. 107-134, 1991.
- [9] N. Viswanadham and Y. Narahari, "Stochastic Petri net models for performance evaluation of automated manufacturing systems," *Information and Decision Technologies*, vol. 14, pp. 125-142, 1988.
- [10] R. Y. Al-Jaar and A. A. Desrochers, "Performance evaluation of automated manufacturing systems using generalized stochastic Petri nets," *IEEE Journal on Robotics and Automation*, vol. 6, no. 6, pp. 621-639, December 1990.
- [11] P. J. Burke, "The output of a queueing system," *Operations Research*, vol. 4, pp. 699-704, 1956.
- [12] J. D. C. Little, "A proof of the queueing formula  $l = \lambda w$ ," *Operations Research*, vol. 9, pp. 383-385, 1961.
- [13] J. R. Jackson, "Job shop like queueing systems," *Management Science*, vol. 10, pp. 131-142, 1963.
- [14] W. J. Gordon and G. F. Newell, "Closed queueing networks with exponential servers," *Operations Research*, 15, pp. 245-260, 1967.
- [15] F. Baskett, K. M. Chandy, R. R. Muntz, and F. Palacios, "Open, closed, and mixed networks of queues with different classes of customers," *Journal of the ACM*, vol. 22, no. 2, pp. 248-260, April 1975.
- [16] M. Reiser and S. S. Lavenberg, "Mean value analysis of closed multichain queueing networks," *Journal of the ACM*, vol. 27, no. 2, pp. 313-322, April 1980.
- [17] P. J. Courtois, "On time and space decomposition of complex structures," *Communications of the ACM*, 28, no. 4, pp. 590-603, April 1985.
- [18] K. S. Trivedi, *Probability and Statistics with Reliability, Queueing, and Computer Science Applications*. Englewood Cliffs, NJ: Prentice Hall, 1982.
- [19] S. M. Ross, *Introduction to Probability Models*. Third Edition, Orlando, Florida: Academic Press, 1985.
- [20] W. K. Grassman, "Transient solution in Markovian queueing systems," *Computers and Operations Research*, vol. 4, no. 1, pp. 47-56, 1977.
- [21] W. Stewart, "A comparison of numerical techniques in Markov modeling," *Communications of the ACM*, vol. 21, no. 2, pp. 144-152, February 1978.
- [22] W. J. Stewart, Ed., in *The Numerical Solution of Markov Chains*. New York: Marcel Dekker, 1991.
- [23] B. Philippe, Y. Saad, and W. J. Stewart, "Numerical methods in Markov chain modeling," *Operations Research*, vol. 40, no. 6, pp. 1156-1179, 1992.
- [24] A. L. Reibman and K. S. Trivedi, "Numerical transient analysis of Markov models," *Computers and Operations Research*, vol. 15, no. 1, pp. 19-36, 1988.
- [25] A. L. Reibman and K. S. Trivedi, "Transient analysis of cumulative measures of Markov chain behavior," *Stochastic Models*, vol. 5, no. 4, pp. 683-710, 1989.
- [26] R. A. Marie, A. L. Reibman, and K. S. Trivedi, "Transient analysis of acyclic Markov chains," *Performance Evaluation*, vol. 7, pp. 175-194, 1987.
- [27] A. Bobbio and K. S. Trivedi, "An aggregation technique for the transient analysis of stiff Markov chains," *IEEE Transactions on Computers*, vol. 35, no. 9, pp. 803-814, 1986.
- [28] A. Bobbio and K. S. Trivedi, "Computing cumulative measures of stiff Markov chains using aggregation," *IEEE Transactions on Computers*, vol. 39, no. 10, pp. 1291-1298, 1990.
- [29] A. L. Reibman, K. S. Trivedi, Sanjayakumar, and G. Ciardo, "Analysis of stiff Markov chains," *ORSA Journal of Computing*, vol. 1, no. 2, pp. 126-133, Spring 1989.
- [30] S. Bavuso, J. B. Dugan, K. S. Trivedi, E. Rothmann, and W. Smith, "Analysis of some fault-tolerant architectures using HARP," *IEEE Transactions on Reliability*, vol. 36, no. 2, pp. 176-185, 1987.
- [31] D. Dyer, "Unification of reliability, availability, and repairability models for Markov systems," *IEEE Transactions on Reliability*, vol. 38, no. 2, pp. 246-252, 1989.
- [32] E. de Souza e Silva and H. R. Gail, "Calculating availability and performance measures of repairable computer systems using randomization," *Journal of the ACM*, vol. 36, no. 1, pp. 171-193, 1989.
- [33] E. de Souza e Silva and H. R. Gail, "Performance analysis of computer systems: From model specification to solution," *Performance Evaluation*, vol. 14, no. 3, pp. 157-196, 1992.
- [34] D. K. Gerber, "Performance evaluation of fault-tolerant systems using transient Markov models," Masters Thesis, Department of EECS, Massachusetts Institute of Technology, 1985.
- [35] J. F. Meyer, "Performance: A retrospective and some pointers to the future," *Performance Evaluation*, vol. 14, pp. 139-156, 1992.
- [36] K. S. Trivedi, J. K. Muppala, S. P. Woollet, and B. R. Haverkort, "Composite performance and dependability analysis," *Performance Evaluation*, vol. 14, pp. 197-215, 1992.
- [37] C. Baiocchi, A. C. Capelo, V. Comincioli, and G. Serazzi, "A mathematical model for transient analysis of computer systems," *Performance Evaluation*, vol. 3, pp. 247-264, 1983.
- [38] T. C. T. Kotiah, "Approximate transient analysis of some queueing systems," *Operations Research*, vol. 26, no. 2, pp. 333-346, 1978.
- [39] P. Konstantopoulos and F. Baccelli, "On the cut-off phenomenon in some queueing systems," Technical Report 90-1290, INRIA, October 1990.
- [40] S. K. Tripathi and A. Duda, "Time dependent analysis of queueing systems," *INFOR*, vol. 15, 1986.
- [41] R. A. Upton and S. K. Tripathi, "An approximate transient analysis of the M(t)/M/1 queue," *Performance Evaluation*, vol. 2, no. 2, pp. 118-132, 1982.
- [42] A. A. Weiss and D. Mitra, "A transient analysis of a data network with a processor sharing switch," *AT & T Technical Journal*, vol. 67, no. 4, pp. 4-16, 1988.

- [43] H. Kobayashi, "Application of the diffusion approximation to queueing networks: Part 2. Non-equilibrium distribution and computer modeling," *Journal of the ACM*, vol. 21, pp. 459-469, 1974.
- [44] R. Ram, "Performance and performance modeling of automated manufacturing systems," Ph.D. Dissertation, Department of Computer Science and Automation, Indian Institute of Science, February 1992.
- [45] V. Gopalakrishna, "Performance-reliability analysis of multiclass manufacturing systems," Doctoral Dissertation (in preparation), Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India, 1993.
- [46] N. Viswanadham and R. Ram, "Composite performance-dependability analysis of cellular manufacturing systems," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 2, 1994.
- [47] R. Ram and N. Viswanadham, "Performability analysis of automated manufacturing systems with centralized material handling," *International Journal of Production Research*. To appear.
- [48] N. Viswanadham, Y. Narahari, and R. Ram, "Performability of automated manufacturing systems," *Control and Dynamic Systems: Volume 47*. Academic Press, Inc., 1991, pages 77-122.
- [49] N. Viswanadham, K. R. Pattipati, and V. Gopalakrishna, "Performability studies of AMSs with multiple part types, in *Proceedings of IEEE International Conference on Robotics and Automation, Georgia, Atlanta, USA*, IEEE Press, 1993.
- [50] G. J. Miltenburg, "Variance of the number of units produced on a transfer line with buffer inventories during a period of length T," *Naval Research Logistics Quarterly*, vol. 34, pp. 811-822, 1987.
- [51] S. B. Gershwin, "Variance of output of a tandem production system," in *Proceedings of the Second International Workshop on Queueing Networks with Finite Capacity* Research Triangle Park, NC, USA, May 1992, pp. 366-405.
- [52] R. P. Malhame and El-Kebir Boukas, "A renewal theoretic analysis of a class of manufacturing systems," *IEEE Transactions on Automatic Control*, vol. 36, no. 5, pp. 580-587, May 1991.
- [53] R. A. Wolff, *Stochastic Processes and Queueing Theory*. Englewood Cliffs, NJ: Prentice Hall, 1989.
- [54] J. G. Shantikumar and J. A. Buzacott, "Open queueing network models of dynamic job shops," *International Journal of Production Research*, vol. 19, pp. 255-266, 1981.
- [55] D. Mitra and I. Mitrani, "Analysis of a Kanban discipline for cell coordination in production lines, part 2: Stochastic demands," *Operations Research*, vol. 39, no. 5, pp. 807-823, 1991.
- [56] T. Rolski, "Queues with non-stationary inputs," *Queueing Systems: Theory and Applications*, vol. 5, pp. 113-130, 1989.
- [57] P. R. Kumar, "Re-entrant lines," Technical report, Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 1992.
- [58] S. H. Lu and P. R. Kumar, "Distributed scheduling based on due dates and buffer priorities," *IEEE Transactions on Automatic Control*, vol. 36, no. 12, pp. 1406-1416, December 1991.
- [59] L. M. Wein, "Scheduling semiconductor wafer fabrication," *IEEE Transactions on Semiconductor Manufacturing*, vol. 1, no. 3, pp. 115-130, August 1988.
- [60] S. X. Bai and S. B. Gershwin, "A manufacturing scheduler's perspective on semiconductor fabrication," Technical Report 89-518, MIT Microsystems Research Center, March 1989.
- [61] G. D. Stamoulis, "Transient Analysis of Some Open Queueing Systems," Masters Thesis, Department of EECS, MIT, 1988.
- [62] G. D. Stamoulis and J. D. Tsitsiklis, "On the settling time of the G/G/1 queue," *Advances Applied Probability*, vol. 22, pp. 929-956, 1990.
- [63] V. Anantharam, "The settling time of a closed Jackson network," Technical report, Preprint, 1988.
- [64] P. Heidelberger and A. Goyal, "Sensitivity analysis of continuous time Markov chains using uniformization," in *Proceedings of the Second International Workshop on Applied Mathematics and Performance/Reliability Models of Computer and Communication Systems*, May 1987.
- [65] G. Ciardo, J. Muppala, and K. S. Trivedi, "SPNP: Stochastic Petri net package," in *Proceedings of the Third International Workshop on Petri Nets and Performance Models, Tokyo, Japan*, IEEE Computer Society Press, 1989, pp. 142-151.
- [66] V. Anantharam, "Threshold phenomena in the transient behavior of Markovian models of communication networks and databases," *Queueing Systems: Theory and Applications*, vol. 5, pp. 77-98, 1989.
- [67] U. Sumita, J. G. Shantikumar, and Y. Masuda, "Analysis of fault-tolerant computer systems," *Microelectronics and Reliability*, vol. 27, no. 1, pp. 65-78, 1987.
- [68] J. Kemeny and W. Snell, *Finite Markov Chains*. Van Nostrand, 1960.
- [69] B. Gnedenko et al., *Mathematical Methods in Reliability Theory*. Academic Press, 1969.
- [70] J. A. Buzacott, "Automatic transfer lines with buffer stocks," *International Journal of Production Research*, vol. 5, pp. 183-200, 1967.
- [71] J. A. Buzacott, "The Markov approach to finding the failure times of repairable systems," *IEEE Transactions on Reliability*, vol. 9, pp. 128-134, 1970.
- [72] D. Gross and D. Miller, "The randomization technique as a modeling tool and solution procedure for transient Markov processes," *Operations Research*, vol. 32, pp. 334-361, 1984.
- [73] A. M. Johnson, Jr. and M. Malek, "Survey of software tools for evaluating reliability, availability, and serviceability," *ACM Computing Surveys*, vol. 20, no. 4, pp. 227-269, December 1988.
- [74] W. L. Miranker, *Numerical Methods for Stiff Equations and Singular Perturbation Problems*. Dordrecht, Holland: D. Reidel, 1981.
- [75] L. Donatiello and B. R. Iyer, "Analysis of a composite performance reliability measure for fault-tolerant systems," *Journal of the ACM*, vol. 34, no. 1, pp. 179-199, 1987.
- [76] A. Goyal and A. Tantawi, "Evaluation of performability for degradable computer systems," *IEEE Transactions on Computers*, vol. 36, no. 6, pp. 738-744, 1987.
- [77] V. Kulkarni, V. F. Nicola, R. M. Smith, and K. S. Trivedi, "Numerical evaluation of performability measures and job completion times in repairable fault-tolerant systems," in *Proceedings of the Sixteenth IEEE International Symposium on Fault-Tolerant Computing Systems*, July 1986.
- [78] C. Moler and C. F. Van Loan, "Nineteen dubious ways to compute the exponential of a matrix," *SIAM Review*, vol. 20, pp. 801-835, 1978.
- [79] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, Madison: Johns Hopkins University Press, 1983.
- [80] R. Bellman, *Introduction to Matrix Analysis*. New York: McGraw-Hill, 1969.
- [81] C. W. Gear, *Numerical Initial Value Problems in Ordinary Differential Equations*. Englewood Cliffs, NJ: Prentice Hall, 1971.
- [82] J. D. Lambert, *Computational Methods in Ordinary Differential Equations*. London: Wiley, 1973.
- [83] R. Geist and K. S. Trivedi, "Ultra-high reliability prediction for fault-tolerant computer systems," *IEEE Transactions on Computers*, vol. 32, no. 12, pp. 1118-1127, 1983.
- [84] W. H. Sanders and J. F. Meyer, "METASAN: A performability evaluation tool based on stochastic activity networks," *Proceedings of the 1986 Fall Joint Computer Conference*, November 1986, pages 807-816.
- [85] R. A. Sahner and K. S. Trivedi, "Reliability modeling using SHARPE," *IEEE Transactions on Reliability*, vol. 36, no. 2, pp. 186-193, June 1987.
- [86] A. Goyal, S. S. Lavenberg, and K. S. Trivedi, "Probabilistic modeling of computer system availability," *Annals of Operations Research*, vol. 8, pp. 285-306, 1987.
- [87] A. Cumani, "ESP: A package for the evaluation of stochastic Petri nets with phase type distributed transition times," in *Proceedings of First International Workshop on Timed Petri Nets, Torino, Italy*, July 1985.



Y. Narahari received the Ph.D. degree from the Indian Institute of Science, where his doctoral dissertation, "Petri Net Models of Flexible Manufacturing Systems" was awarded the Gold Medal for the best thesis in 1988. He has written several papers on Petri net models of manufacturing systems, deadlock analysis in manufacturing systems, and performance analysis of discrete event systems. He recently co-authored a book entitled "Performance Modeling of Automated Manufacturing Systems," published by Prentice-Hall, Englewood Cliffs, NJ, in 1992.

Dr. Narahari is currently an Assistant Professor at the Indian Institute of Science, Department of Computer Science and Automation. From July 1992 to January 1993, he was a Visiting Scientist at the Laboratory of Information and Decision Systems at the Massachusetts Institute of Technology, Cambridge, MA. His research interests are in the areas of performance modeling and evaluation, dynamic and stochastic scheduling, and analysis of discrete event systems.



N. Viswanadham, (SM'86-F'93), received the Ph.D. degree in 1970 from the Indian Institute of Science, Bangalore, India. N. Viswanadham is a TataChem Professor and Chair of the Department of Computer Science and Automation, of the Indian Institute of Science.

Dr. Viswanadham has held visiting appointments at several North American universities. He was a GE Research Fellow at the corporate research center in 1989. His current research interests are in the areas of fault tolerant control system design and modeling and control of flexible manufacturing systems. He is the author of several journal articles and conference papers. He is a joint author of two textbooks: *Reliability in Computer and Control Systems* (North-Holland, 1987) and *Performance Modeling of Automated Manufacturing Systems* (Prentice-Hall, 1992). He is co-editor of four other books. He is currently editor of *Sadhana: Academy Proceedings in Engineering Sciences*. He is also Associate Editor of the journals *Control Theory and Advanced Technology*, *Journal of Manufacturing Systems*, *Information and Decision Technologies*, *Intelligent and Robotic Systems*, *Asia-Pacific Engineering Journal*, and *Systems Automation: Research and Applications*. He was Associate Editor-at-Large for the *IEEE Transactions on Automatic Control*, in 1990 and 1991. He is a Fellow of the IEEE, The Indian National Science Academy, the Indian Academy of Sciences, and the Indian National Academy of Engineering.