

Markovian models for deadlock analysis in automated manufacturing systems

Y NARAHARI, N VISWANADHAM and K R KRISHNA PRASAD

Department of Computer Science & Automation, Indian Institute of Science, Bangalore 560 012, India

Abstract. Deadlocks constitute a major issue in the design and operation of discrete event systems. In automated manufacturing systems, deadlocks assume even greater importance in view of the automated operation. In this paper, we show that Markov chains with absorbing states provide a natural model of manufacturing systems with deadlocks. With illustrative examples, we show that performance indices such as mean time to deadlock and mean number of finished parts before deadlock can be efficiently computed in the modelling framework of Markov chains with absorbing states. We also show that the distribution of time to deadlock can be computed by conducting a transient analysis of the Markov chain model.

Keywords. Markovian models; deadlock analysis; automated manufacturing systems; discrete event systems; mean time to deadlock.

1. Introduction

A typical automated manufacturing system (AMS) comprises a computer controlled configuration of numerically controlled (NC) machine tools and a material handling system (MHS) designed to simultaneously manufacture low to medium volumes of high quality products at low cost. The architecture of such a system is constituted by

- (a) versatile NC machines which can carry out a variety of machining operations,
- (b) an automated MHS comprising conveyors, or automated guided vehicles (AGV) to move parts and tools between machines,
- (c) a hierarchical control system that coordinates the actions of the machines, the MHS, and the workpieces,
- (d) a load/unload workstation through which the entry and exit of a part occurs; a fixturing station where the parts entering the system are fixtured onto pallets; inspection stations; and coordinate measurement machines, and
- (e) a buffer storage in the form of local storage or central storage or both, to store raw and semi-finished workpieces.

An AMS can be described as a discrete event dynamical system (Ho 1987) because changes in the system state are caused by the occurrence of events at discrete instants of time instead of continuously. Therefore the evolution of states in an AMS is not

completely described by partial or ordinary differential equations, as in the case of continuous variable dynamical systems. Examples of discrete events in an AMS are: entry/exit of a part, starting/finishing of part transfer by a robot, starting/finishing of processing by a machine, a robot failure, and a machine breakdown. The number of such activities in a typical AMS is large and there are numerous interactions involving these activities. Further, these interactions exhibit concurrency, contention for resources, synchronization, and randomness. The interactions might also lead to a deadlocked state, in which the system is crippled and produces no output. Any effective model of an AMS should therefore capture the above characteristics.

The need for and the role of mathematical models in automated manufacturing systems has been clearly brought out by Suri (1985) and by Buzacott & Yao (1986). AMS models can be used in a variety of ways such as in logical analysis, performance prediction, optimization, control, and for gaining an insight into the system.

There are two basic types of models of discrete event dynamical systems and hence AMS: qualitative models and quantitative models. Qualitative models such as Petri nets (Narahari & Viswanadham 1985), finite state machines (Wonham 1987), and general algebraic discrete event models (Inan & Varaiya 1989) are useful for investigating logical aspects of FMS behaviour such as boundedness, fairness, absence/existence of deadlocks, mutual exclusion of resources, and correctness of control logic. Quantitative models are basically stochastic models that address quantitative issues of performance such as throughput, manufacturing lead time, machine/robot/AGV utilizations, and reliability. There are three basic types of quantitative modelling tools: simulation models, analytical models, and hybrid models, which use both simulation and analytical models. In this paper, we are concerned with analytical models of automated manufacturing systems.

Markov chains constitute the basic stochastic model of automated manufacturing systems. For real-life systems, Markov chain models have a large state space, entailing large scale computation for transient or steady-state analysis. Further, it is often difficult to generate the Markov chain model directly from the system. Queueing network (QN) models (Buzacott & Yao 1986) effectively overcome the state space explosion problem inherent in Markovian modelling. QN models capture the dynamics, interactions, and uncertainties in the system in an aggregate way and the performance measures produced are average values which assume a steady-state operation of the system. The computational efficiency of QN models applies only to a special class of QN called product form queueing networks. For non-product form systems, exact modelling can only be done via explicit Markov chain analysis. The generation of a Markov chain model can be completely automated with the help of generalized stochastic Petri nets (GSPN) (Marsan *et al* 1986). GSPN constitute a high level modelling tool which can elegantly capture the characteristics of discrete event dynamical systems.

In this paper, we show that the framework of Markov chains with absorbing states can be used to model AMS with deadlocks.

2. Deadlocks in automated manufacturing systems

Deadlocks constitute an undesirable phenomenon in AMS. A deadlock is a situation where each of a set of two or more parts keeps waiting indefinitely for the other parts in the set to release resources. In the context of AMS, resources refer to machines,

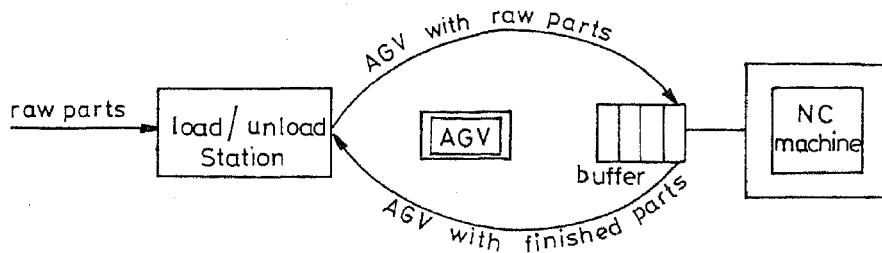


Figure 1. A simple AMS to illustrate deadlock.

buffers, AGV, robots, fixtures, tools etc. Deadlocks may arise as the final state of a complex sequence of operations on several parts passing concurrently through a system, and thus are generally difficult to predict. Deadlocks lead to degraded performance and ultimately zero throughput. The automated operation of an AMS makes deadlocks an important problem in the design of these systems. In an improperly designed AMS, the only remedy for a deadlock may be manual clearing of machines or AGV or buffers, and restart of the system from an initial condition which is known to produce deadlock-free operation under nominal production conditions. The loss of production and the labour cost in resetting the system in this way can be avoided by proper design. Often, deadlocks may result due to simple software bugs in the controllers.

Example 1. To visualize a deadlock, consider the simple AMS depicted in figure 1. There is a load/unload (L/U) station at which raw parts arrive randomly. An AGV carries a raw part from the L/U station to an NC machine which carries out some operations on the raw part. The finished part is carried by the AGV to the L/U station where the finished part is unloaded. After depositing a raw part on the NC machine, the AGV travels back to the L/U station. It will pick up a raw part if one is available and travel to the NC machine. It is assumed that the AGV can only carry one part at a time. Also the AGV takes a negligible amount of time to travel from the L/U station to the machine and vice versa. Imagine the following sequence of events, starting with an initial state in which the AGV and the machine are free and raw parts are available: (i) The AGV carries a raw part, say part 1, and loads it onto the NC machine, which starts processing part 1. (ii) The AGV returns to the L/U station. Before the machine finishes processing, another raw part arrives and the AGV carries the raw part, say part 2, to the machine, but waits for the machine which is still processing part 1. Thus the AGV gets blocked, waiting for the machine. (iii) The machine finishes the operations on part 1 and starts waiting for the AGV to carry the finished part 1 to the L/U station. At this juncture, the machine gets blocked waiting for the AGV. If the machine and the AGV can only accommodate one part at a time and there is no additional buffer space, then the two resources here are involved in a deadlock since each keeps waiting for the other indefinitely. Figure 2 gives a pictorial description of the above sequence of events. Even if some buffer space is provided for raw parts and finished parts in the above system, a deadlock can still occur because the AGV can fill the entire buffer with raw parts during the processing of part 1 by the machine.

Important performance issues in AMS with deadlocks would be: Mean time before deadlock, mean throughput before deadlock, probability of a particular deadlock, and so on. In this paper, we show that a natural model for deadlocks is provided by the absorbing states of a Markov chain. The above performance issues can be effectively addressed using the theory of Markov chains with absorbing states (Trivedi 1982).

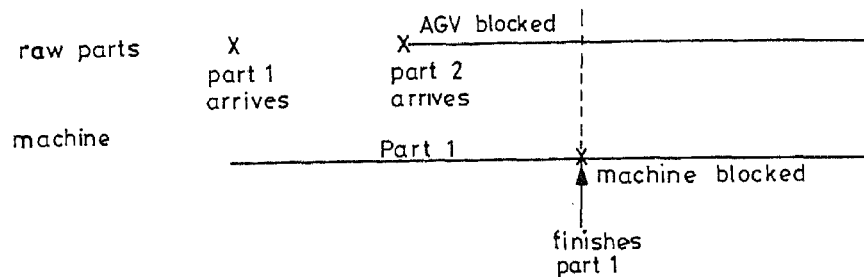


Figure 2. Sequence of events leading to deadlock.

3. Modelling of deadlocks by absorbing states

Here, we present the relevant theory of Markov chains with absorbing states (Trivedi 1982) and show with an example, the application to performance evaluation in the presence of deadlocks.

Let the state space of a finite discrete time Markov chain (DTMC) $\{X_n: n = 0, 1, 2, \dots\}$ be given by

$$S = \{1, 2, \dots, m, m+1, m+2, \dots, m+n\},$$

where $m > 0, n > 0$, the first m states are transient states, and the rest of the states are absorbing states. The transition probability matrix (TPM) of the above DTMC may be partitioned as

$$P = \begin{bmatrix} T & C \\ \mathbf{0} & I \end{bmatrix}.$$

In the above, T is an $(m \times m)$ matrix that describes single step transitions among transient states; I is the identity matrix of order n ; $\mathbf{0}$ is an $(n \times m)$ zero matrix; and C is an $(m \times n)$ matrix that describes transitions from transient states into absorbing states. The k -step TPM can be seen to be

$$P^k = \begin{bmatrix} T(k) & C(k) \\ \mathbf{0} & I \end{bmatrix},$$

where

$$T(k) = T^k$$

and

$$C(k) = (T^{k-1} + T^{k-2} + \dots + T + I)C.$$

Note that $[T^k]_{ij}$, $1 \leq i, j \leq m$, is the probability of arriving in transient state j , starting from transient state i , after exactly k steps.

It can be shown that the $(\lim_{t \rightarrow \infty} \sum_{k=0}^t T^k)$ exists and is given by the matrix $F = (I - T)^{-1}$.

F is called the Fundamental Matrix and is a rich source of information about the given Markov chain with absorbing states. It can be shown (Trivedi 1982) that the (i, j) th element of the fundamental matrix gives the mean number of times transient state j is visited, starting from transient state i , before the Markov chain reaches any absorbing state. Note that

$$F = I + FT.$$

Thus the matrix F can be computed by solving the above system of linear simultaneous equations.

3.1 Mean time to deadlock

Let state 1 be the initial (transient) state. Then for $j = 1, 2, \dots, m$, f_{1j} would give the mean number of visits to state j before deadlock. If m_j is the mean sojourn time in state j , then the mean time to deadlock

$$= \sum_{j=1}^m f_{1j} m_j.$$

More generally, let transient state j be the initial state with probability p_j , where

$$\sum_{j=1}^m p_j = 1.$$

Then, the mean number of visits to state j before deadlock

$$= \sum_{i=1}^m p_i f_{ij}.$$

In this case, the mean time to deadlock

$$= \sum_{j=1}^m \left(\sum_{i=1}^m p_i f_{ij} \right) m_j.$$

Note that the above results would hold for a continuous time Markov chain, by interpreting the above DTMC as an embedded Markov chain.

3.2 Probability of absorbing states

Let us look at the matrices

$$C(k) = (T^{k-1} + T^{k-2} + \dots + T + I)C, \text{ and } G = FC.$$

The (i, j) th element of the matrix $C(k)$, $1 \leq i \leq m$, $1 \leq j \leq n$, gives the probability that the Markov chain would reach the absorbing state $m + j$, starting from transient state i , in exactly k steps. g_{ij} , the (i, j) th element of G , can be interpreted as the long term probability that the Markov chain reaches the absorbing state $m + j$, starting from transient state i .

Therefore, if state 1 is the initial state, then $g_{11}, g_{12}, \dots, g_{1n}$ would give the long term probabilities of the Markov chain ending up in the absorbing states $m + 1, m + 2, \dots$, and $m + n$, respectively.

On the other hand, if p_1, p_2, \dots , and p_m are the probabilities that the initial state is the state $1, 2, \dots$, and m respectively, then, for $j = 1, 2, \dots, n$, the probability of ending up in absorbing state $(m + j)$ is given by

$$\sum_{i=1}^m p_i g_{ij}.$$

The analysis of a continuous time Markov chain (CTMC) with absorbing states will

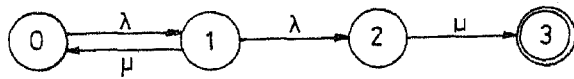


Figure 3. Markov chain model for example 2.

proceed in the same way, except that we have to work with the embedded Markov chain.

Example 2. We now study the system of example 1. Let the raw parts arrive according to a Poisson process with rate λ and let the processing time be exponentially distributed with rate μ . Assume that there is no buffer in front of the NC machine. Then the above system can be modelled as a CTMC with four states $\{1, 2, 3, 4\}$ having the following interpretation.

- 1: Machine idle, AGV idle, waiting for next raw part;
- 2: machine processing a part, no raw parts waiting;
- 3: machine processing a part, AGV waiting at the machine with a raw part;
- 4: machine, with a finished part, waiting for the AGV, and the AGV, with a raw part, waiting for the machine.

The state transition diagram of the above CTMC is shown in figure 3. If there is a buffer with capacity $n(n > 0)$ in front of the machine, then the state space is $\{1, 2, 3, \dots, n+3, n+4\}$, with proper interpretation. The state transition diagram of this CTMC is shown in figure 4. Note that there is exactly one absorbing state, namely $n+4$, and the rest are transient states.

The TPM of the embedded Markov chain of the CTMC of figure 3 is given by

$$P = \begin{bmatrix} 0 & 1 & 0 & | & 0 \\ p & 0 & 1-p & | & 0 \\ 0 & 0 & 0 & | & 1 \\ \hline 0 & 0 & 0 & | & 1 \end{bmatrix},$$

where

$$p = \mu/(\lambda + \mu).$$

Here

$$T = \begin{bmatrix} 0 & 1 & 0 \\ p & 0 & 1-p \\ 0 & 0 & 0 \end{bmatrix} \text{ and } C = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The fundamental matrix can be computed to be given by

$$F = \begin{bmatrix} \frac{1}{1-p} & \frac{1}{1-p} & 1 \\ \frac{1}{1-p} & \frac{1}{1-p} & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Thus, state 1 and state 2 are each visited $1/(1-p)$ times, on an average, before the occurrence of a deadlock. That is, on an average, $1/(1-p)$ parts are produced by the system before the deadlock. If $\mu = 8$ parts/hour and $\lambda = 2$ parts per hour then $p = 0.8$ and hence on the average 5 parts are produced before deadlock.

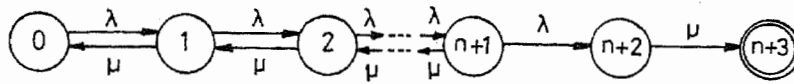


Figure 4. Markov chain model, with a buffer of capacity n .

The mean sojourn times, $m_j (j = 0, 1, 2)$, in the states 0, 1, and 2 can be seen to be

$$m_0 = 1/\lambda; \quad m_1 = 1/(\lambda + \mu); \quad \text{and} \quad m_2 = 1/\mu.$$

Therefore, the mean time to deadlock

$$\begin{aligned} &= \left(\frac{1}{1-p}\right)\left(\frac{1}{\lambda}\right) + \left(\frac{1}{1-p}\right)\left(\frac{1}{\lambda + \mu}\right) + 1\left(\frac{1}{\mu}\right) \\ &= 3.125 \text{ hours.} \end{aligned}$$

From this we can compute the utilization of the machine. If the machine is utilized 100%, then the throughput rate is 8 parts/hour. However, since only 5 parts are produced, the machine is utilized for a mean total of 5/8 hours out of 3.125 hours. Hence machine utilization

$$= \frac{5}{8(3.125)} = \frac{5}{25} = \frac{1}{5}.$$

If we want to increase the mean time to deadlock, thereby increasing the number of parts produced before deadlock, then we can introduce a buffer where raw parts and/or finished parts are deposited. The resulting CTMC has been shown in figure 4. It will be interesting to solve this CTMC for different values of n and to notice that the mean time to deadlock is an increasing function of the capacity of the buffer. Figure 5 shows a graph of the mean time to deadlock versus the capacity of the buffer.

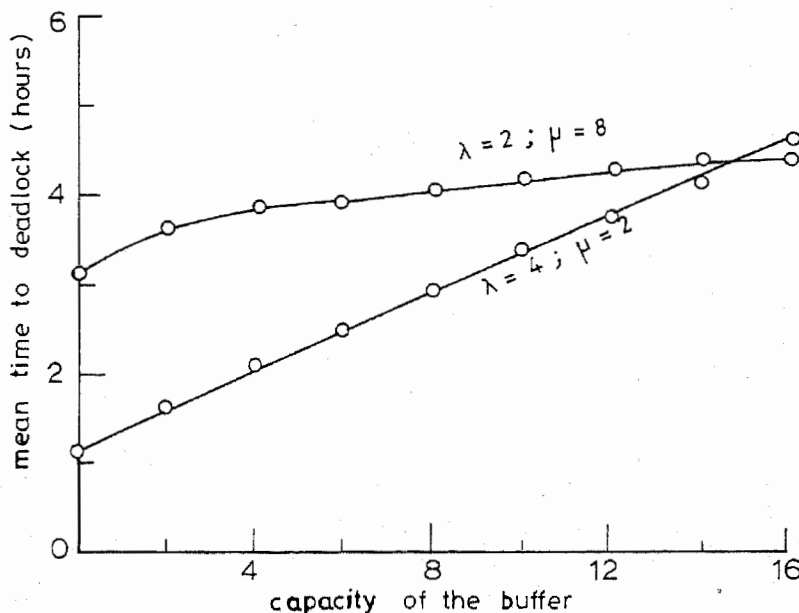


Figure 5. Variation of mean time to deadlock with buffer capacity.

4. Distribution of time to deadlock

In the previous section, it has been shown how the mean time to deadlock can be computed from the fundamental matrix of the Markov chain. If we are interested in the distributions of the time to deadlock, then we need to conduct a transient analysis of the Markov chain model. Let us again consider a CTMC with absorbing states $\{X(t): t \geq 0\}$, with state space $\{1, 2, \dots, m, m+1, \dots, m+n\}$, as in the previous section. Let T be the time to deadlock.

Let

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

Then assuming 1 as the initial state, we have

$$\begin{aligned} P\{T > t\} &= P\{X(t) \neq m+j, \text{ for any } j = 1, 2, \dots, n\}, \\ &= 1 - \sum_{j=1}^n p_{1, m+j}(t). \end{aligned}$$

Hence the distribution function of T is given by

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

The individual probabilities $p_{1, m+j}(t)$ for each j have to be computed by solving the Chapman-Kolmogorov differential equations (Ross 1985). We show the computation of $F_T(t)$ for the system of example 1, by employing Laplace transforms. Such direct methods however do not work for large and complex systems, in which numerical techniques such as uniformization (Gross & Miller 1983) need to be employed.

Example 3. For the Markov chain of figure 3, note that the time T to deadlock has the distribution

$$F_T(t) = p_{14}(t).$$

To compute $p_{14}(t)$, we consider the backward Chapman-Kolmogorov equation for $p_{14}(t)$ (Ross 1985),

$$\frac{d}{dt}(p_{14}(t)) = q_{11}p_{14}(t) + q_{12}p_{24}(t) + q_{13}p_{34}(t) + q_{14}p_{44}(t).$$

We know that $q_{11} = -\lambda$; $q_{12} = \lambda$; $q_{13} = 0$; and $q_{14} = 0$. Therefore we have

$$\frac{d}{dt}(p_{14}(t)) = -\lambda p_{14}(t) + \lambda p_{24}(t). \quad (1)$$

The backward equation for $p_{24}(t)$ is given by

$$\frac{d}{dt}(p_{24}(t)) = q_{21}p_{14}(t) + q_{22}p_{24}(t) + q_{23}p_{34}(t) + q_{24}p_{44}(t).$$

Since $q_{21} = \mu$; $q_{23} = \lambda$; $q_{22} = -(\lambda + \mu)$; and $q_{24} = 0$, the above simplifies to

$$\frac{d}{dt}(p_{24}(t)) = \mu p_{14}(t) - (\lambda + \mu)p_{24}(t) + \lambda p_{34}(t). \quad (2)$$

The backward equation for $p_{34}(t)$ is

$$\frac{d}{dt}(p_{34}(t)) = -\mu p_{34}(t) + \mu p_{44}(t) = -\mu p_{34}(t) + \mu. \tag{3}$$

We shall solve for $p_{14}(t)$ by the Laplace transform method. Let $P_{ij}(s)$ denote the Laplace transform of $p_{ij}(t)$. Taking the Laplace transform on both sides of (1), (2), and (3), we get respectively

$$sP_{14}(s) = -\lambda P_{14}(s) + \lambda P_{24}(s), \tag{4}$$

$$sP_{24}(s) = \mu P_{14}(s) - (\lambda + \mu)P_{24}(s) + \lambda P_{34}(s), \tag{5}$$

$$sP_{34}(s) = -\mu P_{34}(s) + \mu/s. \tag{6}$$

Simplifying using (4), (5), and (6), we get

$$P_{14}(s) = (\lambda^2 \mu) / \{s(s + \mu)(s^2 + s(2\lambda + \mu) + \lambda^2)\}, \tag{7}$$

which can further be simplified as

$$P_{14}(s) = k / \{s(s + \mu)(s + a)(s + b)\},$$

where the constants k , a , and b depend on λ and μ and can be evaluated easily. Note the pole at $s = 0$ in $P_{14}(s)$. It is to be noted that we are employing here the Laplace transform and not the Laplace–Stieltjes transform. The cumulative distribution function (CDF) of the time to deadlock can now be obtained by taking the inverse Laplace transform. The mean time to deadlock can be obtained by simply differentiating $P_{14}(s)$ and evaluating at $s = 0$. Thus, the solution can be written as

$$p_{14}(t) = A + Be^{-\mu t} + Ce^{-at} + De^{-bt},$$

where A , B , C , and D are constants. Since we know that

$$p_{14}(t) = 0, \quad \text{at } t = 0, \\ = 1, \quad \text{at } t = \infty,$$

we have $A = 1$ and $A + B + C + D = 0$.

Thus,

$$p_{14}(t) = 1 + Be^{-\mu t} + Ce^{-at} + De^{-bt},$$

where $B + C + D = -1$.

The above gives the CDF of the time to deadlock. The mean of this random variable has already been shown to be equal to

$$\left(\frac{1}{1-p}\right)\left(\frac{1}{\lambda}\right) + \left(\frac{1}{1-p}\right)\left(\frac{1}{\lambda + \mu}\right) + \frac{1}{\mu},$$

where $p = \mu/(\lambda + \mu)$. Thus the mean time to deadlock can be obtained easily either by differentiating the Laplace transform or by using the fundamental matrix. It is however not so easy to obtain the CDF of the time to deadlock. In general, only transient analysis can help us compute the CDF of the time to deadlock.

5. Concluding remarks

In this paper, we have shown that performance analysis of discrete event systems with deadlocks can be efficiently carried out in the modelling framework of Markov chains with absorbing states. Specifically, we have shown that: (1) mean values of performance measures such as time to deadlock can be computed using the fundamental matrix of the Markov chain, and (2) distributions of these performance measures can be computed by conducting a transient analysis of the Markov chain. A simple illustrative example has been provided to demonstrate the computation of these performance measures. Other interesting examples have been worked out in Krishna Prasad (1990) and Manjunath (1991).

5.1 Computational issues

From the computational viewpoint, the computation of mean performance measures entails solving a system of linear simultaneous equations for obtaining the fundamental matrix. To compute the distributions via transient analysis, the Laplace transform method, illustrated in example 3, is feasible only for small sized Markov chains. For real-life FMS, the following numerical methods can be used.

- (1) Numerical computation of the matrix exponential;
- (2) Runge-Kutta method for solving ordinary differential equations;
- (3) uniformization.

There are several methods for evaluating the matrix exponential series (Moler & Van Loan 1978). However, for large Markov chains, these methods are beset with numerical instabilities such as round-off errors. For large Markov chains, the techniques based on the fourth-fifth order Runge-Kutta-Fehlberg method (RK45 method) (Reibman & Trivedi 1988) and the uniformization or randomization method (Gross & Miller 1983) have proved to be quite efficient. Reibman & Trivedi (1988) have discussed the relative merits of these two methods.

For real-life automated manufacturing systems, another problem to be tackled is the generation of a complete and correct Markov model of the given system. Generalized stochastic Petri nets (Viswanadhan & Narahari 1988) provide a convenient high-level modelling framework for automatically generating the underlying Markov chain. Generalized stochastic Petri nets also provide a convenient framework for detecting deadlocks (absorbing states) in the physical system (Markov model), since the first step in the GSPN analysis is to generate all the states of the underlying Markov chain.

References

- Buzacott J A, Yao D D 1986 Flexible manufacturing systems: A review of analytical models. *Manage. Sci.* 32: 890-905
- Gross D, Miller D R 1983 The randomization technique as a modelling tool and solution procedure for transient Markov processes. *Oper. Res.* 32: 343-361
- Ho Y C 1987 Performance evaluation and perturbation analysis of discrete event systems. *IEEE Trans. Autom. Control* AC32: 563-572
- Inan K M, Varaiya P P 1989 Algebras of discrete event models. *Proc. IEEE* 77: 24-38
- Krishna Prasad K R *Performance analysis of manufacturing systems with deadlocks*, B E Project Report, Dept. of Computer Science and Automation, Indian Institute of Science, April 1990

- Manjunath M S 1991 *Transient analysis of flexible Manufacturing Systems*, M E Project Report, Dept. of Computer Science and Automation, Indian Institute of Science, Bangalore
- Marsan M A, Balbo G, Conte G 1986 *Performance models of multiprocessor systems* (Cambridge, Mass: The MIT press)
- Moler C, Van Loan C F 1978 Nineteen dubious ways to compute the exponential of a matrix. *SIAM Rev.* 20: 801-835
- Narahari Y, Viswanadham N 1985 A Petri net approach to modelling and analysis of flexible manufacturing systems. *Ann. Oper. Res.* 3: 449-472
- Reibman A, Trivedi K S 1988 Numerical transient analysis of Markov models. *Comput. Oper. Res.* 15: 19-36
- Ross S M 1985 *Introduction to probability models* 3rd edn (Orlando, FL: Academic Press)
- Suri R 1985 An overview of evaluative models for flexible manufacturing systems. *Ann. Oper. Res.* 3: 13-21
- Trivedi K S 1982 *Probability and statistics with reliability, queueing and computer science applications* (Englewood Cliffs, NJ: Prentice-Hall)
- Viswanadham N, Narahari Y 1988 Stochastic Petri net models for performance evaluation of automated manufacturing systems. *Inf. Decision Technol.* 14: 125-142
- Wonham W M 1987 A control theory for discrete event systems, Systems and Control Group Report 8714, Department of Electrical Engineering, University of Toronto, Toronto