Approximate Performance Models Of Multimedia Communications Over Fast Packet-Switched Networks

Stanley Shiouming Wang

CENG Technical Report 94-19

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, California 90089-2562
(213)740-4579

August 1994

APPROXIMATE PERFORMANCE MODELS OF MULTIMEDIA COMMUNICATIONS OVER FAST PACKET-SWITCHED NETWORKS

by

Stanley Shiouming Wang

A Dissertation Presented to the

FACULTY OF THE GRADUATE SCHOOL

UNIVERSITY OF SOUTHERN CALIFORNIA

In Partial Fulfillment of the

Requirements for the Degree

DOCTOR OF PHILOSOPHY

(Computer Engineering)

August 1994

To my parents and my wife,

for their encouragement, faith and support.

Acknowledgement

A number of people have helped in making this possible. I would like to express my sincere appreciation to Professor John A. Silvester, my thesis advisor, for his support and guidance throughout the course of my study at USC. I would also like to thank my committee members, Professor Victor O. K. Li and, in particular, Professor Kenneth S. Alexander, for their valuable suggestions and comments. My special thanks to Professor Monte Ung for always being like a friend to me. Thanks from the bottom of my heart to my colleagues at USC, Dr. Syu-Je Wang, Dr. Authur Y. M. Lin, Dr. John S. Yang, Dr. Frank Y. S. Lin, Dr. Ram Krishnan, Nelson L. S. Fonseca, Te-Kai Liu, Gilberto Mayor, Ebrahim Ismail, Chang-Ann Fun, and Tien-Chien Yu for providing constant encouragement and discussions. Thanks also due to the administrative support and friendship from William Bates, Mary Zittercob, Lucille Stivers, Lita Arcana and Regina Morton which have made my stay at USC a much more pleasant one.

Last but not least, my deepest gratitude to my parents, my wife, my sister, my brother, and the rest of my family for always standing by me through the long, often tedious process of getting a Ph.D.

Table of Contents

List of Figures

Abstract

Future fast packet-switched networks are expected to provide various services including video, voice and data. Since different traffic sources have different statistical characteristics and quality-of-service requirements, they need to be modeled accurately. Existing performance models for fast packet-switched networks supporting multimedia applications, however, are either too complicated for real-time traffic control or do not consider multiple traffic classes. In this dissertation, we develope approximate performance models that are well-suited for traffic control of this type of networks.

We reduce the state space of a complex mixture of traffic sources to a two-state representation. We study the arrival process, the queue length distribution, and the delay and loss performance of a multiplexer using a two-state BMMPP/G/1 queueing system. We also study the asymptotic behavior of the queue length tail probabilities using the Z-transform of the queue length distribution. We then extend our model to analyze the system in the discrete-time domain with less computational complexity and a similar accuracy.

We find that these approximate models provide very good delay and loss performance prediction while remaining simple and computationally fast. Since the complexity of the approximation is independent of the number of traffic types and the number of sources in the system, the simplicity of the proposed models becomes much more significant as we consider larger (with more traffic classes or sources) systems.

Chapter 1

Introduction

Since the *Integrated Services Digital Network* (ISDN) standard was adopted by the *International Telegraph and Telephone Consultative Committee* (CCITT) in 1984, the revolution in high-speed communications technologies has provided a tremendous impetus for the introduction of new applications for networks, such as multimedia applications. These increasing new demands for high-quality, high-bandwidth communications have triggered the development of new network services such as *Broadband Integrated Services Digital Network*s (B-ISDN) [26] and *Switched Multi-megabit Data Service* (SMDS) [4]. In this chapter, we reveal some of the challenges faced by these new services and outline our solutions to these challenging problems.

1.1 Statistical multiplexing

Asynchronous Transfer Mode (ATM) is expected to be the underlying transport mechanism for B-ISDN [54], [70]. In ATM networks, the available transmission capacity is shared by statistically multiplexing fixed-length packets (called *cells* in ATM terminology) generated from all traffic sources. The statistical multiplexing gains, which result from offsetting short-term peaks of some traffic sources with below average bandwidth requirements from other sources, become more significant as the traffic sources become burstier. In fact, ATM networks are capable of supporting traffic sources whose aggregated peak data rate is well above the available system bandwidth provided that the overall average bandwidth requirement does not exceed the system capacity. Furthermore, the effectiveness and economics of ATM as the basic network infrastructure largely depends on the ability to derive network efficiency via statistical multiplexing gains.

Due to the statistical nature of the offered traffic, however, load may temporarily exceed the output capacity of the multiplexer resulting in a *temporary overload.* This can be accommodated by the provision of buffers at the expense of increased delay. The buffers will absorb excess traffic load until the instantaneous total bandwidth requirement drops below the output capacity again. Increasing buffer size alone, however, does not eliminate all potential problems. Severe congestion or even cell losses (due to buffer overflow or excessive delay) can result from excessively long overload duration. Thus, traffic offered to a statistical multiplexer should be well engineered in order balance statistical multiplexing gains and desired performance.

1.2 Traffic characteristics and service requirements

Since one of the goals of B-ISDN is to support diverse services, the underlying transport network is expected to be operated in a heterogeneous environment and carry different types of traffic such as data, voice, image, and video, which present substantially different traffic characteristics and *Quality Of Service* (QOS) requirements. Examples of different traffic characteristics include: a constant data rate for *Continuous Bit Rate* (CBR) traffic; and high burstiness for *Variable Bit Rate* (VBR) traffic. Examples of different QOS requirements include: loss sensitivity for data traffic: and a strict delay (jitter) bound for playback type of voice or video applications.

Many key services in B-ISDN such as compressed video and file transfer are bursty in nature. Several measures of the burstiness of the traffic have been defined. Two simple measures are: the ratio of peak to mean arrival rate; and the coefficient of variation, defined to be the ratio of standard deviation to mean of the interarrival time. Recently, a more elaborate measure using the *Hurst parameter* [32] has been considered making use of the concept of self-similarity [44]. A queueing system seeing a bursty arrival process usually has much worse performance (such as average system time and loss probability) as compared to a queueing system with a non-bursty arrival process. Consequently, models that capture the correlated nature of the traffic instead of widely used renewal processes are essential for predicting the performance of the emerging B-ISDN.

Besides the different traffic characteristics and QOS requirements, different services also require a wide range of bandwidths. Fig. 1.1 shows typical bandwidth requirements for different types of B-ISDN network services which range from a few Kbps to hundreds of Mbps.

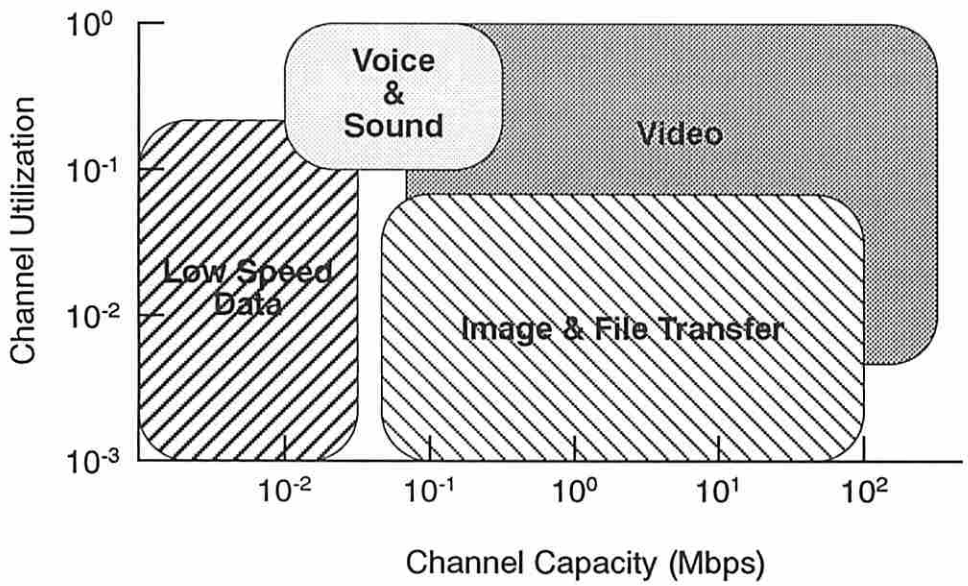

Fig. 1.1 B-ISDN traffic characteristics (source [20]).

1.3 Traffic control

In traditional circuit-switched networks, user demand and performance is generally described in terms of call arrival rates, call holding times, and call blocking probability etc. Hence traffic control has been primarily focussed on the call-level. In packet-switched networks, on the other hand, packet-level traffic control has been the center of attention due to the vast interest in performance measures such as packet throughput, packet delay, and packet loss probabilities. In ATM networks, however, cells from different virtual-paths, each of which is a bundle of virtual circuits, are statistically multiplexed over high-speed channels. Thus, traffic control in these networks must deal with two different time scales:† *call level* and *cell level,* and three different scopes: *entry points*, *intermediate nodes* (switches) and *the entire network.* The major function at the call level is to efficiently assign bandwidth to each connection in order to maximize the system utilization and at the same time reduce the probability of congestion (or recover from congestion when it occurs). The cell level traffic control deals with the actual handling of the cells at the ATM transport layer, e.g., buffering strategies, selective discarding, and traffic shaping. In what follows, "traffic control" refers to call level control at the entry points unless otherwise mentioned.

For B-ISDN, due to the heterogeneous (or multimedia) environment, the design of traffic control becomes one of the fundamental engineering challenges for efficient network resource utilization. There is a tradeoff between efficient resource utilization and the complexity of the traffic control algorithm for such networks. The challenge is to achieve high resource efficiency while maintaining simple control structures especially for large networks. Furthermore, since different traffic classes have diversified QOS requirements, the word "congestion," which (in single traffic class networks) is usually defined independent of the service, is no longer absolute. In other words, the network may appear congested for some services while its performance is still acceptable for others. Thus, congestion control for B-ISDN should be service dependent and may have different objectives for different traffic classes.

Two types of congestion control schemes, *reactive control* and *preventive control*, are possible. In reactive control, congestion is alleviated after it is detected. Bae and Suda [1] and Woodruff and Kositpaiboon [82] point out some problems in using reactive control for high-speed networks. Simply speaking, as the propagation delay becomes the dominating factor (over queueing and processing delays) the inherent latency of reactive control schemes results in a large number of cells dropping out of the pipe before any control reaction can be activated. In contrast to reactive control, preventive control tries to prevent congestion from occurring by limiting the load in the system. This implies use of a two-phase control scheme: *admission control* accepts a new connection only if its traffic, after adding to existing traffic, does not lead to congestion; and *traffic enforcement* (often referred to as *traffic policing*, *usage control* or *bandwidth management*) makes sure that the declared resource requirements are not exceeded.

Recently, admission control for ATM networks have received a lot of attention. The basic principle of admission control is to make a decision to either accept or deny a new connection

† Some researchers suggest the use of an extra burst level to deal with potential problems caused by highly bursty sources [31], [64].

request based on some information of current load. This information could be either an actual measurement of recent arrivals or simply a deduction from the declared traffic descriptors of existing connections. A conservative admission policy will allow relatively little traffic into the network and thus reduce the possibility of congestion and cell loss (in terms of buffers overflows). This conservative approach, however, would result in a lower system utilization and higher call blocking probability than a more aggressive one (i.e., one which accepts more calls). The challenge is to find the maximum number of connections that can be admitted to the system while maintaining an acceptable level of probability of cell loss. Discussion of the basic principles and different approaches for managing traffic in ATM networks can be found in [1], [25], [29], [30], [64], and [82].

Various admission control algorithms for ATM networks have been considered. For example, Murase *et al.* proposed an admission control procedure based on, so called, "individual virtual cell loss probability," which is a function of the incoming call's peak and mean rates [56], [57]. Ferrandiz and Lazar [18] ignore packet loss due to buffer overflow and use a *Markov modulated point process* to decide whether to admit new calls or not. More recently, Shim *at el.* [76] use the cell loss probability calculated from their *three-state Markov model* as the criterion for admission control. In order to achieve real-time control, current proposals either make use of a table driven call acceptance policy (which is restrictive in the number and mixes of calls that can be considered); or a fast algorithm to compute expected performance (which is typically a drastic over-simplification of the traffic characteristics). Development of an analytical model that retains detailed characteristics of the various types of traffic yet is fast to compute is believed to be the first step towards a successful implementation of any traffic control algorithm for the future networks. This is a key objective of this dissertation.

In order to better understand the complexity in the modeling, we now review some of the key traffic models found in the literature.

1.4 Traffic models for voice sources

Numerous studies that characterize packetized voice sources, e.g., [3], [8], [9], [27], [33], [46], [55], [77], [79], and [84], have been made among which Brady's *ON-OFF process* [9] has been widely adopted as the model for the arrival process corresponding to a single voice source. In the ON-OFF process model, the voice source alternates between exponentially distributed (or geometrically distributed if we use a discrete time scale) ON periods and exponentially distributed OFF periods. The ON periods correspond to talkspurts while the OFF periods represent silence durations. Packets are created with a constant interarrival time during the ON periods and no packets are generated during the OFF periods. The transition rates (or *transition probabilities* for discrete time case), from ON to OFF (α) and from OFF to ON (β), are determined by the expected length of the talkspurts and the expected length of the silence durations (see Fig. 1.2).

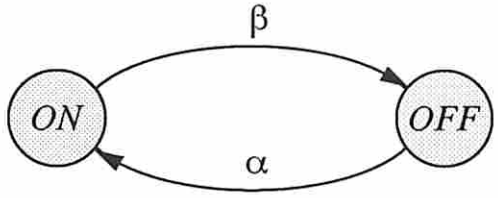

Fig. 1.2 An ON-OFF process.

A superposition of N such ON-OFF processes forms a *finite-state birth-death process* with the states representing the number of voice sources in talkspurt (in the ON-state) as shown in Fig. 1.3.

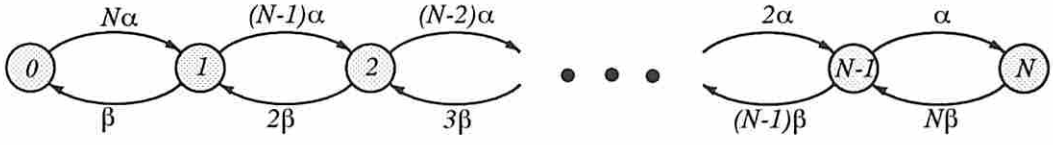

Fig. 1.3 The birth-death process for the number of voice sources in talkspurt.

Yuan and Silvester [84] have successfully used the above model in their analysis to study a packet switched system by solving the system explicitly. Other models also include: an *Interrupted Poisson Process* (IPP)[†] for an individual voice source and an (N+1)-state *Markov Modulated Poisson Process* (MMPP) for the aggregation of N IPP's suggested in [33]; and a renewal (deterministic) process for each voice source and $\Sigma_i D_i$ process for the superposition analyzed in [66].

1.5 Traffic models for video sources

Applications of digital video type, such as videotelephony, videoconferencing, and broadcast video, impose huge bandwidth requirements and are considered likely to be one of the major consumers of network bandwidth in the future. The amount of information in a coded video signal depends on the activity in the captured scene as well as the compression algorithm[‡] used. After compression, the resulting bit-rate may vary on a frame by frame basis, which is referred to as *Variable-Bit-Rate* (VBR) coding. Two types of correlations can be observed in the cell arrival stream generated by digital video applications: *short-term correlation* and *long-term correlation.*

[†] An IPP is identical to an ON-OFF process except that packet arrivals form a Poisson process (or a Bernoulli trial for discrete case) during the ON period instead of arriving constantly.

[‡] Presently, there are three major compression standards, namely JPEG [81], pX64 [49], and MPEG [43].

UNIVERSITY OF SOUTHERN CALIFORNIA
THE GRADUATE SCHOOL
UNIVERSITY PARK
LOS ANGELES, CALIFORNIA 90007

This dissertation, written by

Stanley Shiouming Wang

under the direction of his..... Dissertation Committee, and approved by all its members, has been presented to and accepted by The Graduate School, in partial fulfillment of requirements for the degree of

DOCTOR OF PHILOSOPHY

.....
Dean of Graduate Studies

Date July 25, 1994

DISSERTATION COMMITTEE

.....
Chairperson

.....

.....

Short-term correlation corresponds to uniform activity levels and lasts for only a few hundred milliseconds. Long-term correlation corresponds to scene changes and may last for several seconds [75], [80].

There are two possible transmission modes, *burst mode* (Fig. 1.4) and *uniform mode* (Fig. 1.5), that can be used to transmit the cells generated in each frame (typically 30 frames per second). In burst mode, (assuming that the codec is fast enough) video signals from the current frame are compressed while the cells from the previous frame are being sent. The new cells are transmitted at the beginning of the next frame at peak rate. In uniform mode, the cells generated in a frame are stored in a pre-buffer (see Fig. 1.6), pre-smoothed, and then transmitted at a uniform rate spread over the duration of the next frame.

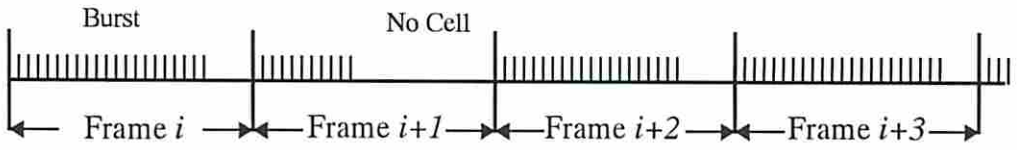

Fig. 1.4 Cell arrival process for burst-mode transmission.

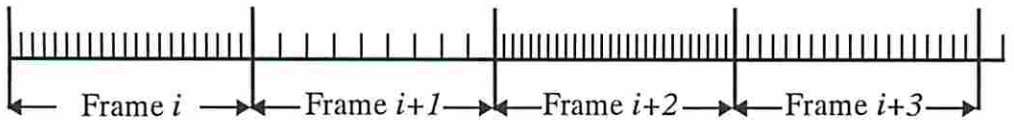

Fig. 1.5 Cell arrival process for uniform-mode transmission.

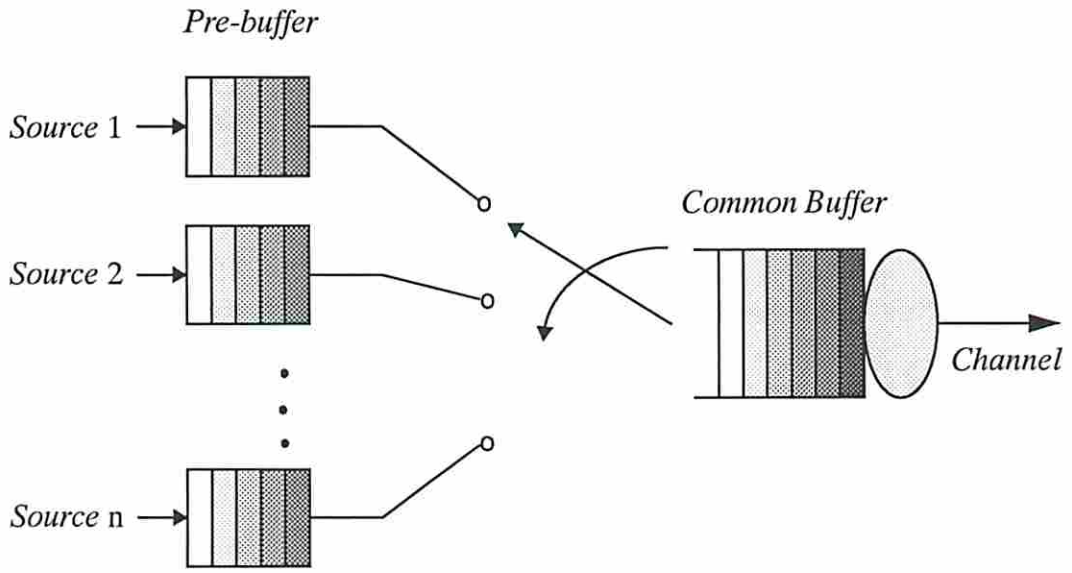

Fig. 1.6 A statistical multiplexer using pre-buffers.

Several models for video sources have been proposed in the literature. In [42], Lazer *et al.* assume bursty-mode transmission and model a video source as a *two-state random process*. In their model, state one represents a burst while state two corresponds to an off period. Burst lengths are assumed to be independent variables uniformly distributed between pre-specified upper and lower bounds, and the constant cycle time (one visit to both states) is set to be the frame duration.

Maglaris *et al.* [53] use uniform-mode transmission and suggest a *continuous-time Markov chain* to model video sources without sudden scene changes, e.g., videotelephony showing a person talking in front of the camera. In this model, the output data rates from a codec are assumed to be a set of discrete levels evenly spaced between zero and the peak rate. Transitions are assumed to occur only between adjacent levels according to a birth-death Markov chain. The transition rates are determined by matching the model's parameters with statistical measurements of real traces. Fig. 1.7 shows the state transition diagram of Maglaris' model where A is the quantization step and $M+1$ is the number of quantization levels. Sen *et al.* [75] extend this model to include scene changes by means of a *two-dimensional Markov chain*. As shown in Fig. 1.8, the possible data-rate levels are the linear combinations of two basic levels: a high rate A_h, which represents scene changes (activity-level changes), and a low rate A_l, which represents a uniform activity level. Thus, a total of $(M_h+1)\times(M_h+1)$ distinct data-rate levels are possible. As before, the system parameters, a, b, c, d, A_h, M_h, A_l, and M_l are matched by real measurements.

Lam *et al.* [40] propose a new model for cell streams generated by a coding scheme which considers both intra and inter-frame compression techniques such as MPEG. In their model, arrival rate fluctuations are smoothed out by providing a pre-buffer similar to the one used in uniform-mode transmission. Unlike the uniform-mode transmission, however, the smoothing is not done on a per-frame basis. Instead, they observe a repeating pattern (which typically last about 9 or 12 frames for MPEG) on the arrival rate generated by a codec and smooth the arrival rate over the duration of the repeating pattern. They also introduce an algorithm to reduce the excessive buffer delay due to this smoothing technique. The resulting model appears to be similar to Sen's model except that in Sen's model no inter-frame compression is assumed.

Other models can also be found in the literature, for example: the *autoregressive process* discussed in [61] and the *autoregressive moving average process* presented in [23]. However, the analysis of a queueing model using these processes as the arrival process may not be tractable. Hence, these models are useful only for simulation models and have little value in analytical methods.

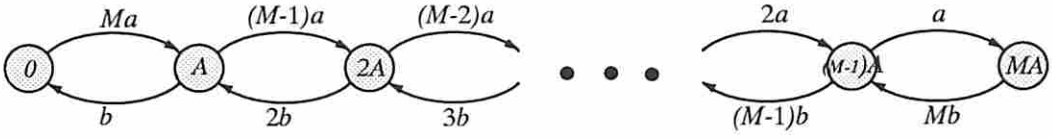

Fig. 1.7 State transition diagram for Maglaris' model.

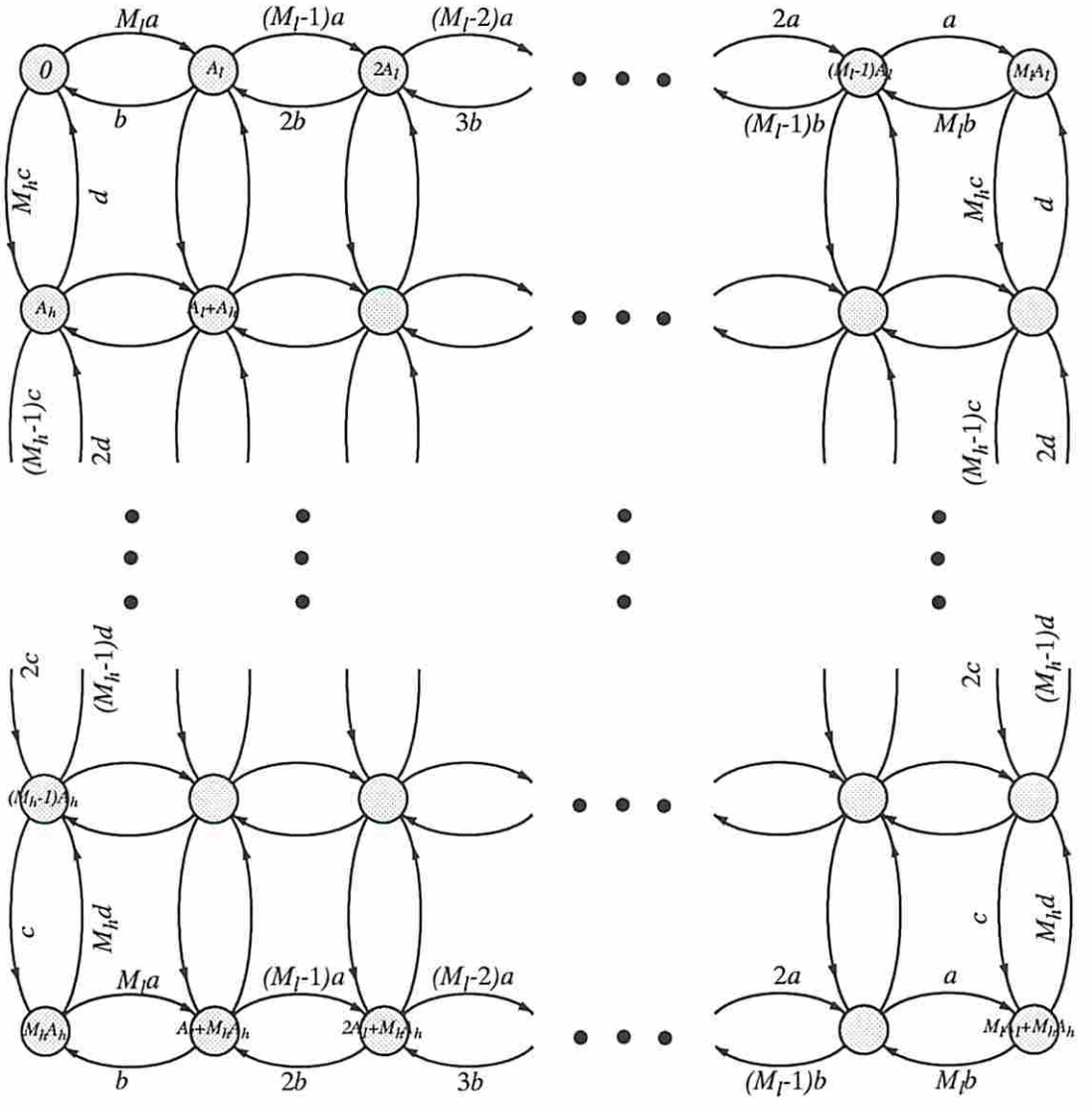

Fig. 1.8 State transition diagram for Sen's model.

1.6 Traffic models for data sources

The dominate traffic source for computer communication networks so far is believed to be still data traffic, which have been traditionally modeled by a *renewal process*. Among the commonly adopted renewal processes, *Poisson process* (or *Bernoulli process* for discrete-time systems) is by far the most widely used model due to its relative mathematical simplicity. Poisson processes enjoy some nice analytical characteristics such as the memoryless and independent arrival properties. Also, the superposition of independent Poisson processes and (probabilistic) decomposition of a Poisson process result in new Poisson processes whose parameters can be easily computed. All of these, in fact, greatly simplify the queueing (network) problems assuming Poisson arrival processes.

It may be reasonable to model each individual data source as a Poisson process. It is also commonly argued that with a constant aggregate arrival rate and an increasing number of independent sources, the overall arrival stream is expected to approach to a Poisson process. However, some recent research results (see [44], for example) have pointed out that the superposition of a large number of data sources may not always result in a Poisson process. In what is referred to as self-similar traffic in [44], it was shown that the aggregate traffic from LAN's does not form a Poisson process. In other words, by increasing the observation time scale, the aggregate traffic does not "smooth" out as it would for Poisson processes.

1.7 Traffic aggregation and approximate models

As was pointed out, some of the source models discussed here are too complex except for driving a simulation. Others such as Maglaris' and Sen's models for video traffic may be analytically tractable but only for a small system (with a small number of sources). The composite model for more than one traffic class is, however, typically too complicated to be used in an analytical model while still maintaining a reasonable computational complexity. For example, combining Maglaris' model (for video traffic) with the model given by Fig. 1.3 (for voice traffic) would result in a two dimensional Markov chain where the state space is proportional to the product of the numbers of video sources and voice sources; while using Sen's model would result in a similar three dimensional Markov chain.

Failure to generate useful analytical results using one of the voice traffic models discussed above as the arrival process for systems serving voice sources has forced researchers to look for approximations. Two major approaches have been proposed before. The first approach uses a fluid flow approximation [12], [58], [79], which somewhat overlooks the randomness (or, in other words, the short-term variation) of the arrivals. The second approach carefully matches the process to a simpler one, e.g., the two-state MMPP used in [3], [27], and [58], and the *renewal process* used in [58]. Heffes and Lucantoni, in their one of the earliest work using the second approach [27], use the following parameter matching procedure for the two-state MMPP:

i. the overall expected arrival rate;

ii. the variance-to-mean ratio of the number of arrivals in some time interval;

iii. the long-term variance-to-mean ratio of the number of arrivals;

iv. the third moment of the number if arrivals in some time interval.

Upon observing the strong role of the overload† period in determining the performance of the system, Nagarajan *et al.*, in their matching procedure [58], replace the last step above by "*the variance of the number of arrivals in some time interval giving that the system is in overload state*," and display an improvement on predicting packet loss in a finite-buffered system. In [3], Baiocchi *et al.* attack the problem from a different angle. They use a theorem proved in [59] (theorem 2.3.1, p. 62) and present an "asymptotic matching procedure" which leads to more accurate results than those of [27]. In their procedure, they compute a parameter which controls the duration of an overload period and match the rest of the parameter set of a two-state MMPP by first-order statistic of arrivals. All these models work reasonably well for a low to moderate traffic load with a small number of voice sources. As the traffic load increases, however, the performance prediction obtained from these approximations starts to diverge from the actual performance. Furthermore, as the number of voice sources in the system increases, the computational complexity is still too high to be implemented as a real-time traffic control algorithm.

There are only a few studies dealing with the performance of systems with integrated traffic (video, voice and data). Part of the reason is that video or voice traffic alone already makes the model so complex that any effort to integrate these two together usually makes the analysis intractable. In [83], Ye and Li use a *Quasi-Birth-Death* (QDB) process to analyze the queueing behavior of a multiplexer with multimedia input traffic. In [45], Li proposes a general solution technique using generating functions for the queueing analysis of multimedia multiserver systems. As in the approximations for voice systems, these models fail to resolve one of the most crucial problems: the high computational complexity.

1.8 Contributions of the research

Existing performance models for fast packet-switched networks supporting multimedia applications are either too complicated for real-time traffic control or do not consider multiple traffic classes. Our objective is to develop approximate performance models that are well-suited for real-time traffic control. Thus, simplicity, while retaining accuracy, is the main focus of the proposed models. The simplicity of the proposed approximation includes both the simplicity of the representation of complex traffic sources and the computational solution technique itself. For describing the approximate model, a set of simple traffic descriptors is preferred; while the solution to a desired performance measure should be computable in real-time.

We reduce the state space of the combined traffic model to represent the traffic as a two-state BMMPP and analyze an ATM multiplexer using a BMMPP/G/1 queueing system. We study the arrival process, the queue length distribution, and the loss probability of the system using this approximate model. We also study the asymptotic behavior of the queue length tail distribution

† Defined to be when the instantaneous bandwidth requirement exceeds the system capacity.

using the Z-transform of the queue length distribution. We then move on to analyze the system in discrete-time domain with less computational complexity and a similar accuracy.

We find that these approximate models provide very good delay and loss performance prediction while remaining simply and computationally fast. Since the complexity of the approximate models is independent of the number of traffic types as well as sources in the system, the contributions of this research become much more significant as we consider more complicated (with more traffic classes as well as sources) systems.

1.9 Outline of the dissertation

The rest of the dissertation is organized as follows. Chapter 2 defines an MMPP followed by the analysis of a single server queue where the arrival process is assumed to be an MMPP with batch arrivals (denoted BMMPP). We include the analysis of the arrival process of a BMMPP, the queue length distribution for BMMPP/G/1 queues, and the average delay and loss probability for BMMPP/G/1/K queues. For the queue length distribution, we emphasize determination of the tail probabilities for which we present an easy way to find asymptotic behavior using the Z-transform of the queue length distribution. We also include algorithms to calculate the average delay and loss probability for BMMPP/G/1/K queues which are useful for real-time traffic control. These results are used in later chapters to predict the performance of a multimedia communication system.

In Chapter 3 we use a two-state BMMPP to specifically model the arrival process of an ATM multiplexer loaded with voice and data sources. We also propose a new way to match the parameters which leads to a more accurate delay and loss performance prediction and less computational complexity then previous work.

We extend this approach in Chapter 4 to include video traffic without significant increase in the computational complexity of the model. In fact, the computational complexity of the proposed approximation is independent of the number of traffic sources. Simulation is used to verify the accuracy of this extended model.

Since ATM is really a discrete time system, in Chapter 5 we modify our model to analyze the performance of a multimedia communication system in the discrete-time domain. ATM multiplexer performance based on *Discrete-time Batch Markovian Arrival Process* (D-BMAP) models is presented. As we found in the continuous-time domain, the performance of an ATM multiplexer loaded with different types of traffic can be approximated by reducing the arrival processes to a two-state D-BMAP. We provide extensive validation of these results and also present a couple of examples on how the proposed work can be used for real-time admission control.

Finally, in Chapter 6 we discuss and summarize the results presented in this dissertation and identify some directions for future research.

Chapter 2

Batch Arrival Markov Modulated Poisson Process

In this chapter, we first define an MMPP. Some related analytical results are cited followed by a detail analysis of BMMPP/G/1 queues. Although part of the analysis presented in this chapter is a special case of the analysis for N/G/1 queues presented in [60] and [67], the results, which are specifically derived for BMMPP/G/1 queues, are much simpler and more comprehensible. We include the analysis of the arrival process of a BMMPP, the queue length distribution for BMMPP/G/1 queues, and the average delay and loss probability for BMMPP/D/1/K queues. For the queue length distribution, our emphasis is on the tail probabilities for which we present an easy way to find the asymptotic behavior using the Z-transform of the queue length distribution. We also include algorithms for calculating the average delay and loss probability for a BMMPP/G/1/K queueing system.

2.1 Definitions and related results

Markov Modulated Poisson Process, a special case of the *Neuts' versatile Markovian process* [60] (called the *N-process*, which is equivalent to the *Batch Markovian Arrival Process*, BMAP, defined by Lucantoni in [50]), is a *doubly stochastic Poisson process* where the arrival rate is determined by the state (or *phase*) of a finite-state Markov chain. (If we explicitly model batch arrivals we have BMMPP.) For example, a two-state MMPP can be defined by four parameters: the transition rates from phase one to phase two and from phase two to phase one, and the arrival rates at phase one and phase two.

In performance modeling, MMPP is of interest due to its analytical tractability and versatility in modeling correlated arrival processes. Some results related to Markov modulated queueing systems are available in the literature. Knessl and Mathkowsky [37] study the stationary distribution of the unfinished work for a two-state MMPP/G/1 queue. Prabhu and Zhu [63] complete a detailed analysis, including the waiting time, the idle time and the busy period, of an MMPP/G/1 queue. Recently, Zhu, in his work [86], includes some results on the arrival and departure processes for an MMPP/M/1 queue with bulk arrivals. Neuts [60] defines and analyzes the N-process which is used in [67] by Ramaswami to study N/G/1 queues, a more general case of MMPP/G/1 queues, including the stationary queue length at departures and the waiting time distribution. Ramaswami's results are later extended by Blondia in [6] to a finite-buffered system.

In the next two sections, we specialize Neuts' and Ramaswami's analysis for N/G/1 queues to that of a single server queueing system in which the arrival process is a BMMPP and the service times are identically distributed random variables with an arbitrary probability distribution, i.e, a BMMPP/G/1 queueing system.

In order to describe the system, the following notation is defined:

- m the number of phases of the underlying Markov chain;
- π_i the steady state probability of the system being in phase i;
- λ_i the arrival rate for the system in phase i;
- $g_i(k)$ $Prob\{$bulk size is $k \mid$ the system is in phase $i\}$;
- $N(t)$ the number of arrivals by time t with $N(0) = 0$ (the arrival process);
- $\Phi(t)$ the phase of the arrival process at time t (the phase process);
- $p_{ij}(n, t)$ $Prob\{N(t) = n, \Phi(t) = j \mid \Phi(0) = i\}$;
- r_{ij} transition rate from phase i to phase j with $\sum_{j=1}^{m} r_{ij} = 0$, $1 \le i \le m$, (assuming stationary).

We also introduce the following probability distributions:

- $\varepsilon_{ij}^{k}(h)$ given $\Phi(0) = i$ and the system is empty at time $t = 0$, the conditional probability that the first arrived bulk, which occurs at some time $t \le h$, has a size of k and $\Phi(t^+) = j$;
- $\alpha_{ij}^{n}(h)$ given $\Phi(0) = i$, the conditional probability that n customers arrive during a service time $t \le h$ and $\Phi(t^+) = j$;
- $\beta_{ij}^{n}(h)$ given $\Phi(0) = i$ and the system is empty at time $t = 0$, the conditional probability that there are n customers left behind by the first departure, which occurs at some time $t \le h$, and $\Phi(t^+) = j$;
- $S(h)$ the service time probability distribution, i.e., the probability that the service time $t \le h$.

It is then not difficult to see that:

$$\varepsilon_{ij}^{k}(h) = \left[\int_0^h p_{ij}(0, \tau)d\tau\right][\lambda_i g_i(k)], \quad k \ge 1, h \ge 0 \tag{2.1}$$

$$\alpha_{ij}^{n}(h) = \int_0^h p_{ij}(n, \tau)dS(\tau), \quad n \ge 0, h \ge 0 \tag{2.2}$$

$$\beta_{ij}^{n}(h) = \sum_{k=1}^{n+1} (\varepsilon_{ij}^{k} \otimes \alpha_{ij}^{n-k-1})(h), \quad n \geq 0, h \geq 0 \tag{2.3}$$

where $\otimes$ is a convolution operator.

Throughout this chapter, $\hat{\mathcal{F}}(s)$ represents the *Laplace-Stieltjes Transform* (LST) of a continuous probability distribution function $f(t)$; and $\tilde{\mathcal{A}}(z)$ indicates the *Z-transform* of a discrete probability distribution function a_n. Boldfaced uppercase characters are reserved to denote matrices of the corresponding elements. For instance, $\boldsymbol{R}$ is the matrix having r_{ij} as the element in the ith row and the jth column. Vectors will be represented as $\vec{\pi} = (\pi_1, \pi_2, \ldots, \pi_m)$. For the vector $\vec{g} = (g_1, g_2, \ldots, g_n)$, Δg refers to the diagonal matrix having g_i as the ith element on the main diagonal of the matrix, i.e., $\Delta g = diag(g_1, g_2, \ldots, g_n)$.

Remark. For the system described above, the system utilization is given by:

$$\rho \equiv (\text{Average Arrival Rate}) \cdot (\text{Average Service Time})$$

$$= \left[\sum_{i=1}^{m} \lambda_i \left(\sum_{k=1}^{\infty} k g_i(k)\right) \pi_i\right] \left[\int_0^{\infty} t\, dS(t)\right]$$

$$= [\vec{\pi} \cdot \Delta\lambda \cdot \Delta\tilde{g}'(1) \cdot \vec{e}]\, [-\hat{S}'(0)], \quad \text{where } \vec{e} = (1, 1, \ldots, 1)' \; \square$$

2.2 The BMMPP arrival stream

In this section we derive the Z-transform for the number of arrivals during a time period, say $[0, t]$, from a BMMPP. We first look at the differential-difference equations for the (Markov) random process $\{N(t), \Phi(t); t \geq 0\}$. The differential-difference equations can be written as:

$$\frac{\partial}{\partial t} p_{ij}(n, t)$$

$$= \sum_{\substack{k=1 \\ k \neq j}}^{m} p_{ik}(n, t) \cdot r_{kj} - \sum_{\substack{k=1 \\ k \neq j}}^{m} p_{ij}(n, t) \cdot r_{jk} + \sum_{k=0}^{n} p_{ij}(k, t) \cdot g_j(n-k) \cdot \lambda_j - p_{ij}(n, t) \cdot \lambda_j$$

$$= \sum_{\substack{k=1 \\ k \neq j}}^{m} p_{ik}(n, t) \cdot r_{kj} + p_{ij}(n, t) \cdot r_{jj} + \sum_{k=0}^{n} p_{ij}(k, t) \cdot g_j(n-k) \cdot \lambda_j - p_{ij}(n, t) \cdot \lambda_j$$

Thus we have:

$$\frac{\partial}{\partial t}p_{ij}(n, t) = \sum_{k=1}^{m} p_{ik}(n, t) \cdot r_{kj} - p_{ij}(n, t) \cdot \lambda_j + \sum_{k=0}^{n} p_{ij}(k, t) \cdot g_j(n-k) \cdot \lambda_j \tag{2.4}$$

$$\frac{\partial}{\partial t}p_{ij}(0, t) = \sum_{k=1}^{m} p_{ik}(0, t) \cdot r_{kj} \tag{2.5}$$

for $1 \le i, j \le m$. (2.4) and (2.5) can be written in matrix form as:

$$\frac{\partial}{\partial t}\boldsymbol{P}(n, t) = \boldsymbol{P}(n, t) \cdot \boldsymbol{R} - \boldsymbol{P}(n, t) \cdot \Delta\lambda + \sum_{k=0}^{n} \boldsymbol{P}(k, t) \cdot \Delta g(n-k) \cdot \Delta\lambda \tag{2.6}$$

$$\frac{\partial}{\partial t}\boldsymbol{P}(0, t) = \boldsymbol{P}(0, t) \cdot \boldsymbol{R} \tag{2.7}$$

Applying Z-transform on both sides of (2.6), we obtain:

$$\frac{\partial}{\partial t}\tilde{\mathcal{P}}(z, t) = \tilde{\mathcal{P}}(z, t)\,[\boldsymbol{R} - \Delta\lambda + \Delta\lambda \cdot \Delta\tilde{g}(z)] \tag{2.8}$$

Remark.

$$\sum_{n=0}^{\infty} z^n \sum_{k=0}^{n} \boldsymbol{P}(k, t) \cdot \Delta g(n-k) \cdot \Delta\lambda$$

$$= \sum_{k=0}^{\infty} z^k \cdot \boldsymbol{P}(k, t) \cdot \Delta\lambda \left[\sum_{n=k}^{\infty} z^{n-k} \cdot \Delta g(n-k) \right]$$

$$= \tilde{\mathcal{P}}(z, t) \cdot \Delta\lambda \cdot \Delta\tilde{g}(z) \qquad \square$$

We then obtain the following closed-form expression† for the Z-transform of the number of arrivals in t:

---

† $\exp(At)$ is defined to be $I + At + A^2t^2/2! + A^3t^3/3! + \ldots$, where $\boldsymbol{A}$ is a square matrix and $\boldsymbol{I}$ is the identity matrix.

$$\tilde{\mathcal{P}}(z, t) = \exp\{t[\boldsymbol{R} - \Delta\lambda(\boldsymbol{I} - \Delta\tilde{g}(z))]\} \tag{2.9}$$

with $\tilde{\mathcal{P}}(z, 0) = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix. Note that it can be proved that (2.9) is a special case of equation (7) of [60] (page 768).

Remark. For a Poisson process, i.e., one in which $\boldsymbol{R} = 0$, $\Delta\lambda = \lambda$, and $\Delta\tilde{g}(z) = z$, (2.9) becomes $\tilde{\mathcal{P}}(z, t) = e^{\{-\lambda(1-z)t\}}$ which is a well known result. Also as derived in [27], for MMPP (i.e., no batch arrivals and $\Delta\tilde{g}(z) = z$), (2.9) reduces to $\tilde{\mathcal{P}}(z, t) = \exp\{t[\boldsymbol{R} - \Delta\lambda(1-z)]\}$. ❑

To find the mean number of arrivals, we need to evaluate the derivative of (2.9) at $z = 1$. Let $\boldsymbol{F}(z) = \boldsymbol{R} - \Delta\lambda(\boldsymbol{I} - \Delta\tilde{g}(z))$, we have the following:

$$\frac{\partial}{\partial z}\tilde{\mathcal{P}}(z, t)\Big|_{z=1} = \boldsymbol{0} + \boldsymbol{F}'(z)t + \frac{t^2}{2!}[\boldsymbol{F}(z)\boldsymbol{F}'(z) + \boldsymbol{F}'(z)\boldsymbol{F}(z)]$$

$$+ \frac{t^3}{3!}\{[\boldsymbol{F}(z)]^2\boldsymbol{F}'(z) + \boldsymbol{F}(z)\boldsymbol{F}'(z)\boldsymbol{F}(z) + \boldsymbol{F}'(z)[\boldsymbol{F}(z)]^2\}$$

$$+ \ldots\Big|_{z=1}$$

$$= \sum_{n=1}^{\infty}\frac{t^n}{n!}\sum_{i=0}^{n-1}[\boldsymbol{F}(z)]^i\boldsymbol{F}'(z)[\boldsymbol{F}(z)]^{n-1-i}\Big|_{z=1}$$

Since $\boldsymbol{F}'(z)\big|_{z=1} = \Delta\lambda \cdot \Delta\tilde{g}'(1)$ and $[\boldsymbol{F}(z)]^i\big|_{z=1} = \boldsymbol{R}^i$, we get:

$$\frac{\partial}{\partial z}\tilde{\mathcal{P}}(z, t)\Big|_{z=1} = \sum_{n=1}^{\infty}\frac{t^n}{n!}\sum_{i=0}^{n-1}\boldsymbol{R}^i \cdot \Delta\lambda \cdot \Delta\tilde{g}'(1) \cdot \boldsymbol{R}^{n-i-1} \tag{2.10}$$

Remark. If $\Delta\lambda \cdot \Delta\tilde{g}'(1)$ and $\boldsymbol{R}$ are *interchangeable* (*commutative*) then:

$$\frac{\partial}{\partial z}\tilde{\mathcal{P}}(z, t)\Big|_{z=1} = t\Delta\lambda \cdot \Delta\tilde{g}'(1)\sum_{n=1}^{\infty}\frac{\boldsymbol{R}^{n-1}t^{n-1}}{(n-1)!}$$

$$= t\Delta\lambda \cdot \Delta\tilde{g}'(1) \cdot e^{\boldsymbol{R}t}$$ ❑

Remark. Again, if we specialize (2.10) for a regular Poisson process, i.e., $\Delta\tilde{g}'(1) = 1$, we will have $\tilde{\mathcal{P}}'(z, t)\big|_{z=1} = \lambda t$. Meanwhile, for an MMPP, we will get:

$$\tilde{\mathcal{P}}'(z, t)\big|_{z=1} = \sum_{n=1}^{\infty} \frac{t^n}{n!} \sum_{i=0}^{n-1} \boldsymbol{R}^i \cdot \Delta\lambda \cdot \boldsymbol{R}^{n-i-1}$$

or,

$$\tilde{\mathcal{P}}'(z, t)\big|_{z=1} = t\Delta\lambda \cdot e^{\boldsymbol{R}t}$$

if $\Delta\lambda$ and $\boldsymbol{R}$ are interchangeable. ❑

2.3 Queue length distribution for BMMPP/G/1 queues

In order to analyze the queue length distribution, we assume that the system is work-conservative, i.e., the server is never idle as long as there are customers in the system. We also assume the system is stable, i.e., the system utilization (defined in section 2.1) $\rho < 1$, and is in steady state. We then define the following notation:

- C_n the nth customer to enter the system;
- δ_n the departure time of C_n;
- $\Phi(n)$ the phase of the arrival process right after the departure time instant of C_n, denoted δ_n^+;
- $x(n)$ the number of customers (including the server, if any) left behind by C_n, at δ_n^+.

One can easily see that $\{x(n), \Phi(n); n \geq 0\}$ forms a *semi-Markov process*, since $\{\Phi(n); n \geq 0\}$ is a Markov chain and $\{x(n); n \geq 0\}$ is a semi-Markov process. We then can use the *Matrix-Geometric Technique* [59] to analyze $\{x(n), \Phi(n); n \geq 0\}$.

Let $\boldsymbol{E}_n(h) = \left[\varepsilon_{ij}^n(h)\right]$, $\boldsymbol{A}_n(h) = \left[\alpha_{ij}^n(h)\right]$, and $\boldsymbol{B}_n(h) = \left[\beta_{ij}^n(h)\right]$. The transition probability matrix of the arrival process can be written as:

$$Q(h) = \begin{bmatrix} \boldsymbol{B}_0(h) & \boldsymbol{B}_1(h) & \boldsymbol{B}_2(h) & \boldsymbol{B}_3(h) & \cdot & \cdot & \cdot \\ \boldsymbol{A}_0(h) & \boldsymbol{A}_1(h) & \boldsymbol{A}_2(h) & \boldsymbol{A}_3(h) & \cdot & \cdot & \cdot \\ \mathbf{0} & \boldsymbol{A}_0(h) & \boldsymbol{A}_1(h) & \boldsymbol{A}_2(h) & \cdot & \cdot & \cdot \\ \mathbf{0} & \mathbf{0} & \boldsymbol{A}_0(h) & \boldsymbol{A}_1(h) & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}, \quad h \geq 0 \tag{2.11}$$

and,

$$Q \equiv \lim_{h \to \infty} Q(h) = \begin{bmatrix} \boldsymbol{B}_0 & \boldsymbol{B}_1 & \boldsymbol{B}_2 & \boldsymbol{B}_3 & \cdot & \cdot & \cdot \\ \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{A}_2 & \boldsymbol{A}_3 & \cdot & \cdot & \cdot \\ \mathbf{0} & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{A}_2 & \cdot & \cdot & \cdot \\ \mathbf{0} & \mathbf{0} & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \tag{2.12}$$

where $\boldsymbol{A}_n = \lim_{h \to \infty} \boldsymbol{A}_n(h)$; $\boldsymbol{B}_n = \lim_{h \to \infty} \boldsymbol{B}_n(h) = \sum_{k=1}^{n+1} \boldsymbol{E}_k \boldsymbol{A}_{n-k+1}$ and $\boldsymbol{E}_n = \lim_{h \to \infty} \boldsymbol{E}_n(h)$.

Let:

- $x_{ni} = \lim_{k \to \infty} prob\{x(k) = n, \Phi(k) = i\}, 1 \leq i \leq m, n = 0, 1, 2, \ldots$
- $\vec{x}_n = (x_{n1}, x_{n2}, \ldots, x_{nm}), n = 0, 1, 2, \ldots$, i.e., the steady state joint probability of the queue length and the phase of the BMMPP at departure instants.
- $x_n = x_{n1} + x_{n2} + \ldots + x_{nm}, n = 0, 1, 2, \ldots$, i.e., the steady state queue length probability of the BMMPP at departure instants.
- $\vec{X} = (\vec{x}_0, \vec{x}_1, \vec{x}_2, \vec{x}_3, \ldots)$

It is then not difficult to see that $\vec{X}$ is the invariant probability vector of the matrix Q defined in (2.12), i.e., $\vec{X}Q = \vec{X}$. Thus, we have:

$$\vec{x}_i = \vec{x}_0 \boldsymbol{B}_i + \sum_{k=1}^{i+1} \vec{x}_k \boldsymbol{A}_{i-k+1}, \quad i = 0, 1, 2, \ldots \tag{2.13}$$

By multiplying both sides of (2.13) and summing over all possible i, we have:

$$\sum_{i=0}^{\infty} z^i \vec{x}_i = \sum_{i=0}^{\infty} z^i \left(\vec{x}_0 \boldsymbol{B}_i + \sum_{k=1}^{i+1} \vec{x}_k \boldsymbol{A}_{i-k+1} \right)$$

$$\Rightarrow \quad \tilde{X}(z) = \vec{x}_0 \sum_{i=0}^{\infty} z^i \boldsymbol{B}_i + \sum_{i=0}^{\infty} z^i \sum_{k=1}^{i+1} \vec{x}_k \boldsymbol{A}_{i-k+1}$$

$$\Rightarrow \quad \tilde{X}(z) = \vec{x}_0 \sum_{i=0}^{\infty} z^i \sum_{k=0}^{i} \boldsymbol{E}_{k+1} \boldsymbol{A}_{i-k} + \sum_{i=0}^{\infty} z^i \sum_{k=0}^{i} \vec{x}_{k+1} \boldsymbol{A}_{i-k}$$

$$\Rightarrow \quad \tilde{X}(z) = \vec{x}_0 \sum_{k=0}^{\infty} z^k \boldsymbol{E}_{k+1} \sum_{i=k}^{\infty} z^{i-k} \boldsymbol{A}_{i-k} + \sum_{k=0}^{\infty} z^k \vec{x}_{k+1} \sum_{i=k}^{\infty} z^{i-k} \boldsymbol{A}_{i-k}$$

$$\Rightarrow \quad z\tilde{X}(z) = \vec{x}_0 \left(\sum_{k=0}^{\infty} z^{k+1} \boldsymbol{E}_{k+1} \right) \tilde{\mathcal{A}}(z) + \left(\sum_{k=0}^{\infty} z^{k+1} \vec{x}_{k+1} \right) \tilde{\mathcal{A}}(z)$$

$$\Rightarrow \quad z\tilde{X}(z) = \vec{x}_0 [\tilde{E}(z) - \boldsymbol{E}_0] \tilde{\mathcal{A}}(z) + [\tilde{X}(z) - \vec{x}_0] \tilde{\mathcal{A}}(z)$$

Thus, we have the following closed-form expression for the Z-transform of $\vec{x}_i$ (a more general expression for BMAP/G/1 queues is available from equation (3.2.6) of [67]):

$$\tilde{X}(z) [z\boldsymbol{I} - \tilde{\mathcal{A}}(z)] = \vec{x}_0 [\tilde{E}(z) - \boldsymbol{I}] \tilde{\mathcal{A}}(z) \tag{2.14}$$

In [50], Lucantoni suggested an improvement of the computational algorithm originally presented in [67] by Ramaswami to calculate the $\boldsymbol{A}_n$'s and $\boldsymbol{B}_n$'s. Using this algorithm, the simultaneous equations defined in (2.13) for the stationary queue length distribution are resolved by truncating the number of equations to a sufficiently large value (where the truncation error is small enough to be ignored) and then by applying any numerical technique for solving systems of linear equations. Note that according to our experience for an error tolerance of 10^{-20} (which is assumed to be the case in all numerical examples presented in the later chapters where solving a BMMPP/D/1/∞ system is required), around 190 $\boldsymbol{A}_n$'s and $\boldsymbol{B}_n$'s have to be calculated.

2.4 Loss probability for BMMPP/G/1/*K* queues

For a BMMPP/G/1 queue with a limited buffer size of, say, K, it can be readily seen from (2.12) that $\vec{X} = (\vec{x}_0, \vec{x}_1, \vec{x}_2, \ldots, \vec{x}_{K-1})$ is the invariant probability vector of the following stochastic matrix:

$$Q = \begin{bmatrix} B_0 & B_1 & B_2 & \dots & B_{K-2} & \sum_{i=K-1}^{\infty} B_i \\ A_0 & A_1 & A_2 & \dots & A_{K-2} & \sum_{i=K-1}^{\infty} A_i \\ 0 & A_0 & A_1 & \dots & A_{K-3} & \sum_{i=K-2}^{\infty} A_i \\ . & . & . & \dots & . & . \\ 0 & 0 & 0 & \dots & A_0 & \sum_{i=1}^{\infty} A_i \end{bmatrix} \tag{2.15}$$

Thus, we can solve $\vec{X}$ explicitly by using the facts that $\vec{X}Q = \vec{X}$ and $\sum_{n=0}^{K-1} \vec{x}_n \cdot \vec{e} = 1$. Specifically,

$$\vec{X} = \vec{b} \cdot (Q')^{-1} \tag{2.16}$$

where $\vec{b} = (0, 0, \dots, 0, 1)$ and $Q' = Q - I$ with all elements of its last column replaced by 1. Let:

- l_K be the loss probability for a system with a buffer size of K;
- N_a be the average number of customers that arrive between two consecutive departures;
- N_l^K be the average number of customers that are lost between two consecutive departures for a system with a buffer size of K.

Making use of the queue length distribution at departure instants obtained in (2.16), it is not difficult to show that:

$$N_a = x_0 \left(\sum_{n=1}^{\infty} nB_{n-1} \vec{e}\vec{\pi} \right) + (1 - x_0) \left(\sum_{n=1}^{\infty} nA_n \vec{e}\vec{\pi} \right) \tag{2.17}$$

$$N_l^K = x_0 \left(\sum_{n=1}^{\infty} (n-K)^+ B_{n-1} \vec{e}\vec{\pi} \right) + \left(\sum_{n=1}^{\infty} \sum_{k=1}^{K} x_k (n+k-K)^+ A_n \vec{e}\vec{\pi} \right) \tag{2.18}$$

where $(x)^+$ is defined to be $\max\{0, x\}$. Note that in (2.17) and (2.18), the first term counts the number of arrivals and number of losses for the situation where a customer departs leaving an

2.7 Summary of the chapter

MMPP/G/1 queues have been widely studied in the literature (see the citations in section 2.1) and widely adopted in performance modeling. The reason that MMPP has received far more attentions than the more general BMAP does is because the nice properties it shares with the Poisson processes. First of all, given that the MMPP is in a specific phase, the process degenerated to an ordinary Poisson process which, of course, enjoy some well-known properties such as memoryless property. Secondly, the superposition of MMPP's simply results in another MMPP with the state space to be the convolution of the state spaces of the components. Similar to a superposition, a (probabilistic) decomposition of an MMPP results in several MMPP's which have the same state space as the original MMPP. Lastly, in some cases, the MMPP can be easily decomposed into a number of ON-OFF processes which can then be used to characterize the MMPP using just a few parameters.

Besides enjoying all the nice properties stated above, BMMPP provides another degree of freedom, the batch size, in performance modeling. In terms of modeling bursty traffic such as the ones expected for the emerging B-ISDN, BMMPP can create far more burstier traffic stream than MMPP can (by specifying the bulk size). Yet, only one [86] was found that analyzes BMMPP queueing systems with an exponentially distributed service time. In sections 2.2 and 2.3, we analyzed the arrival process and the queue length distribution of a more general case, a BMMPP queueing system with an arbitrary service time distribution. We also specialized our results to some special cases, i.e., $M^{[X]}/G/1$ and MMPP/G/1 queues, whose corresponding results are readily available in the literature. The results appear to be similar in their complexity with that of MMPP/G/1 queues but much simpler and more comprehensible than that for BMAP/G/1 queues derived in [60] and [67].

We use these results to compute the loss probability of a BMMPP/G/1/K queueing system. Specifically, by using the *Law of Large Number for Markov Chains* (Theorem 9.4 of [5]), we prove that the loss probability of a BMMPP/G/1/K
queueing system can be easily calculated. Using two other theorems recently introduced by Choudhury and Lucantoni, we develop a very simple algorithm to find the asymptote of the queue length tail probabilities for a BMMPP/G/1 queue with an infinite buffer. We also include a computational algorithm to compute the expected system delay for a BMMPP/G/1 queue, which eliminates most of the computation in solving the system and can be useful in real-time traffic control.

Chapter 3

Integrated Voice and Data Traffic

In this chapter, we study an ATM multiplexer loaded with voice and data sources. The cell arrival process from integrated voice and data sources is approximated by a two-state MMPP with batch arrivals. Our approximation makes use of a new matching technique which leads to more accurate prediction of delay and loss performance. In addition, the computation is less complex than previous similar models. The approximation is verified by simulation and comparisons are made to a different matching procedure proposed by Baiocchi *et al.* [3].

3.1 Model a voice/data ATM multiplexer

We consider an ATM multiplexer loaded with N independent voice sources each of which is modeled by an ON-OFF process with the following parameters:

i. ω, the constant arrival rate in the ON-state;

ii. α, the transition rate from the OFF-state to the ON-state;

iii. β, the transition rate from the ON-state to the OFF-state.

Let C denote the capacity (in cells per unit of time excluding the average capacity consumed by data sources, if any) of the multiplexer; $L = \lfloor C/\omega \rfloor$ be the maximum number of active voice sources which the multiplexer can support; T_i be the expected time until the superposition process of the N ON-OFF processes visits state i-1 for the first time starting from state i; and π_k be the steady state probability that k out of the N voice sources are in the ON-state equal to:

$$\pi_k = \binom{N}{k} \left(\frac{\alpha}{\alpha+\beta}\right)^k \left(\frac{\beta}{\alpha+\beta}\right)^{N-k}, \quad 0 \le k \le N \tag{3.1}$$

To find T_i, let us first observe that it satisfies the following recurrence relation:

$$T_i = \frac{1}{(N-i)\alpha + i\beta} + \frac{(N-i)\alpha}{(N-i)\alpha + i\beta}(T_{i+1} + T_i), \quad 1 \le i \le N-1 \tag{3.2}$$

with the boundary condition, $T_N = 1/(N\beta)$. Note that, in (3.2), the first term on the right-hand

side gives the expected sojourn time in state i. The first part of the second term on the right-hand side specifies the probability that the process will make a transition to state i+1 given that it is currently in state i, and the second part is due to that fact that the process must now go from i+1 to i and then from i to i-1. It can be easily proved, by successive substitution, for example, that T_{N-i} satisfies the following equation:

$$T_{N-i} = \sum_{k=0}^{i} \frac{\frac{i!}{k!}\alpha^{i-k}}{\frac{(N-k)!}{(N-i-1)!}\beta^{i+1-k}}$$

Thus, T_{N-i} has the following explicit closed-form expression:

$$T_{N-i} = \frac{1}{N\beta\binom{N-1}{i}} \sum_{k=0}^{i} \binom{N}{k} \left(\frac{\alpha}{\beta}\right)^{i-k}, \quad 0 \le i \le N-1 \tag{3.3}$$

Using (3.3), we can find the expected time until the system load drops below the system capacity, T_{L+1}, once the load exceeds the system limit, L.

Our state reduction technique works as follows: referring to Fig. 3.1, the states of the superposition arrival process are divided into two disjoint subsets, the overload states, $\{L+1, L+2, \ldots, N\}$, and the underload states, $\{0, 1, 2, \ldots, L\}$. These two sets of states are mapped into the state-II (the overload-state) and the state-I (the underload-state) of a two-state MMPP respectively.

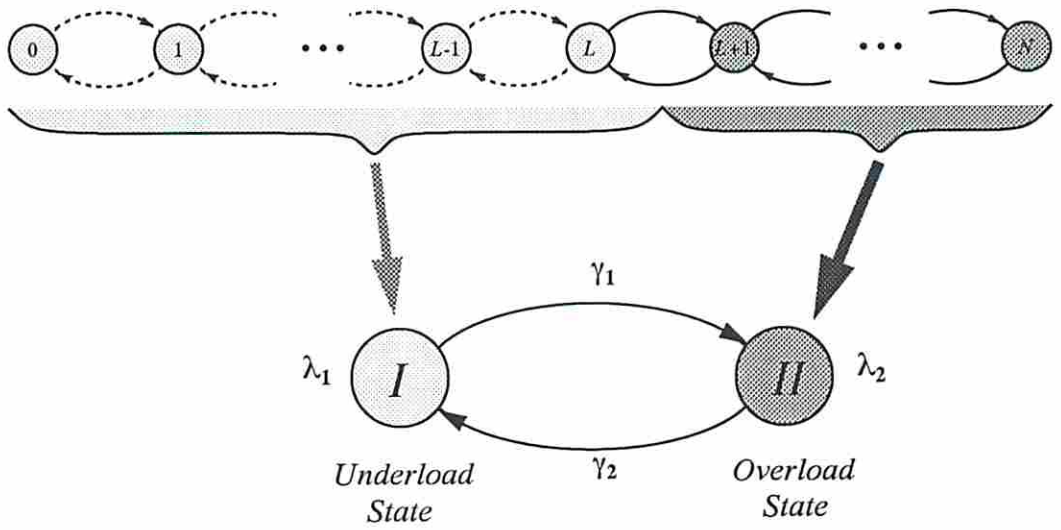

Fig. 3.1 State mapping between the arrival process and the two-state MMPP for superposition of N voice sources.

In the above figure, γ_1 and γ_2 are the transition rates and λ_1 and λ_2 are the cell arrival rates for state-I and state-II correspondingly. We then propose the following approach to match the parameters (note that, *ii-iv* are effectively the same as those used in [3]):

i. $\gamma_2 = \frac{1}{T_{L+1}} = \left(\frac{1}{N\beta \binom{N-1}{N-L-1}} \sum_{k=0}^{N-L-1} \binom{N}{k} \left(\frac{\alpha}{\beta}\right)^{N-L-1-k} \right)^{-1}$

ii. $\lambda_1 = \omega \sum_{k=0}^{L} k\left(\frac{\pi_k}{\Pi_u}\right)$, where $\Pi_u = \sum_{k=0}^{L} \pi_k$

iii. $\lambda_2 = \omega \sum_{k=L+1}^{N} k\left(\frac{\pi_k}{\Pi_o}\right)$, where $\Pi_o = \sum_{k=L+1}^{N} \pi_k$

iv. $\gamma_1 = \gamma_2 \frac{\omega\phi - \lambda_1}{\lambda_2 - \omega\phi}$, where $\phi = N\left(\frac{\alpha}{\alpha+\beta}\right)$ is the expected number of active calls.

As one might expect, the arrival stream from the superposed process is burstier than that from the corresponding two-state MMPP especially when the two-state MMPP is in its overload-state (when there are a relatively larger number of voice sources in talkspurt). This can be understood by observing the "smoothing" effect introduced by the reduced number of states in the approximate model, because when the two-state MMPP is in either one of its states the arrival process is just an ordinary Poisson process rather than each source emitting cells at the peak rate while it is "*ON*" and none while it is "*OFF*." Since, on the average, a longer queue is expected by a new arrival for a burstier arrival process due to a shorter interarrival time within the same burst, the approximation is expected to perform worse as the system load increases, i.e, the number of active sources and the burstiness of the arrival process increase. The same problem is also experienced in the asymptotic matching procedure proposed in [3].

3.2 Improve the accuracy of the approximation

In order to improve the accuracy of our model, let us observe that if we overestimate T_{L+1}, we may, in fact, improve the model's accuracy. This is because the arrival rate in the overload-state is higher than that in the underload-state (see step *ii* and *iii* of the matching procedure on page 27). Although increasing the average overload time also causes an increase in the average underload time (see step *iv* of the matching procedure on page 27), staying in the overload-state longer each visit would mean a higher burstiness (for the same overall arrival rate). Therefore, overestimating the average overload time, T_{L+1}, will increase the burstiness of the two-state MMPP. Note also that (for the same number of voice calls) the number of states which are mapped into the overload-state of the two-state MMPP increases, i.e., L decreases, as the channel capacity decreases. This corresponds to the MMPP staying a longer time in the overload-state on the average. In addition, the longer the average overload time is the more severe the underestimation of the burstiness is. Thus, a higher overestimation for T_{L+1} is needed to offset the increasingly severe underestimation of the burstiness.

One approach that overestimates T_{L+1} is to replace (3.3) by its upper-bound and one upper-bound, which leads to very good results, is:

$$\frac{i+1}{N\beta\binom{N-1}{i}} \max_{0 \le k \le i} \binom{N}{k} \left(\frac{\alpha}{\beta}\right)^{i-k} \tag{3.4}$$

By applying *Stirling's formula* [22] to (3.4), we have:

$$\frac{i+1}{N\beta\binom{N-1}{i}} \max_{0 \le k \le i} O(\sqrt{N}) \left(\frac{N^N}{k^k (N-k)^{N-k}}\right) \left(\frac{\alpha}{\beta}\right)^{i-k}$$

$$= \frac{i+1}{N\beta\binom{N-1}{i}} \max_{0 \le k \le i} O(\sqrt{N}) e^{\{N\log N - (N-k)\log(N-k) - k\log k + (i-k)\log\left(\frac{\alpha}{\beta}\right)\}}$$

$$= \frac{i+1}{N\beta\binom{N-1}{i}} \max_{0 \le k \le i} O(\sqrt{N}) e^{\{N\left[-\frac{N-k}{N}\log\left(\frac{N-k}{N}\right) - \frac{k}{N}\log\left(\frac{k}{N}\right)\right] + (i-k)\log\left(\frac{\alpha}{\beta}\right)\}} \tag{3.5}$$

The problem now reduces to determining which value of k that provides the largest exponent in (3.5). Let us replace k/N by x and rewrite the exponent of (3.5) as:

$$N\left[(x-1)\log(1-x) - x\log x + \left(\frac{i}{N} - x\right)\log\left(\frac{\alpha}{\beta}\right)\right] \tag{3.6}$$

After taking the derivative of (3.6) and equating to zero, we have:

$$\log(1-x) - \log x - \log\left(\frac{\alpha}{\beta}\right) = 0 \tag{3.7}$$

From (3.7), it can be easily verified that (3.6) is maximized at $k = \frac{N\beta}{\alpha+\beta}$. Thus, we conclude that if

$$i < \frac{N\beta}{\alpha+\beta} \tag{3.8}$$

$k = i$ should be chosen to find the upper-bound for (3.3); otherwise, $k = \left\lfloor \frac{N\beta}{\alpha+\beta} \right\rfloor$ should be used.

Theorem 3.1 For any stable system, i.e., with a system utilization $\rho = \frac{N\alpha\omega}{C(\alpha+\beta)} < 1$, $N - L - 1 < \frac{N\beta}{\alpha+\beta}$ always holds.

Proof. Assume the theorem is not true then for some α, β, and N:

$$N - L - 1 \geq \frac{N\beta}{\alpha+\beta}$$

$$\Rightarrow L + 1 \leq N - \frac{N\beta}{\alpha+\beta}$$

$$\Rightarrow \frac{C}{\omega} < L + 1 \leq N - \left(1 - \frac{\alpha}{\alpha+\beta}\right)N$$

$$\Rightarrow \frac{N\alpha\omega}{(\alpha+\beta)C} > \frac{\alpha+\beta}{\alpha}\left[1 - \left(1 - \frac{\alpha}{\alpha+\beta}\right)\right]$$

$$\Rightarrow \rho > 1$$

which contradicts the assumption. Therefore, the theorem is true. ❑

Theorem 3.1 implies that for the systems of interest, (3.8) always holds for $i = N - L - 1$. The upper-bound that we found for $T_{N-(N-L-1)}$, denoted $\tilde{T}_{L+1}$, is then given by:

$$\tilde{T}_{L+1} = \frac{N-L}{N\beta\binom{N}{N-L-1}}\binom{N}{N-L-1} = \frac{N-L}{L+1}\cdot\frac{1}{\beta} \tag{3.9}$$

Note that, in the extreme case, i.e., when $L + 1 = N$, $\tilde{T}_N$ is equal to T_N. Furthermore, for the same number of voice calls, as the value of $L + 1$ decreases, i.e., the channel capacity decreases, the value of $(\tilde{T}_{L+1} - T_{L+1})$ increases. This is precisely how we want $\tilde{T}_{L+1}$ to behave, as was pointed out at the end of previous section. We can now refine our matching procedure using the result of (3.9) to be the following:

i. $\gamma_2 = \dfrac{1}{\tilde{T}_{L+1}} = \dfrac{(L+1)\beta}{N-L}$

ii-iv. (*as before*).

3.3 Add data traffic to the model

We have thus reduced the system description to a two-state MMPP to represent the arrival process from a number of voice sources. To include data sources into the model, let us assume that data sources generate data packets according to a *Poisson process* with rate μ. The packet length (in number of cells) has an arbitrary probability mass function, P_k=*Pr*{data packet is comprised of k cells}. This is the same as saying that the data cell arrival process is a *batch-arrival Poisson process* with a random batch size. If we combine the data sources with the two-state MMPP, we have a two-state MMPP with batch arrivals. The transition rates, γ_1 and γ_2, are determined by the matching procedure. The arrival rates are $\lambda_1 + \mu$ and $\lambda_2 + \mu$ for the underload-state and the overload-state respectively. The probability that a batch has a size of k cells is:

$$P_k \frac{\mu}{\mu + \lambda_1} + \delta_{k-1} \frac{\lambda_1}{\mu + \lambda_1}, \quad k = 1, 2, 3, \ldots \tag{3.10}$$

if the process is in the state-I; and:

$$P_k \frac{\mu}{\mu + \lambda_2} + \delta_{k-1} \frac{\lambda_2}{\mu + \lambda_2}, \quad k = 1, 2, 3, \ldots \tag{3.11}$$

if the process is in the state-II, where:

$$\delta_k = \begin{cases} 1, & \text{if } k = 0 \\ 0, & \text{otherwise} \end{cases}$$

3.4 Numerical results

We model a voice/data ATM multiplexer as a continuous-time two-state BMMPP/D/1 queue and the results derived in Chapter 2 are applied to get various approximate performance figures. In these examples, each voice call is characterized by $\omega = 1/6$ cells/msec (this corresponding to a 64 Kbps PCM coding with a speech activity detector and a standard 48-octet payload size), $\alpha = 1.538$ and $\beta = 2.778$ (according to the conclusions drawn by [8]). The aggregated data packet arrival rate is assumed to be $20N_d/3$ packets per second, where N_d (an user-defined parameter) is the number of data calls. We assume that the packet size is geometrically distributed with an average of 5 cells per packet. In all figures presented in this section, BMMPP-1 is the

model suggested by [3] extended to include data sources. BMMPP-2 and BMMPP-3 are the models proposed by this research using equation (3.9) and equation (3.3) respectively. For Fig. 3.2 and Fig. 3.3, we fix the system load (by having a constant number of voice and data calls) and plot average system time versus channel utilization (by changing the channel capacity). 20 voice calls plus 20 data calls and 200 voice calls plus 200 data calls are assumed for Fig. 3.2 and Fig. 3.3 correspondingly. For Fig. 3.4, which shows the relationship between average system time and channel capacity, a constant system utilization of 0.9 and a fixed ratio of the number of voice and data calls of 3:5 are assumed. We see that BMMPP-2 which is based on the approximation method on a bound for the expected time in overload gives the best results.

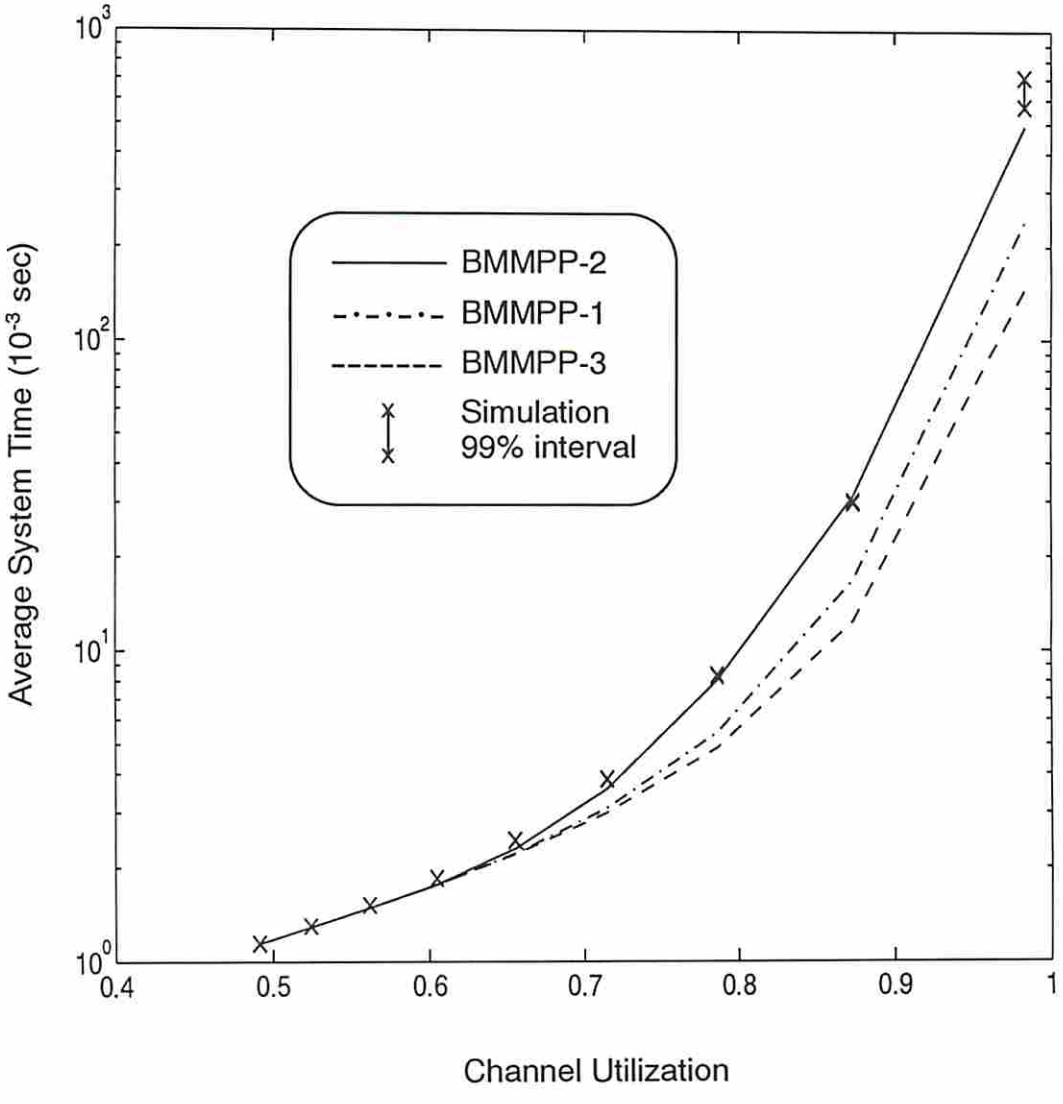

Fig. 3.2 Expected delay vs. channel utilization for 20 voice calls and 20 data calls.

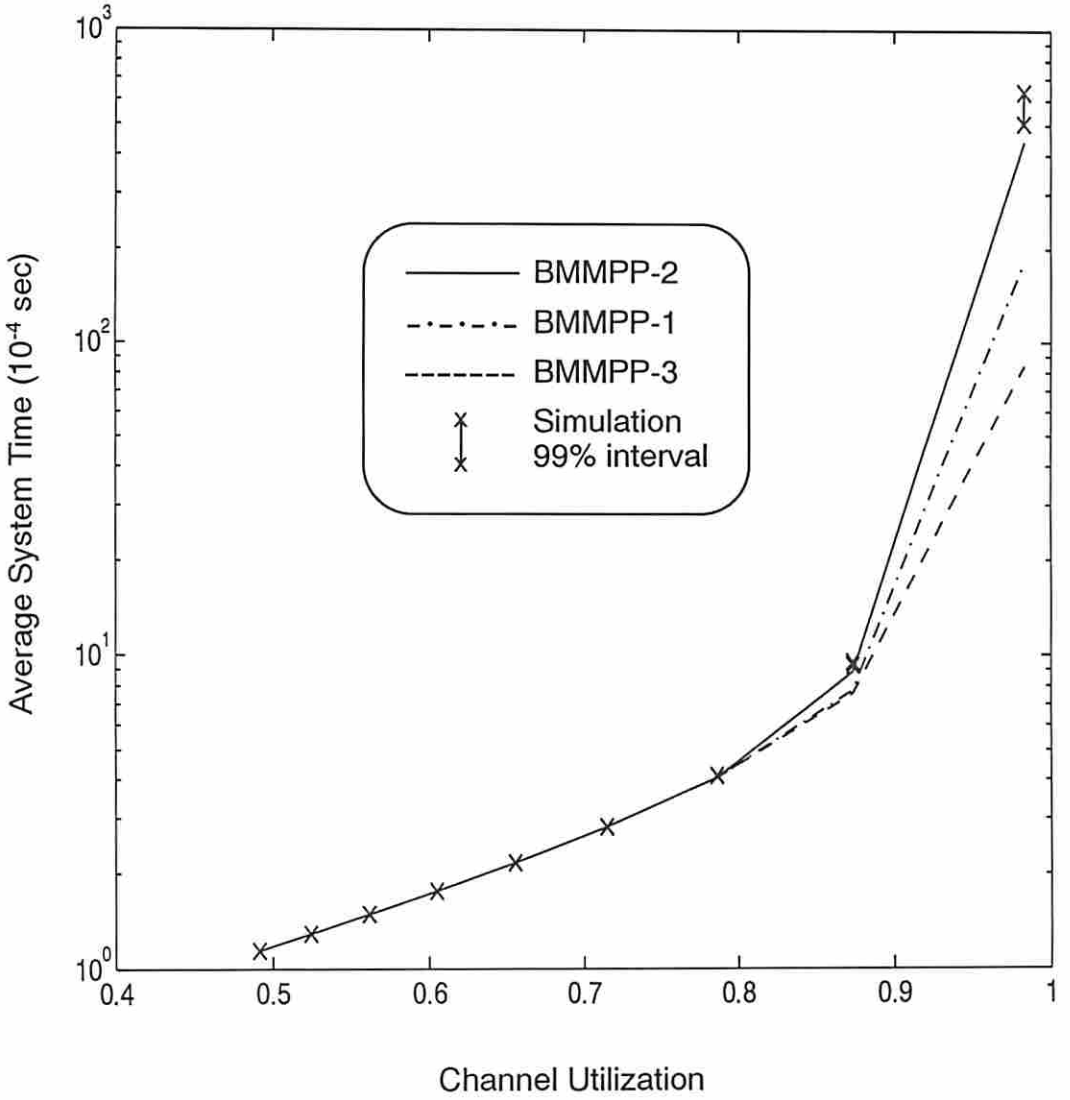

Fig. 3.3 Expected delay vs. channel utilization for 200 voice calls and 200 data calls.

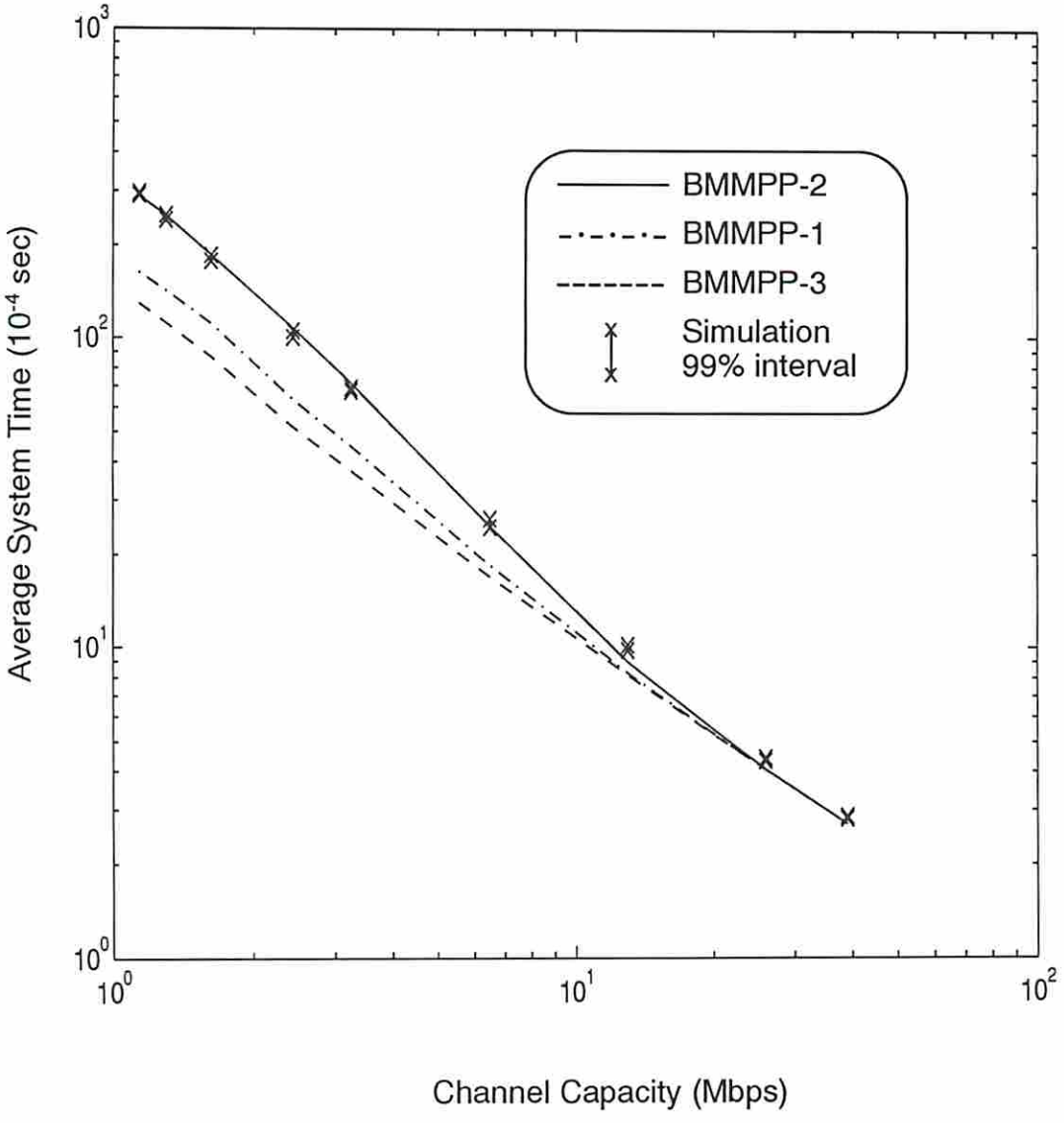

Fig. 3.4 Expected system time vs. channel capacity for $\rho = 0.9$.

Fig. 3.5 through Fig. 3.8 show the tail probabilities for different system utilizations. In these plots, the solid line represents the tail probabilities obtained by simulating the exact model. The dashed lines are the tail probability and its asymptote for the proposed approximation (denoted BMMPP-2) obtained analytically as discussed in Chapter 2. The dot-dashed lines are the tail probability and its asymptote using the approximation model presented in [3] extended to allow batch arrivals (denoted BMMPP-1). Note that in Fig. 3.5 the dot-dashed lines overlap the dashed lines. As can be seen from these figures, both approximations perform well for a moderate system load, say 0.6 where the simulation drops off at a buffer position > 160 because too few events were observed. With an increasing system load, however, the results from the approximation presented in [3] diverge from the simulation for large buffer positions, while the proposed model stays within a reasonably close range. Fig. 3.9 shows the cell loss probabilities for different system loads. Note that, in this figure, the dashed line and dot-dashed line overlay each other for ρ=0.6. Again, by comparing to the simulation results, we readily see that the proposed approximation method outperforms the one presented in [3] for all ranges of system loads. We use Fig. 3.10 to demonstrate that tail probabilities, which are generally used as an estimation of loss probabilities (see, for example, [16] and [34]), fail to predict loss probabilities for heavily loaded systems. In this figure, dotted lines represent cell loss probabilities, while solid lines represent the tail probabilities (both obtained from simulation). As shown in this figure: using tail probabilities would overestimate the loss probabilities as high as 14 times according to the simulation for a system with a buffer size of 200 and a system load of 0.9. Even for a system load of only 0.7, the tail probabilities still overestimate the loss probabilities by a factor of 5.

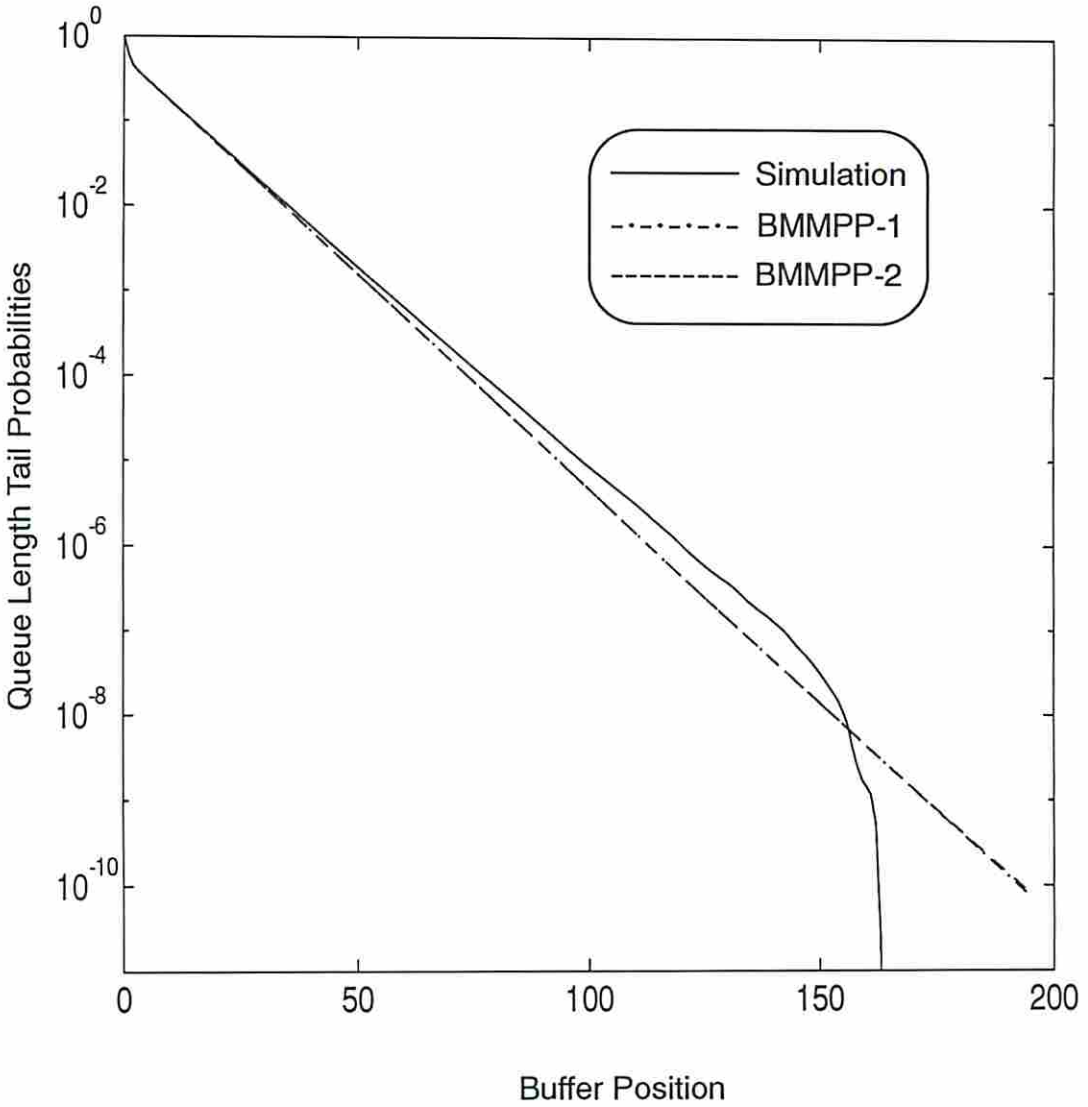

Fig. 3.5 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.6.

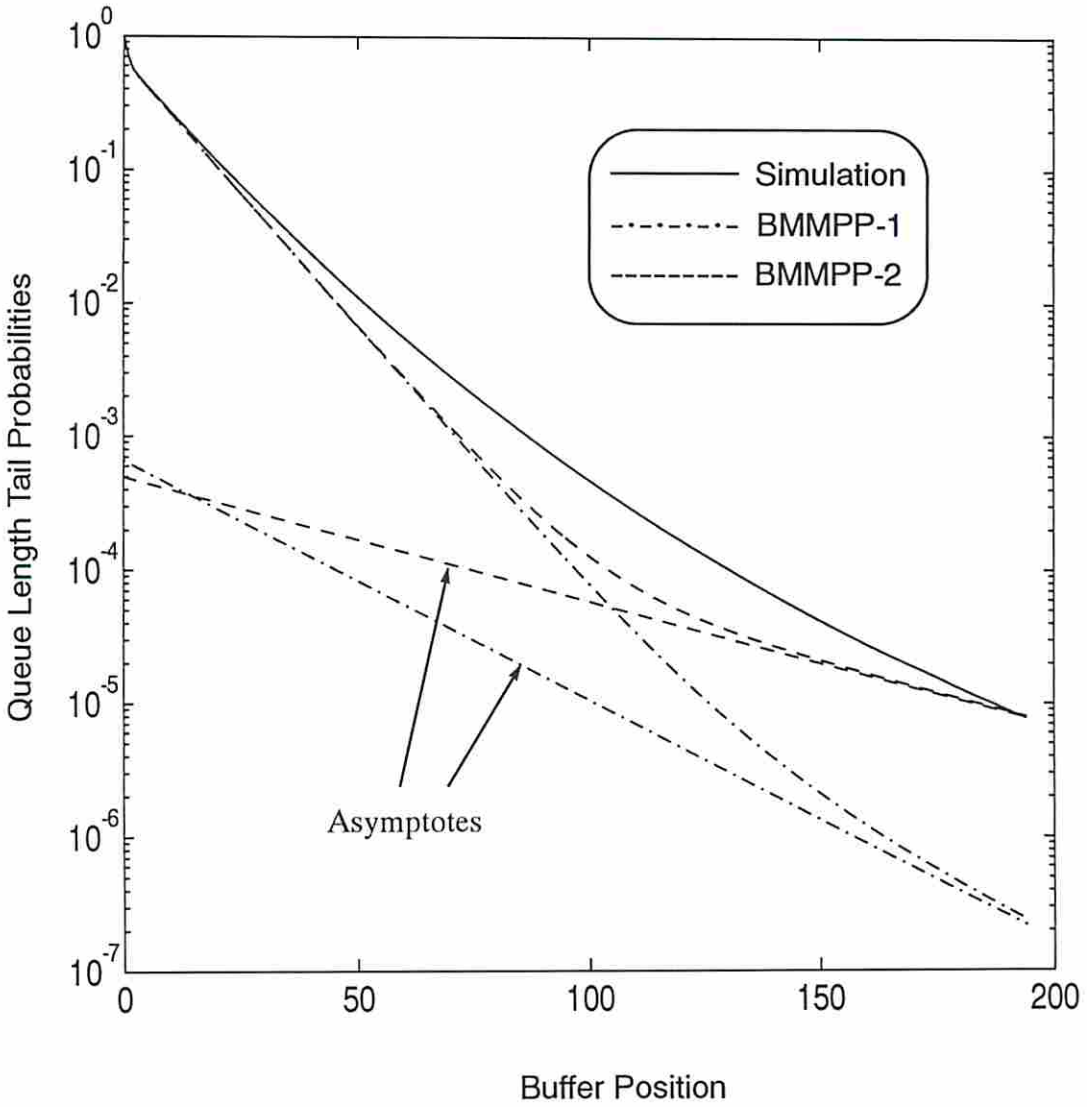

Fig. 3.6 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.7.

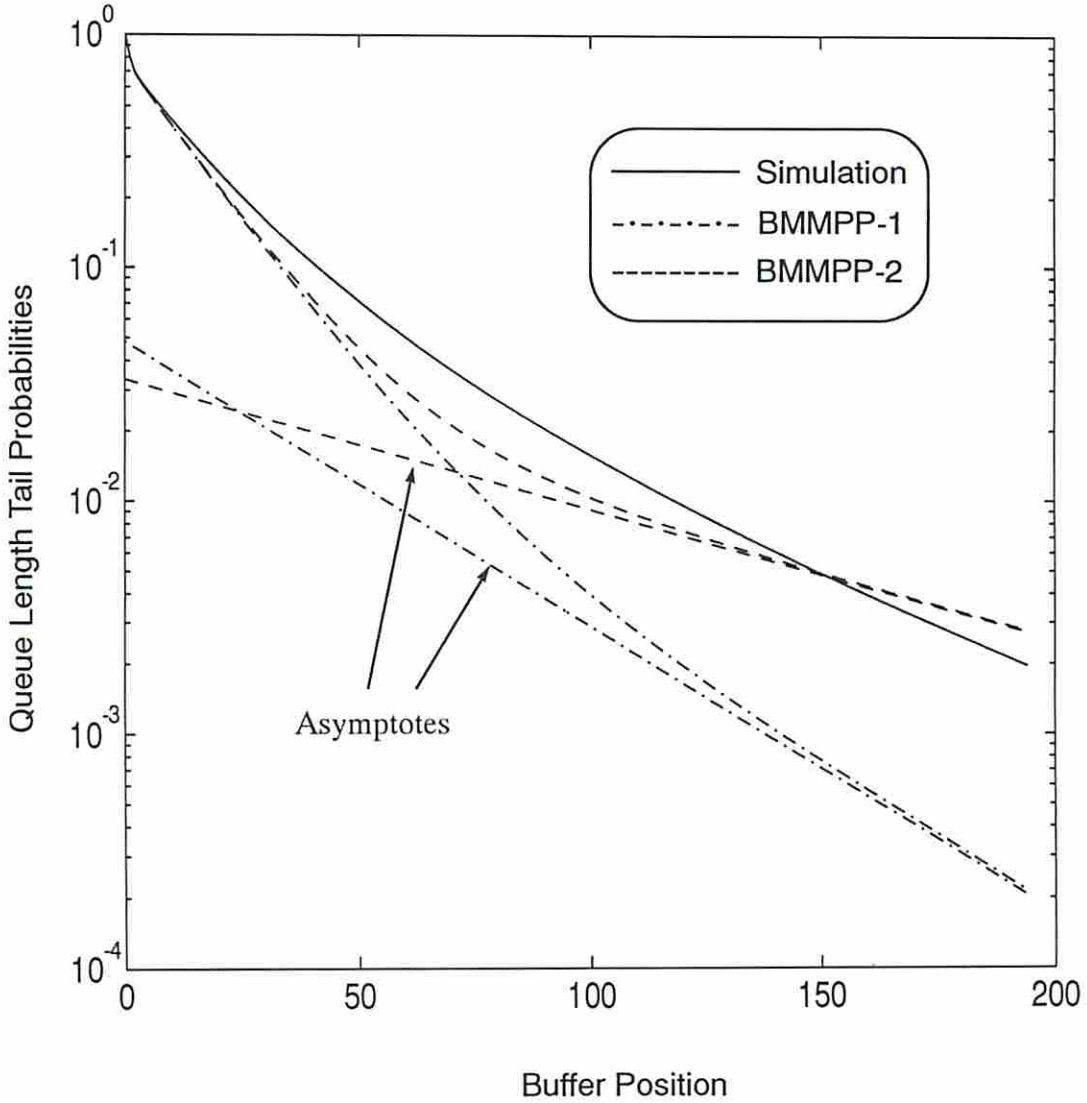

Fig. 3.7 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.8.

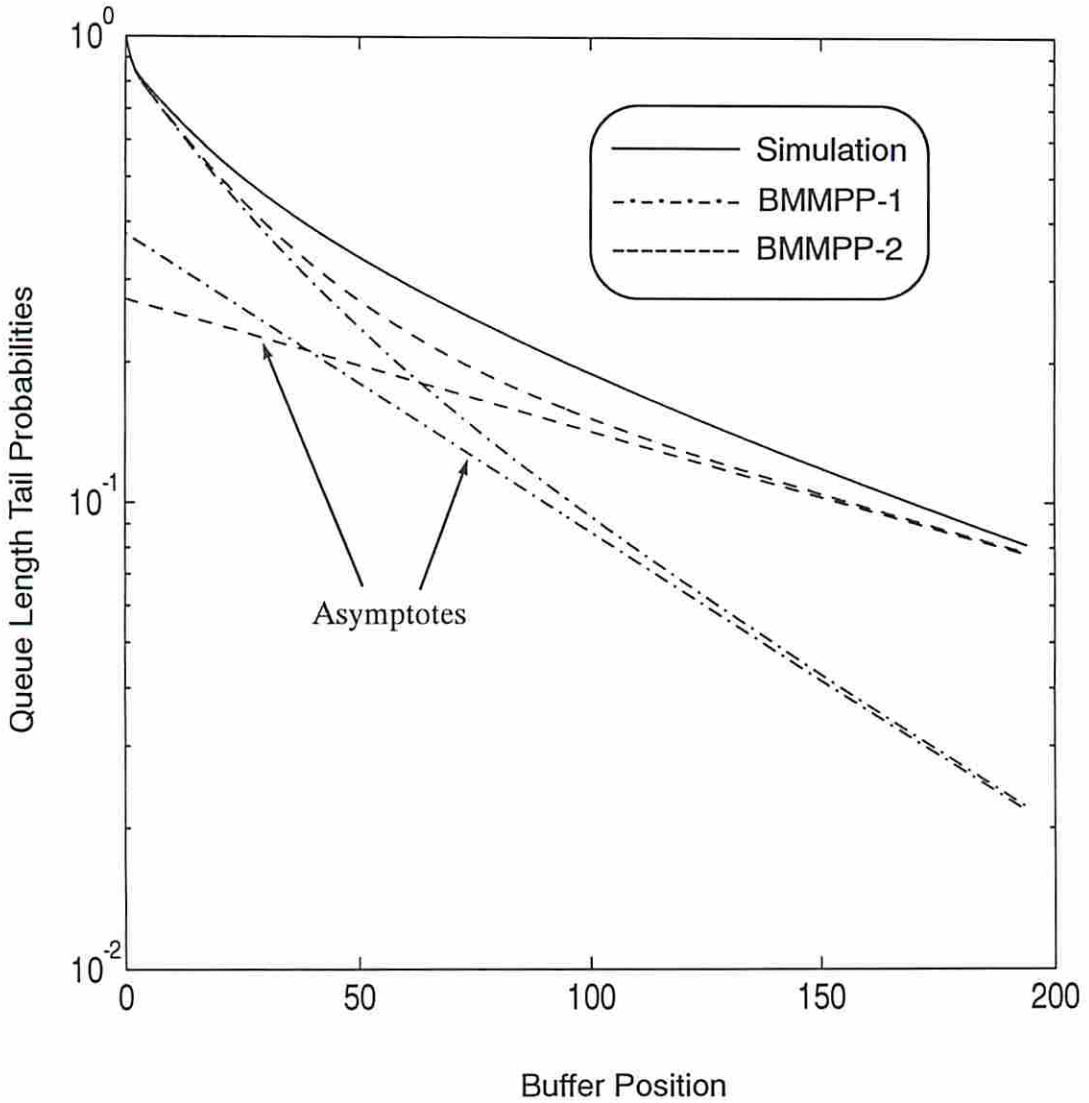

Fig. 3.8 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.9.

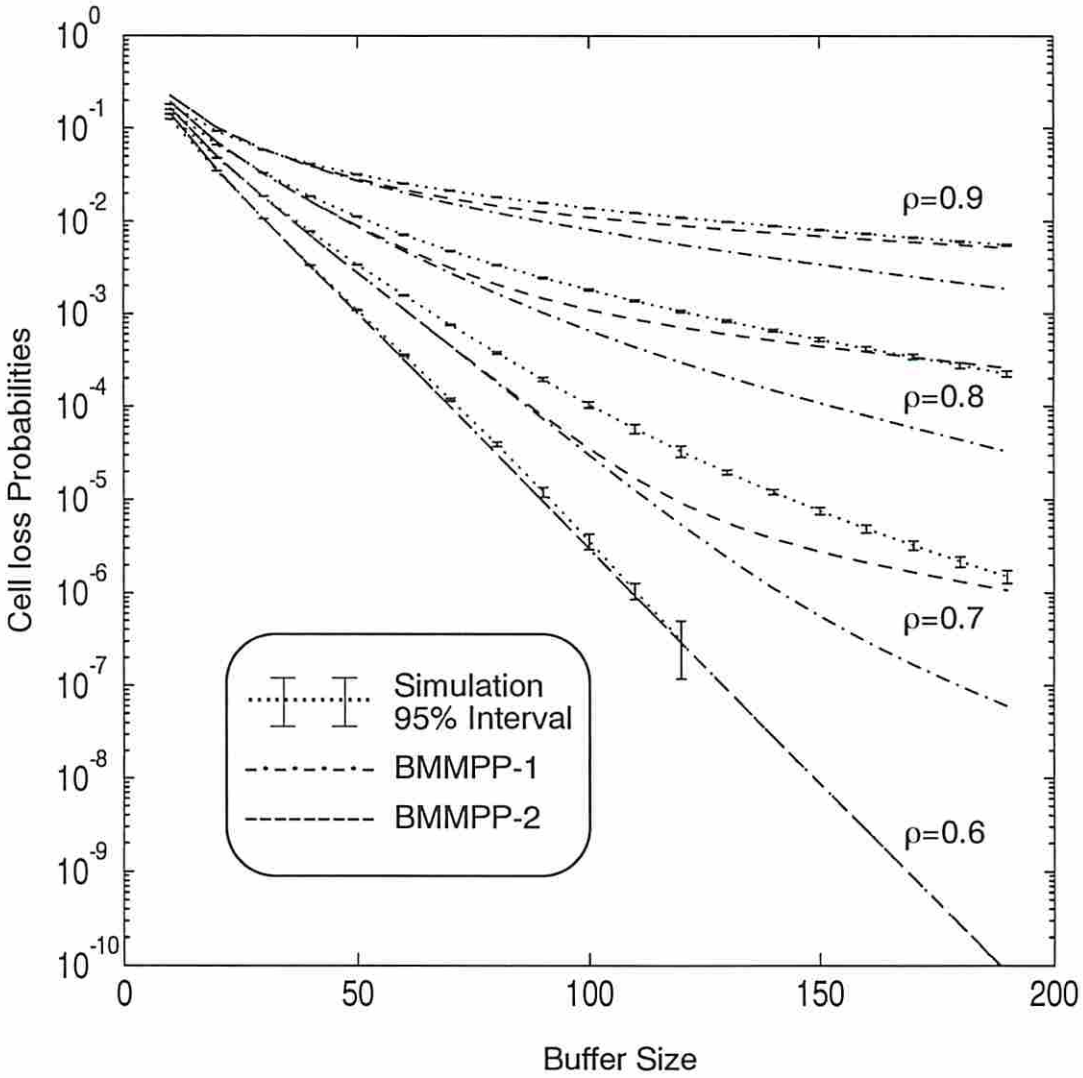

Fig. 3.9 Cell loss probabilities for 20 voice calls and 50 data calls with different system loads.

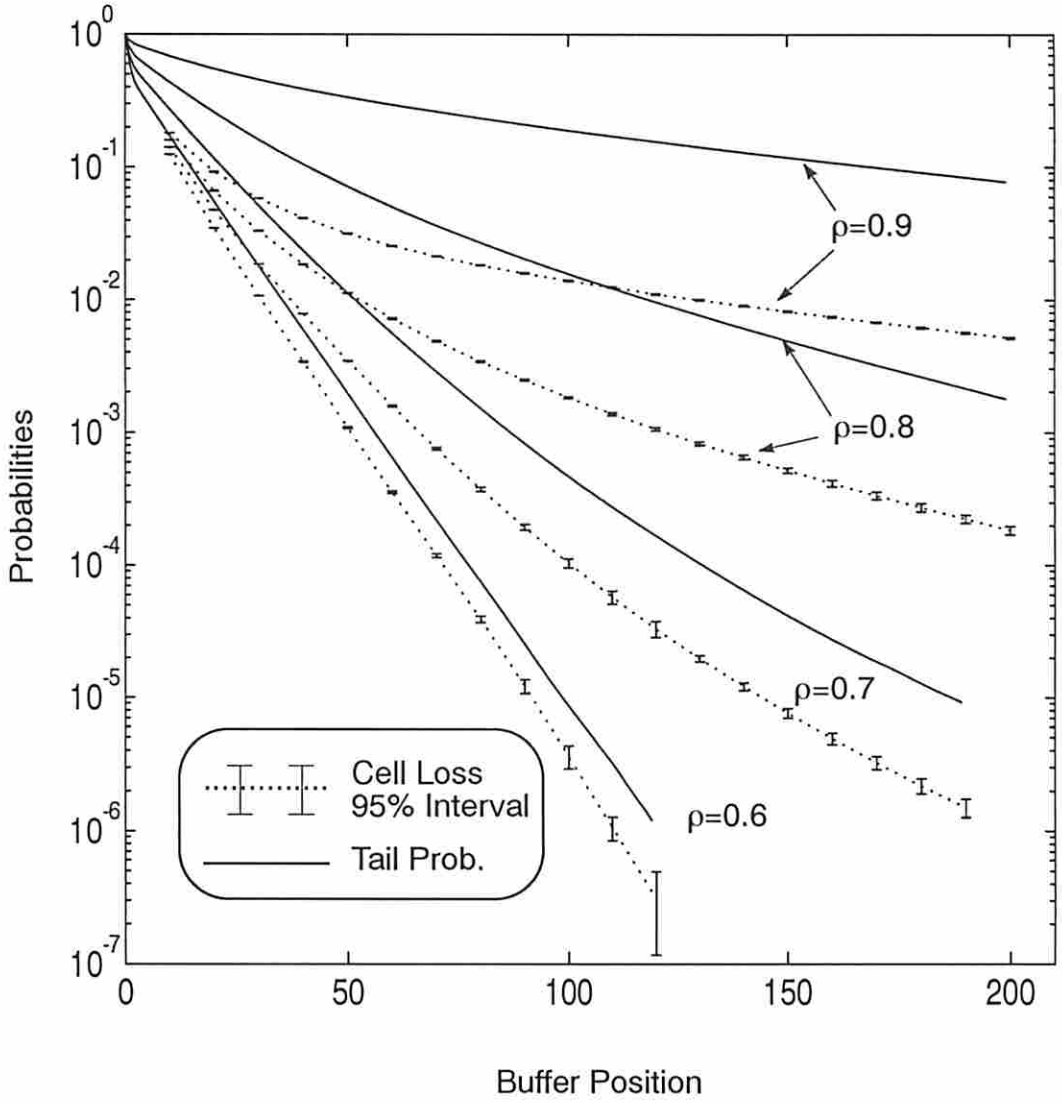

Fig. 3.10 Tail probabilities versus loss probabilities for 20 voice calls and 50 data calls (simulation results).

Chapter 4

Integrated Video, Voice, and Data Traffic

In this chapter, we extend the approach used in Chapter 3 to approximate the superposition of video and voice sources by a two-state MMPP. Data traffic are included into the model in the same way as in the previous case. Since there is are similar approaches in the literature, no comparison is made. Instead, simulation is used to verify the accuracy of the extension of the approach.

4.1 Superposition of the source models

We assume that the video sources are of the videotelephony type (showing a person talking in front of the camera without sudden movements) and use a *continuous-time Markov chain* to model the superposition of several video sources, as suggested by Maglaris *et al.* (Fig. 1.7). In this model, the output bit rate from a codec is assumed to fall into a set of discrete bit rate levels evenly spaced between zero and the peak rate. Transitions are assumed to occur between adjacent levels only, according to a *birth-death Markov chain*. The transition rates are determined by matching the model's parameters with the statistical characteristics of the traffic, as described in [53]. Combining it with the model for voice sources used in the previous chapter results in a *two-dimensional continuous-time Markov chain*. Fig. 4.1 shows the transition diagram for such a process, where N and M denote the number of voice sources and the number of discrete bit-rate levels for all video sources respectively. Let π_{ij} be the steady state probability of the process being in state (i, j). It can be verified that:

$$\pi_{ij} = \binom{N}{j}\binom{M}{i}\left(\frac{a}{a+b}\right)^{i}\left(\frac{b}{a+b}\right)^{M-i}\left(\frac{\alpha}{\alpha+\beta}\right)^{j}\left(\frac{\beta}{\alpha+\beta}\right)^{N-j} \tag{4.1}$$

for $0 \le i \le M$ and $0 \le j \le N$.

Remarks. One can easily write down the set of *global-balance equations* for π_{ij}. But, let us assume that π_{ij}'s also satisfy the following *local-balance equations*:

$$\begin{cases} j\beta\pi_{ij} = (N-j+1)\,\alpha\pi_{i,j-1}; & 1 \le j \le N,\ 0 \le i \le M \\ ib\pi_{ij} = (M-i+1)\,a\pi_{i-1,j}; & 1 \le i \le M,\ 0 \le j \le N \end{cases}$$

i.e.,

$$\begin{cases} \pi_{ij} = \dfrac{(N-j+1)}{j}\left(\dfrac{\alpha}{\beta}\right)\pi_{i,j-1} = \binom{N}{j}\left(\dfrac{\alpha}{\beta}\right)^{j}\pi_{i0}; & 1 \le j \le N,\ 0 \le i \le M \\ \pi_{ij} = \dfrac{(M-i+1)}{i}\left(\dfrac{a}{b}\right)\pi_{i-1,j} = \binom{M}{i}\left(\dfrac{a}{b}\right)^{i}\pi_{0j}; & 1 \le i \le M,\ 0 \le j \le N \end{cases}$$

$$\Rightarrow \pi_{ij} = \binom{N}{j}\binom{M}{i}\left(\frac{\alpha}{\beta}\right)^{j}\left(\frac{a}{b}\right)^{i}\pi_{00}; \quad 0 \le i \le M,\ 0 \le j \le N \tag{4.2}$$

Since $\sum_{j}\sum_{i}\pi_{ij} = 1$, (4.2) implies:

$$\frac{1}{\pi_{00}} = \sum_{j=0}^{N}\sum_{i=0}^{M}\binom{N}{j}\binom{M}{i}\left(\frac{\alpha}{\beta}\right)^{j}\left(\frac{a}{b}\right)^{i}$$

$$= \sum_{j=0}^{N}\binom{N}{j}\left(\frac{\alpha}{\beta}\right)^{j}\sum_{i=0}^{M}\binom{M}{i}\left(\frac{a}{b}\right)^{i}$$

$$= \left(1+\frac{\alpha}{\beta}\right)^{N}\left(1+\frac{a}{b}\right)^{M}$$

Hence we conclude that (4.1) is the closed-form expression for the steady-state distribution. Upon substitution, we can also verify that the π_{ij}'s given by (4.1) satisfy the global-balance equations. Thus, the results are indeed correct. ❑

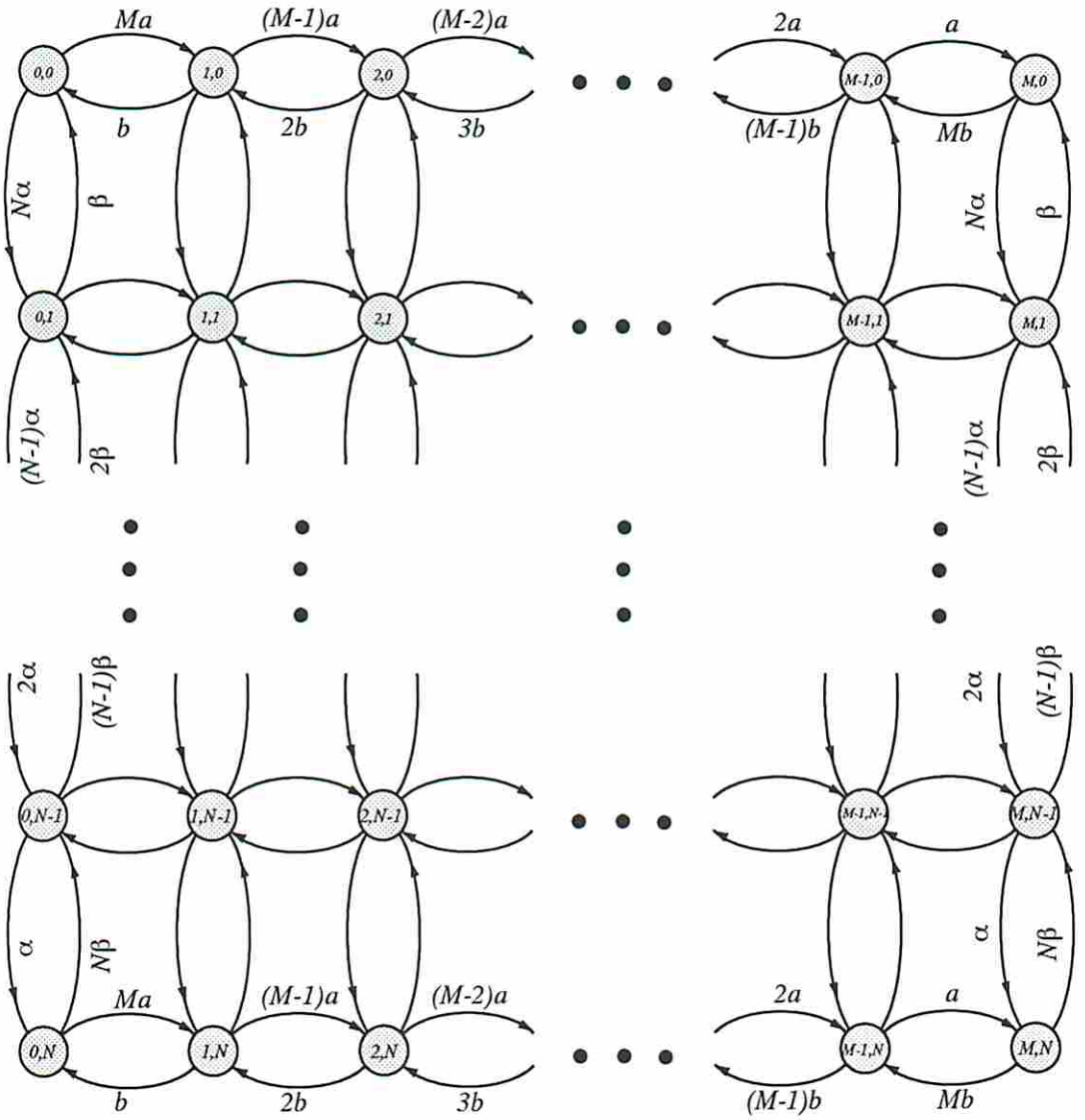

Fig. 4.1 The transition diagram for integrated video and voice traffic.

4.2 Approximation using a two-state BMMPP

To motivate the discussion of our extension to include video sources into the model, let us observe that the expected time that the system will stay in the overload state depends on two major factors: for the same number of traffic sources, (a) the heavier the system load is, the longer each visit to the overload state is; and, (b) the quicker the system can reduce its current load, the faster it will leave the overload state. Note that, in (3.9), $(N-L)/(L+1)$ is the ratio of the number of overload states to the number of underload states, which gives a measure of how heavy the system load is. The other term, $1/\beta$, is the expected time until an active source becomes idle,

Data traffic is introduced into the model by using the same approach as in the voice/data case. Namely, if we assume that data sources generate data packets according to a *Poisson process* with rate μ and P_k is the probability of a data packet being size k (in number of cells), the overall traffic model can then be approximated as a two-state BMMPP with the following four parameters: γ_1, γ_2, $\lambda_1+\mu$, and $\lambda_2+\mu$. The batch size probability mass functions (for different phases) are determined by (3.10) and (3.11).

4.3 Numerical results

We use the same voice and data sources as in the previous chapter and use the same set of parameters for the video sources as used by [7] and [53], i.e., video sources are characterized by: an average bit rate of 3.9 Mbps, a peak bit rate of 10.58 Mbps, a standard deviation of the bit rate of 1.73 Mbps and a parameter for the autocorrelation function of 3.9. The total number of discrete cell arrival rate levels for video sources is assumed to be 16 times the number of video sources (as suggested by [53]). In Fig. 4.3, we assume 100 voice sources and 100 data sources in the background and plot the relation between average system time and the number of video sources under a system utilization of 0.8 (varying the channel capacity). Our model is shown as curve BMMPP-2 and for comparison purposes we plot a curve BMMPP-1 using the minimum of $1/\beta$ and $1/b$ rather then $1/(\beta + b)$ as the measure of how fast the current system load will drop. In Fig. 4.4, we keep the ratio of the number of video, voice and data calls fixed at 1:20:20 and show average system time versus channel capacity with different (constant) system utilizations (i.e., by varying the total traffic load).

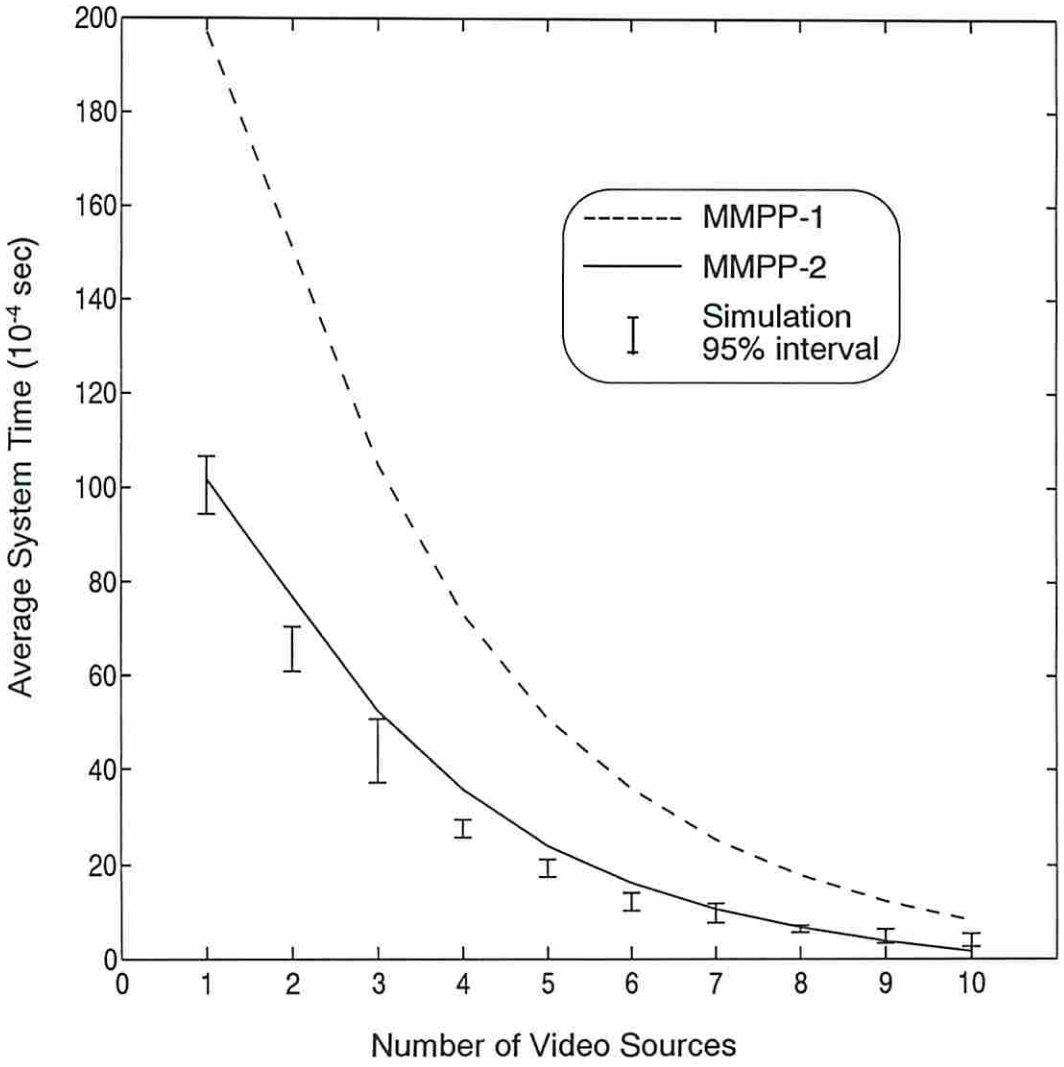

Fig. 4.3 Expected system time vs. number of video sources for ρ=0.8.

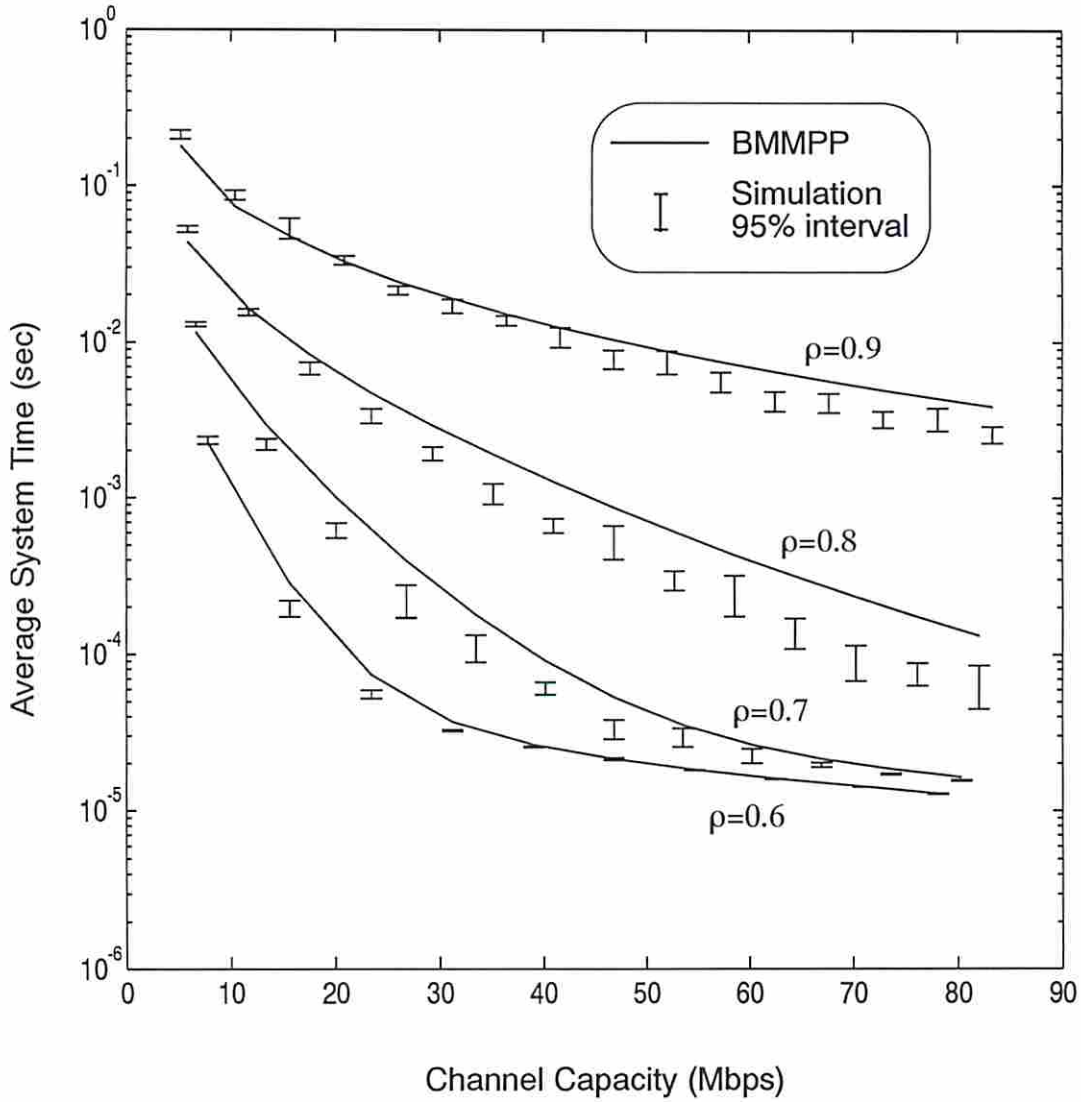

Fig. 4.4 Expected system delay versus channel capacity for different system utilizations.

Fig. 4.5 shows the approximation results on the tail probabilities and its asymptote for different system loads (defined to be $\rho = \left(\frac{\eta aM}{a+b} + \frac{\omega\alpha N}{\alpha+\beta} + \mu\bar{s}\right)/C$). In this example, the numbers of video, voice and data sources are assumed to be 5, 100 and 100 respectively. The corresponding cell loss probabilities for different system loads obtained from both simulation and the proposed approximation are then presented in Fig. 4.6.

We use Fig. 4.7 to demonstrate the impact of the number of video sources on the system's overall cell loss probabilities. In this example, the system load is fixed at 0.8 and the number of voice and data sources is also fixed as 100 each. By varying the system capacity, we plot the cell loss probabilities for systems with 1, 5 and 10 video sources.

Finally, Fig. 4.8 is used to show the difference between the tail probabilities and cell loss probabilities. We assume the same number of sources as in Fig. 4.5 and Fig. 4.6. This plot confirms again (as we concluded in the previous chapter) that using tail probabilities is not a good estimate of the corresponding cell loss probabilities, especially for heavy system loads.

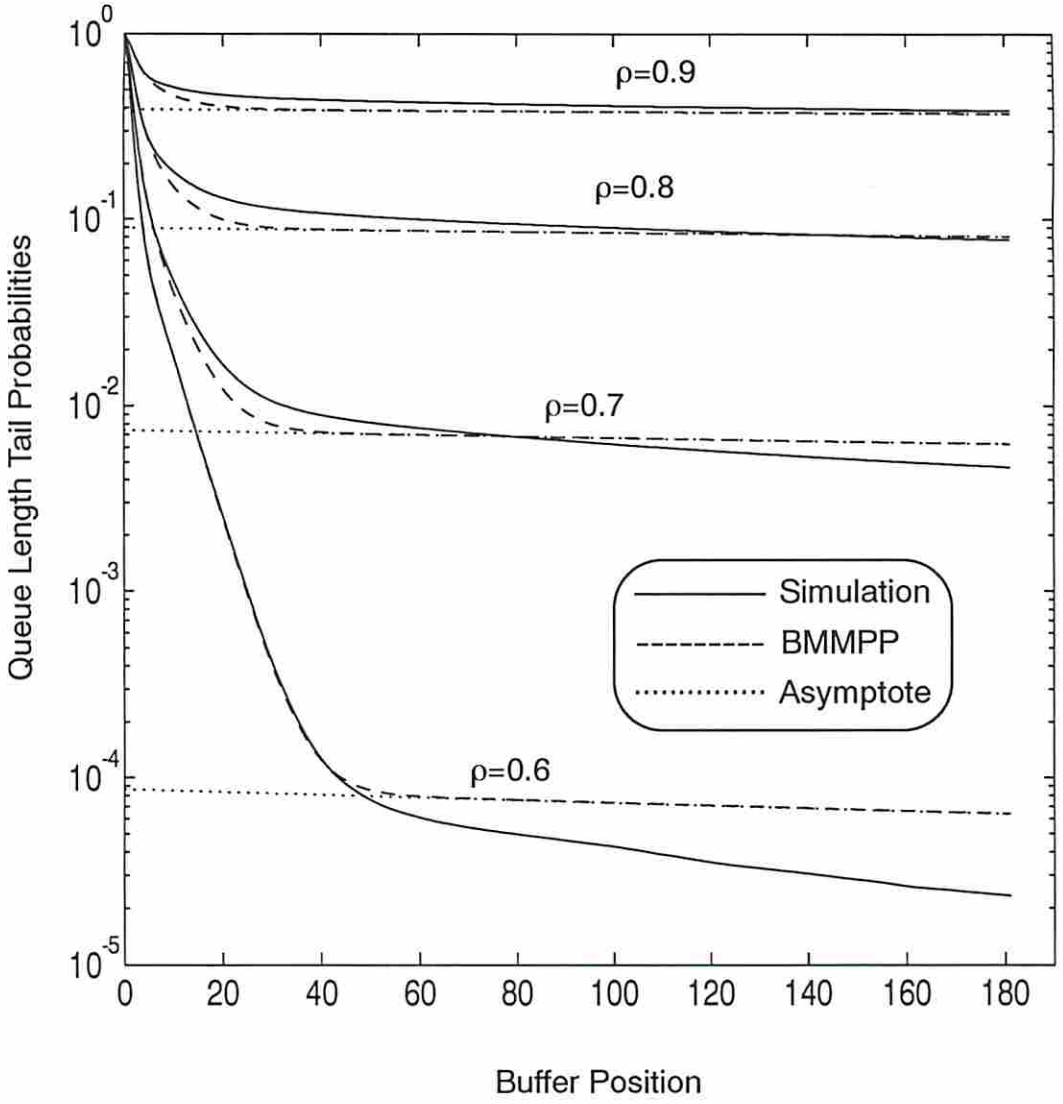

Fig. 4.5 Survivor function and its asymptote for 5 video calls, 100 voice calls and 100 data calls for different system loads.

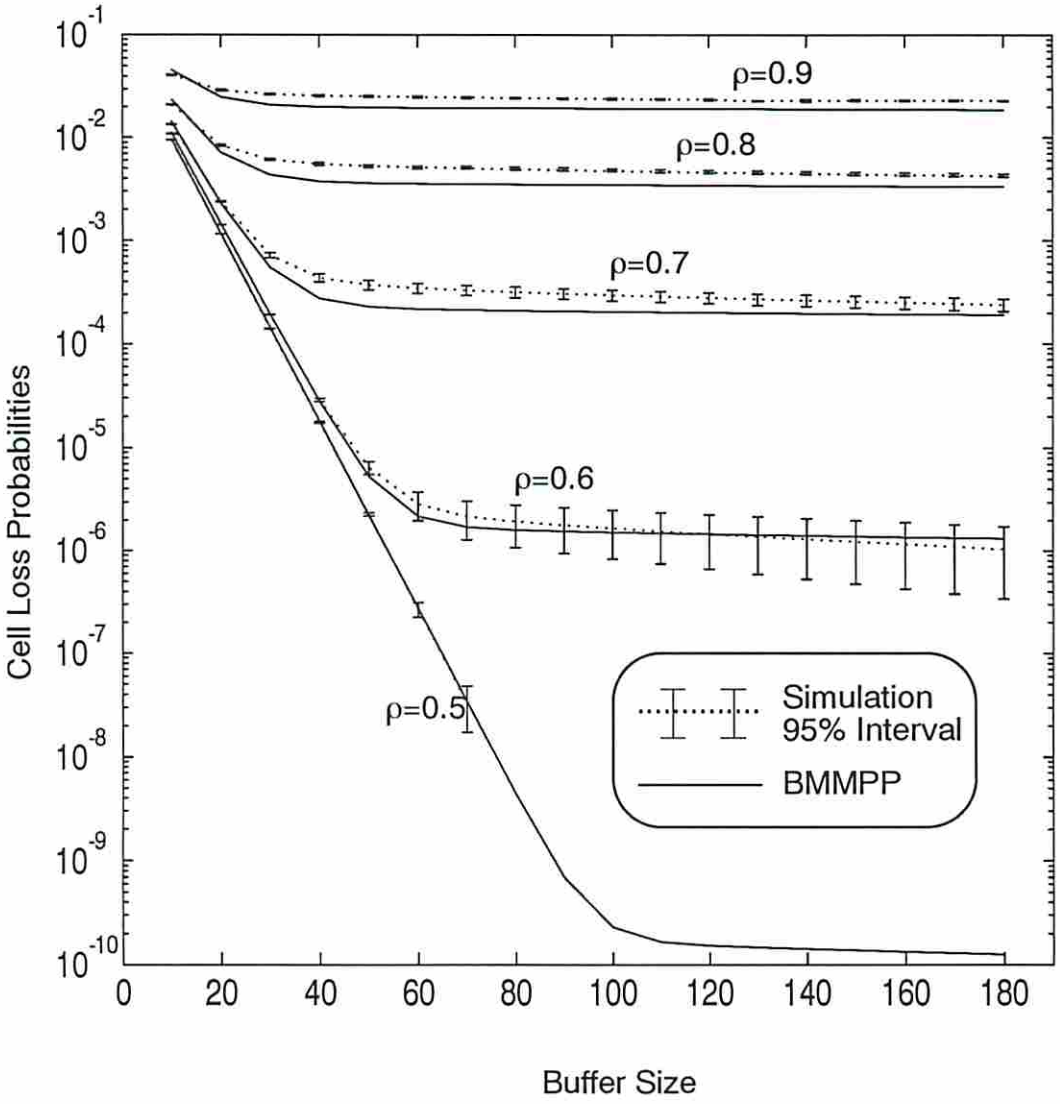

Fig. 4.6 Cell loss probabilities for 5 video calls, 100 voice calls and 100 data call with different system loads.

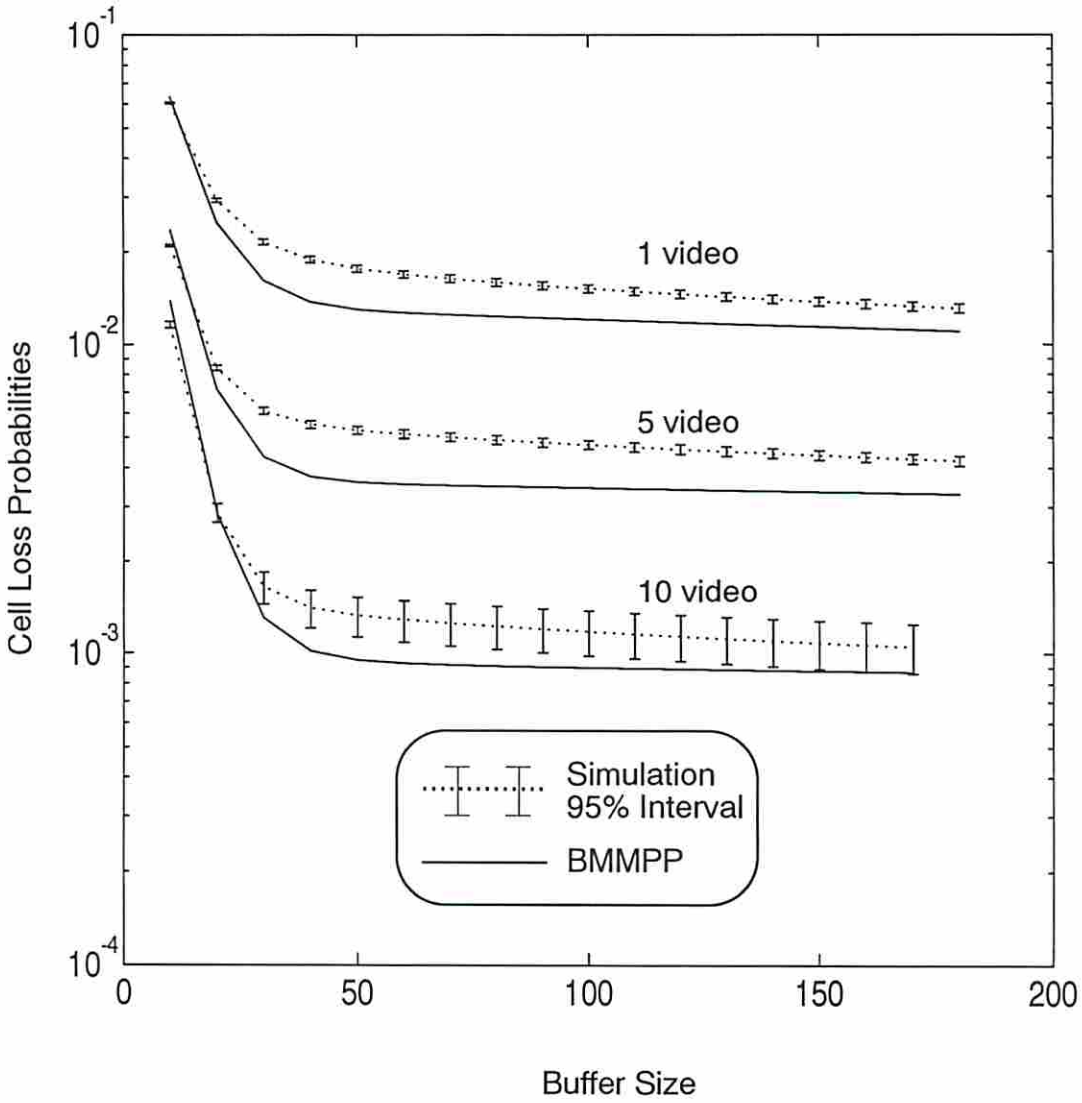

Fig. 4.7 Cell loss probabilities for a system load of 0.8 and 100 voice calls and 100 data calls with different number of video calls.

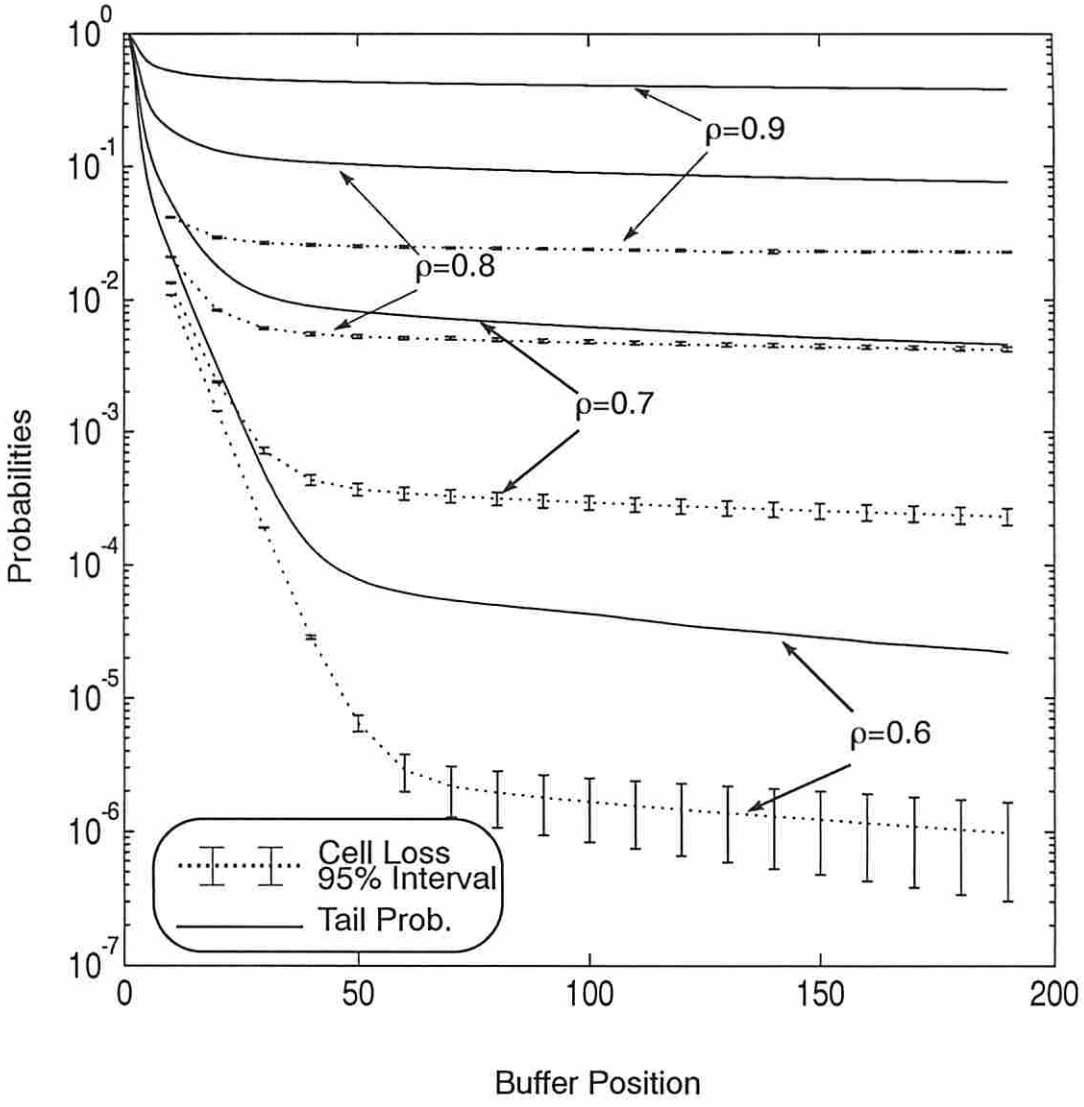

Fig. 4.8 Tail probabilities versus loss probabilities for 5 video calls, 100 voice calls and 100 data calls (simulation results).

Chapter 5

Discrete-Time Models

The fixed cell size used in ATM networks naturally leads to the use of discrete-time models for performance evaluation. In this chapter, we describe our work on estimating ATM multiplexer performance based on *Discrete-time Batch Markovian Arrival Process* (D-BMAP) models. In the continuous-time domain, the performance of an ATM multiplexer loaded with different types of traffic has been approximated by reducing the arrival processes to a two-state BMMPP and has been presented in the previous chapters. In this chapter we extended the approach to the discrete-time domain with accurate results and even faster computation.

5.1 Discrete-time batch Markovian arrival process (D-BMAP)

D-BMAP, which was originally formulated by Blondia and Casals in [7], is the discrete-time analog of BMAP. It can be defined by a two-dimensional Markov chain, $\{ (R_k, J_k), \quad k = 1, 2, 3, \ldots \}$, where R_k is the total number of arrivals by the end of slot k and J_k is the phase of the D-BMAP in slot k. The transition probability matrix of the Markov chain (for m phases) is given by:

$$T = \begin{bmatrix} \boldsymbol{D}_0 & \boldsymbol{D}_1 & \boldsymbol{D}_2 & \boldsymbol{D}_3 & \ldots \\ \mathbf{0} & \boldsymbol{D}_0 & \boldsymbol{D}_1 & \boldsymbol{D}_2 & \ldots \\ \mathbf{0} & \mathbf{0} & \boldsymbol{D}_0 & \boldsymbol{D}_1 & \ldots \\ \cdot & \cdot & \cdot & \cdot & \ldots \\ \cdot & \cdot & \cdot & \cdot & \ldots \end{bmatrix}$$

where $(\boldsymbol{D}_k)_{i,j}$, $0 \le i, j \le m, k = 0, 1, 2, \ldots$, is the probability that there is a phase change from phase i to phase j accompanied by an arrival of size k. Note that we let the probability of bulk size being 0 be the probability of no arrival and normalize the bulk size probability mass function accordingly, i.e, the probability of a bulk arrival of some size is 1 for every phase. Thus, an m-state D-BMAP can be completely defined by:

- p_{ij} the transition probability from phase i to phase j, $1 \le i, j \le m$;
- $q_i(k)$ the conditional probability of bulk size being k, $k = 0, 1, 2, \ldots$, given phase i, $1 \le i \le m$.

From the definition, we can easily see that:

$$\boldsymbol{D}_k = \Delta\vec{q}_k \cdot \boldsymbol{P} \tag{5.1}$$

where $\Delta\vec{q}_k = diag(q_1(k), q_2(k), \ldots, q_m(k))$ is a diagonal matrix and $\boldsymbol{P} = [p_{ij}]$ (a matrix of which the (i, j)th element is p_{ij}), and:

$$\boldsymbol{D} \equiv \sum_{k=0}^{\infty} \boldsymbol{D}_k = \boldsymbol{P} \tag{5.2}$$

5.2 Performance studies of D-BMAP/D/1 queues

We focus our attention on systems with a constant service time (corresponding to the constant transmission time of the fixed-size cells used in ATM networks) and consider a D-BMAP/D/1 queueing system with a finite buffer of size K. We observe the state of the system at the end of each time slot just before the cell in the server (if there is one) leaves the system. Then, the steady state joint probability distribution of the queue length (including the server) and the phase of the D-BMAP at the observed time instants can be obtained by solving the invariant probability vector (denoted $\vec{X} = \{\vec{x}_0, \vec{x}_1, \ldots, \vec{x}_K\}$, where the jth element of the vector $\vec{x}_i$, $0 \le j \le m$, is the steady state joint probability of phase j and a queue length of i) of the following matrix [7]:

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{D}_0 & \boldsymbol{D}_1 & \boldsymbol{D}_2 & \ldots & \boldsymbol{D}_{K-1} & \sum_{i=K}^{\infty} \boldsymbol{D}_i \\ \boldsymbol{D}_0 & \boldsymbol{D}_1 & \boldsymbol{D}_2 & \ldots & \boldsymbol{D}_{K-1} & \sum_{i=K}^{\infty} \boldsymbol{D}_i \\ \boldsymbol{0} & \boldsymbol{D}_0 & \boldsymbol{D}_1 & \ldots & \boldsymbol{D}_{K-2} & \sum_{i=K-1}^{\infty} \boldsymbol{D}_i \\ \cdot & \cdot & \cdot & \ldots & \cdot & \cdot \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \ldots & \boldsymbol{D}_0 & \sum_{i=1}^{\infty} \boldsymbol{D}_i \end{bmatrix} \tag{5.3}$$

Note that, from (5.2), we know that:

$$\sum_{i=k}^{\infty} \boldsymbol{D}_i = \boldsymbol{D} - \sum_{i=0}^{k-1} \boldsymbol{D}_i$$

Thus, we can solve $\vec{X}$ directly using the facts that $\vec{X}\boldsymbol{R} = \vec{X}$ and $\sum_{i=1}^{K} \vec{x}_i \vec{e} = 1$. Once $\vec{X}$ is found, the cell loss probability can be obtained systematically as follows:

$$P_{\text{loss}} = \frac{1}{\rho} \sum_{i=0}^{K} \sum_{j=1}^{\infty} max\{j - (K-i) - 1 + \delta_i, 0\} \vec{x}_i \boldsymbol{D}_j \vec{e} \tag{5.4}$$

where ρ is the *traffic intensity* (defined later); $\vec{e}$ is a column vector of all 1's; and

$$\delta_i \equiv \begin{cases} 1, & \text{if } i = 0 \\ 0, & \text{otherwise} \end{cases}$$

5.3 Traffic models

We model a voice source by a discrete-time ON-OFF process in which the voice source alternates between geometrically distributed ON and OFF periods. Cells are generated with a constant arrival probability (instead of constant interarrival time) during the ON periods and no cells are generated during the OFF periods. In discrete time, this point process is called an *Interrupted Bernoulli Process* (IBP), which can be defined by three parameters: the transition probabilities from ON to OFF (β) and from OFF to ON (α) and the constant arrival probability (ω). To model the superposition of N independent voice sources, we further assume that during any time slot only one voice source can change its state (either from ON to OFF or from OFF to ON). This is a reasonable assumption, since the values of α and β are typically in the order of 10^{-5} (or less). For example: a channel capacity of 44.736 Mbps (standard DS-3 data rate) would set α and β to 1.46×10^{-5} and 2.63×10^{-5} respectively (for an average ON duration of 352 msec and an average OFF period of 650 msec [8]) and the higher the channel capacity is the smaller the values of α and β are. The superposition of N such independent voice sources can be represented by an (N+1)-state D-BMAP (which is the same as Fig. 1.3 except for transition probabilities between states instead of transition rates) where the states represent the number of voice sources in the ON state. The bulk size distribution is *binomial* with parameters k and ω, denoted $\mathcal{B}(k, \omega)$, for phase k, $0 \le k \le N$.

We adopt the discrete-time version of the model originally proposed by Maglaris *et al.* (Fig. 1.7) to model video sources with uniform activity level. This model can be viewed as the superposition of M mini-sources each of which is an IBP with a cell arrival probability of, say, η and transition probabilities of, say, a and b (see [53] for details). Thus, the expected number of cells arriving in a slot from all video sources can be fitted into $M + 1$ equal-distance discrete levels, 0,

η, 2η,..., $M\eta$. As in our voice model, for level i, $0 \le i \le M$, the bulk size distribution is *binomial* with parameters i and η, $\mathcal{B}(i, \eta)$. Transitions are assumed to take place only to adjacent levels where the transition probabilities are obtained by matching the statistical characteristics of the process to that of the video sources.

We assume that cell arrivals from aggregated data sources form a *bulk arrival Bernoulli process* with an arrival probability of μ per time slot, i.e., the data bulk interarrival time is geometrically distributed with the same parameter. We assume that the size of a data bulk is a random variable with an arbitrary probability mass function where $s_k = Pr\{\text{a data bulk is comprised of } k \text{ cells}\}$, $k = 0, 1, 2, \ldots$, and that it is non-trivial, i.e., $s_k > 0$ for some $k > 0$. Note that we have adjusted the bulk size distribution so that the event of no arrival is presented in the form of an arrival with bulk size 0, i.e., $s_0 = 1 - \mu$. This facilitates integration with other traffic types.

5.4 Voice and data integration

Consider an ATM multiplexer loaded with N independent voice sources, which is modeled by the discrete-time *birth-death process* described in section 5.3. Let $\bar{s} = \sum_{k=1}^{\infty} k \cdot s_k$ be the average number of cells in a data bulk; $L = \lfloor (1 - \mu\bar{s}) / \omega \rfloor$ be the maximum number of active voice sources that the multiplexer can support (we assume that $1 < L < N$ exists; otherwise the system load is either too light to be interesting or too heavy to be considered); T_i, $1 \le i \le N$, be the expected time until the superposition arrival process visits state $i - 1$ for the first time starting from state i; and π_k defined by:

$$\pi_k = \binom{N}{k} \left(\frac{\alpha}{\alpha + \beta}\right)^k \left(\frac{\beta}{\alpha + \beta}\right)^{N-k}, \quad 0 \le k \le N \tag{5.5}$$

be the steady state probability that k out of the N voice sources are in the ON-state.

We can draw on the continuous-time results (see Chapter 3) to find that the expected time until the system load drops below the system capacity (once the load exceeds the system capacity, L) to be:

$$T_{N-i} = \frac{1}{N\beta\binom{N-1}{i}} \sum_{k=0}^{i} \binom{N}{k} \left(\frac{\alpha}{\beta}\right)^{i-k}, \quad 0 \le i \le N-1 \tag{5.6}$$

which is upper bounded by:

$$\tilde{T}_{L+1} = \frac{N-L}{L+1} \cdot \frac{1}{\beta} \tag{5.7}$$

Note that, as was pointed out in Chapter 3, $(N-L)/(L+1)$ measures the system load and $1/\beta$ measures how quickly the system load will drop. As we will show by examples, $\tilde{T}_{L+1}$ serves as a very good estimate for how long the system stays in overload.

We approximate the superposed arrival process corresponding to N voice source by a two-state D-BMAP using the following matching procedure:

i. $p_2 = \dfrac{1}{\tilde{T}_{L+1}} = \dfrac{(L+1)\beta}{N-L}$

ii. $q_1(k) \sim \mathcal{B}\left(L, \dfrac{\upsilon_1}{L}\right)$, where $\upsilon_1 = \omega \sum_{k=0}^{L} k\left(\dfrac{\pi_k}{\Pi_u}\right)$ with $\Pi_u = \sum_{k=0}^{L} \pi_k$

iii. $q_2(k) \sim \mathcal{B}\left(N, \dfrac{\upsilon_2}{N}\right)$, where $\upsilon_2 = \omega \sum_{k=L+1}^{N} k\left(\dfrac{\pi_k}{\Pi_o}\right)$ with $\Pi_o = \sum_{k=L+1}^{N} \pi_k$

iv. $p_1 = p_2 \dfrac{\omega\phi - \upsilon_1}{\upsilon_2 - \omega\phi}$, where $\phi = N\left(\dfrac{\alpha}{\alpha+\beta}\right)$ is the expected number of active calls.

Note that the first step estimates the transition probability from state-II to state-I of the two-state D-BMAP using $\tilde{T}_{L+1}$, the estimated time that the arrival process stays in overload situation. Step *ii*) is set up in such a way that the average number of cell arrivals in underload states is equal to the expected number of cell arrivals in state-I. The way step *iii)* is set up for overload states is similar to that of step *ii*) for underload states. While the choice of binomial distribution is due to its simplicity; the choice of its parameters has little impact on the resulting model as long as the average number of cell arrivals matches that of the exact analytical model (as we will show later by examples). The last step matches the overall average cell arrival probability of the two-state D-BMAP, $(\upsilon_1 p_2 + \upsilon_2 p_1)/(p_2 + p_1)$, with that of the original arrival process, $\omega\phi$.

Let us continue our discussion by pointing out that the integration of the voice sources modeled by the above two-state D-BMAP and the data sources modeled by a *batch arrival Bernoulli process* is clearly another two-state D-BMAP with:

$$p'_i = p_i, i = 1, 2 \tag{5.8}$$

$$q'_i(k) = (s \otimes q_i)(k), \; i = 1, 2, \; k = 0, 1, 2, \ldots \tag{5.9}$$

where $\otimes$ denotes convolution.

Applying (5.8) and (5.9) to (5.1), we have the following for the two-state D-BMAP representing the integration of voice and data sources:

$$\boldsymbol{D}_k = \begin{bmatrix} q'_1(k)(1-p_1) & q'_1(k)p_1 \\ q'_2(k)p_2 & q'_2(k)(1-p_2) \end{bmatrix}, \quad k = 0, 1, 2, \ldots \tag{5.10}$$

Also, the transition probability matrix for the phase process is given by:

$$\boldsymbol{D} \equiv \sum_{k=0}^{\infty} \boldsymbol{D}_k = \begin{bmatrix} 1-p_1 & p_1 \\ p_2 & 1-p_2 \end{bmatrix} \tag{5.11}$$

5.5 Video, voice and data integration

To include video traffic into the model, we combine the models for voice and video sources which results in a discrete-time *two-dimensional Markov chain* similar to Fig. 4.1 where the states represent the level of expected number of arrivals for all video sources and the number of active voice sources. Again, we have assumed that a state change for voice sources and a level change for video sources cannot occur in a single slot. (This is reasonable due to the fact that both α, β and a, b are very small.) As in continuous-time case, the steady-state distribution of this two-dimensional Markov chain can be solved and found to be:

$$\pi_{ij} = \binom{N}{j}\binom{M}{i}\left(\frac{q}{a+b}\right)^i\left(\frac{b}{a+b}\right)^{M-i}\left(\frac{\alpha}{\alpha+\beta}\right)^j\left(\frac{\beta}{\alpha+\beta}\right)^{N-j} \tag{5.12}$$

for $0 \le i \le M$ and $0 \le j \le N$.

It can be verified that, in state (i, j), the bulk size distribution is a multinomial (more precisely, a trinomial) distribution with the following two parameter sets: i, η and j, ω (recall that η is the step size of the levels of expected number of arrivals from all video sources and ω is the probability of arrival of a single voice source). We say a state (i, j) is overloaded if $(\eta i + \omega j) > 1 - \mu\bar{s}$; otherwise, we say that it is underloaded. As in continuous-time case, let S_u, S_o be the sets and θ_u, θ_o be the numbers of the underload states and overload states respectively; σ_i, $0 \le i \le M$ be the number of voice sources which the system can support given that the video sources are currently in level i; and $F = max_i\{\sigma_i + i\}$ be the largest possible number of active mini-sources among underload states. Note that $\sigma_i = \lfloor (1 - \mu\bar{s} - \eta i)/\omega \rfloor$ and is constrained to $-1 \le \sigma_i \le N$ (where $\sigma_i = -1$ counts the situation where the video sources in cell arrival rate level i alone overload the system).

Following a similar approach to that used for voice, we propose to use the following $\tilde{T}$ as an

estimate for the time that the process stays in overload each visit:

$$\tilde{T} = \frac{\theta_o}{\theta_u} \cdot \frac{1}{\beta + b} = \left(\sum_{i=0}^{M} (N - \sigma_i) / \sum_{i=0}^{M} (\sigma_i + 1) \right) \frac{1}{\beta + b} \tag{5.13}$$

The matching procedure for the discrete-time two-state D-BMAP approximation is then given by:

i. $p_2 = \frac{1}{\tilde{T}} = \left(\sum_{i=0}^{M} (\sigma_i + 1) / \sum_{i=0}^{M} (N - \sigma_i) \right) (\beta + b)$

ii. $q_1(k) \sim \mathcal{B}\left(F, \frac{\upsilon_1}{F}\right)$, where $\upsilon_1 = \sum_{(i,j) \in S_u} (\eta i + \omega j) \left(\frac{\pi_{ij}}{\Pi_u}\right)$ with $\Pi_u = \sum_{(x,y) \in S_u} \pi_{xy}$

iii. $q_2(k) \sim \mathcal{B}\left(N + M, \frac{\upsilon_2}{N + M}\right)$, where $\upsilon_2 = \sum_{(i,j) \in S_o} (\eta i + \omega j) \left(\frac{\pi_{ij}}{\Pi_o}\right)$ with $\Pi_o = \sum_{(x,y) \in S_o} \pi_{xy}$

iv. $p_1 = p_2 \frac{(\omega\phi_1 + \eta\phi_2) - \upsilon_1}{\upsilon_2 - (\omega\phi_1 + \eta\phi_2)}$, where $\phi_1 = \frac{N\alpha}{\alpha + \beta}$ and $\phi_2 = \frac{Ma}{a + b}$

As before, the first term of $\tilde{T}$ provides an indication as to how heavy the system load is and the second term measures how quickly the current system load will drop. Note that, it is the mean of a random variable which is the minimum of two geometrically distributed random variables, i.e., the time for an active voice source and video mini-source to switch state, with a parameter of β and b respectively. The setup for bulk size distribution is similar to that used in the integration of voice and data sources. The last step of the procedure matches the average cell arrival probability of the two-state D-BMAP, $(\upsilon_1 p_2 + \upsilon_2 p_1) / (p_2 + p_1)$, with that of the actual cell arrival process, $\omega\phi_1 + \eta\phi_2$. Once p_1, p_2, $q_1(k)$ and $q_2(k)$ have been determined, the integration of video, voice and data sources can be done as before using (5.8) and (5.9); and $\boldsymbol{D}_k$'s can be obtained using (5.10).

5.6 Numerical results for voice and data integration

Our reference model for voice sources is characterized by a cell arrival rate of $1/6$ cells/msec (assume 64 Kbps PCM coding with speech activity detector and a standard 48-octet payload size per cell) and average ON and OFF durations of 352 msec and 650 msec respectively (as concluded by [8]). Aggregated data traffic is assumed to have a data bulk arrival probability of $20N_d/3$ bulks/sec, where N_d is the number of data calls. We assume that the data bulk size is geometrically distributed with an average of 5 cells/bulk (note that this has to be adjusted for the probability of bulk size being 0 before it can be applied to our model). We assume that the system has a total of 20 voice calls and 30 data calls.

In Fig. 5.1 and Fig. 5.2, we show the impact of different choices of $q_i(k)$, the distribution of bulk size that we use in the approximate model, on the queue length distribution and the corresponding survivor function with a buffer size of 200. In these figures, three different choices of $q_i(k)$'s have been used: (a) a *constant* bulk size, (b) a binomial distributed bulk size with the *maximum* number of active sources being one of the parameters (as in our matching procedure), and (c) a binomial distributed bulk size with the *average* number of active sources being one of the parameters. All of them, however, have the same expected bulk size of υ_1 for state-I and υ_2 for state-II. As pointed out before, the results show that the choice of $q_i(k)$ has very little impact on the performance of the approximation (the curves for (b) and (c) above overlap each other in the figures).

Fig. 5.3 and Fig. 5.4 assume the buffer size is 100 and show the queue length distribution and the corresponding survivor function for the exact analytical model and the approximation with different traffic intensities (defined to be $\rho = \mu\bar{s} + \omega\phi_1$). In Fig. 5.5, average buffer occupancy for different buffer sizes (in number of cells) are compared. Finally, the comparisons of cell loss probabilities for different buffer sizes (in number of cells) are presented in Fig. 5.6. Note that in Fig. 5.5 and Fig. 5.6, we show results for buffer sizes up to only 100 for the exact analytical model due to the complexity of the computation. As the results show, the relative error in predicting the average buffer occupancy is within 11% of its actual value and the asymptotic slope is about the same. On the prediction of cell loss probability, the relative error ranges from 11% for high system load ($\rho = 0.9$) to as much as 50% for medium system load ($\rho = 0.6$) and intermediate buffer sizes. As we consider larger (more realistic) buffer sizes, the agreement of the approximation with the exact model improves. This is readily apparent for $\rho = 0.7$ through $\rho = 0.9$ and we predict that it is also true for lower utilizations. We will prove this point by comparing results from the approximation with simulation results later in this chapter when we present an example which includes video, voice and data sources.

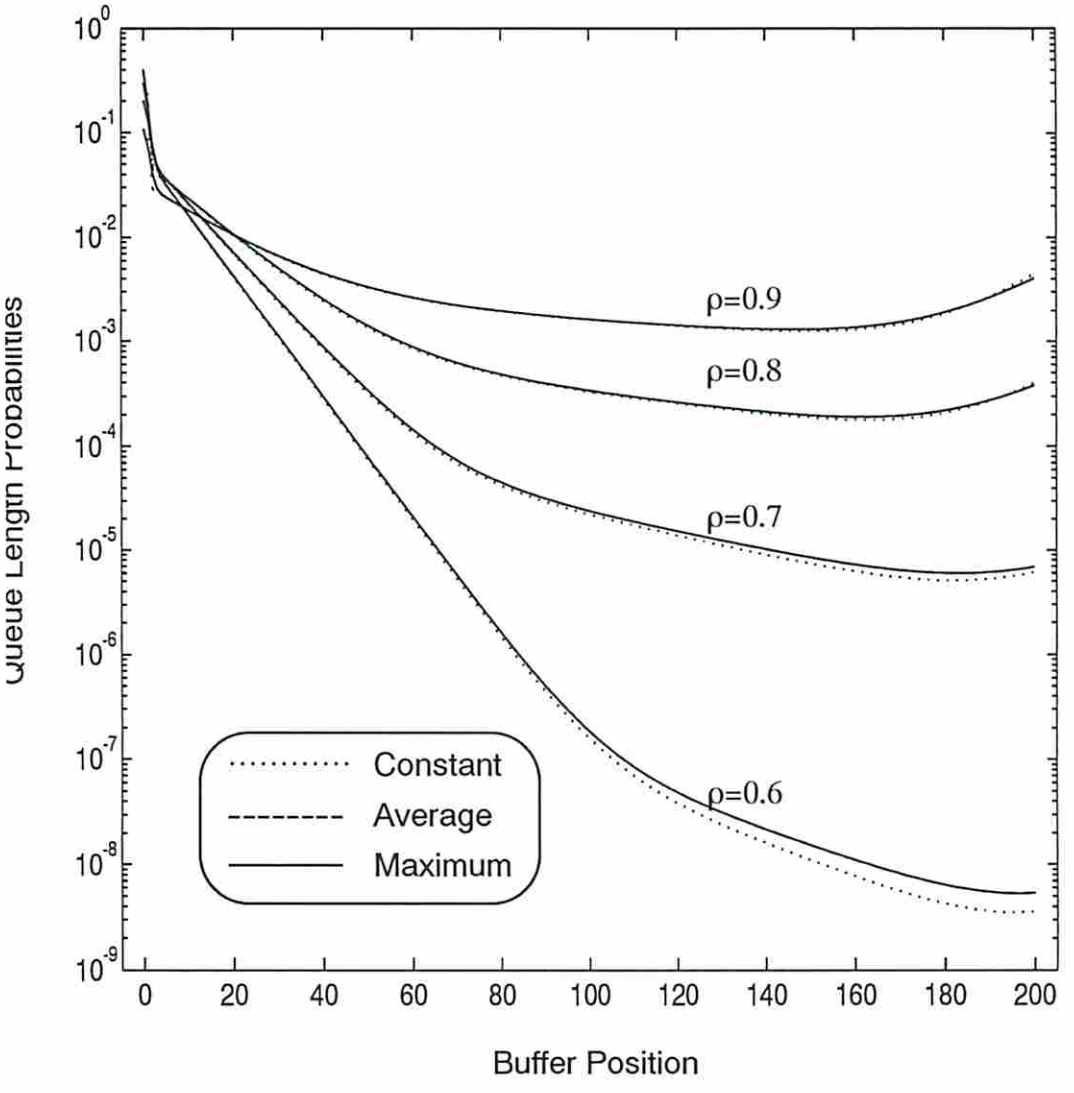

Fig. 5.1 Queue length distribution for different choices of $q_i(k)$.

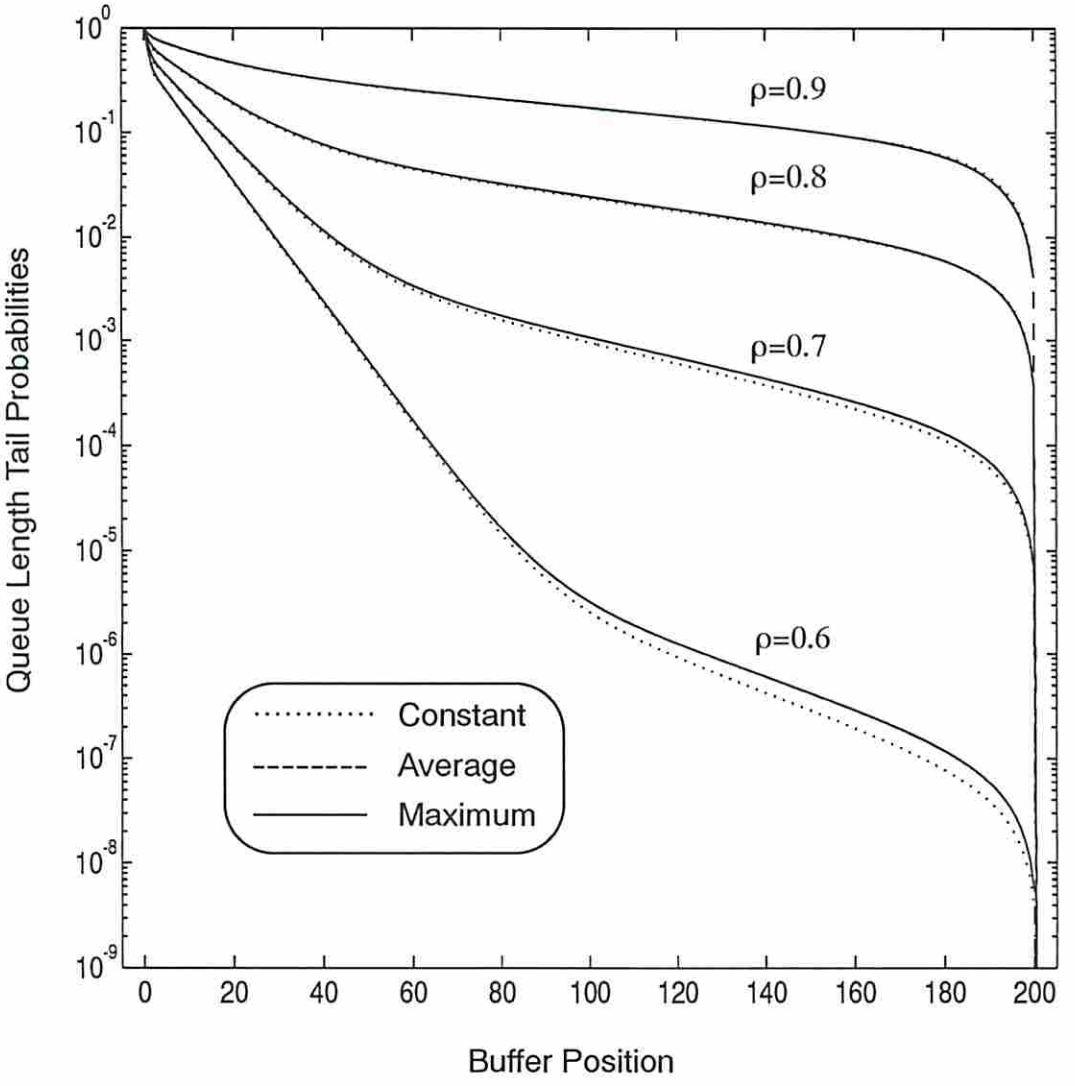

Fig. 5.2 Survivor function for different choices of $q_i(k)$.

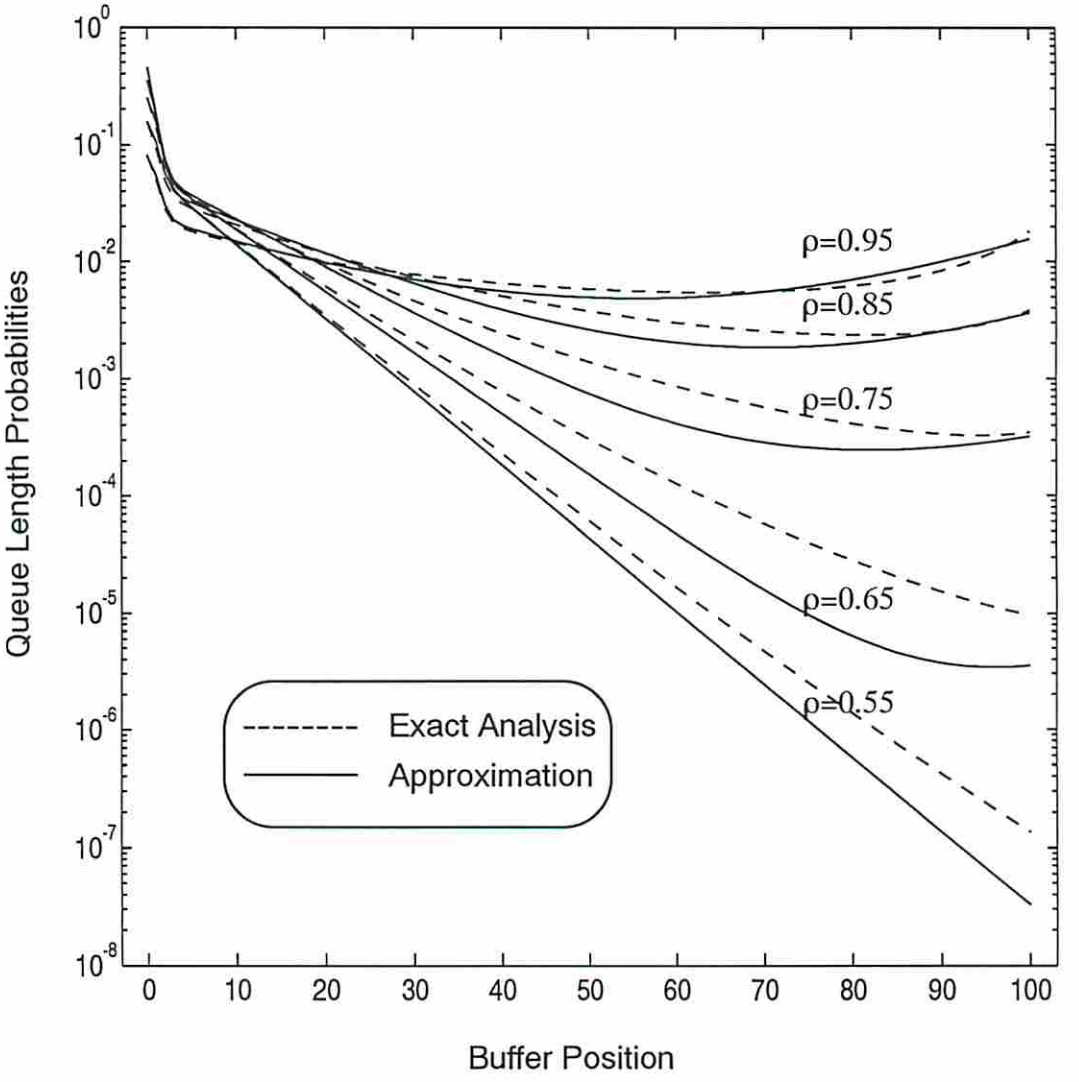

Fig. 5.3 Queue length distribution for 20 voice calls and 30 data calls.

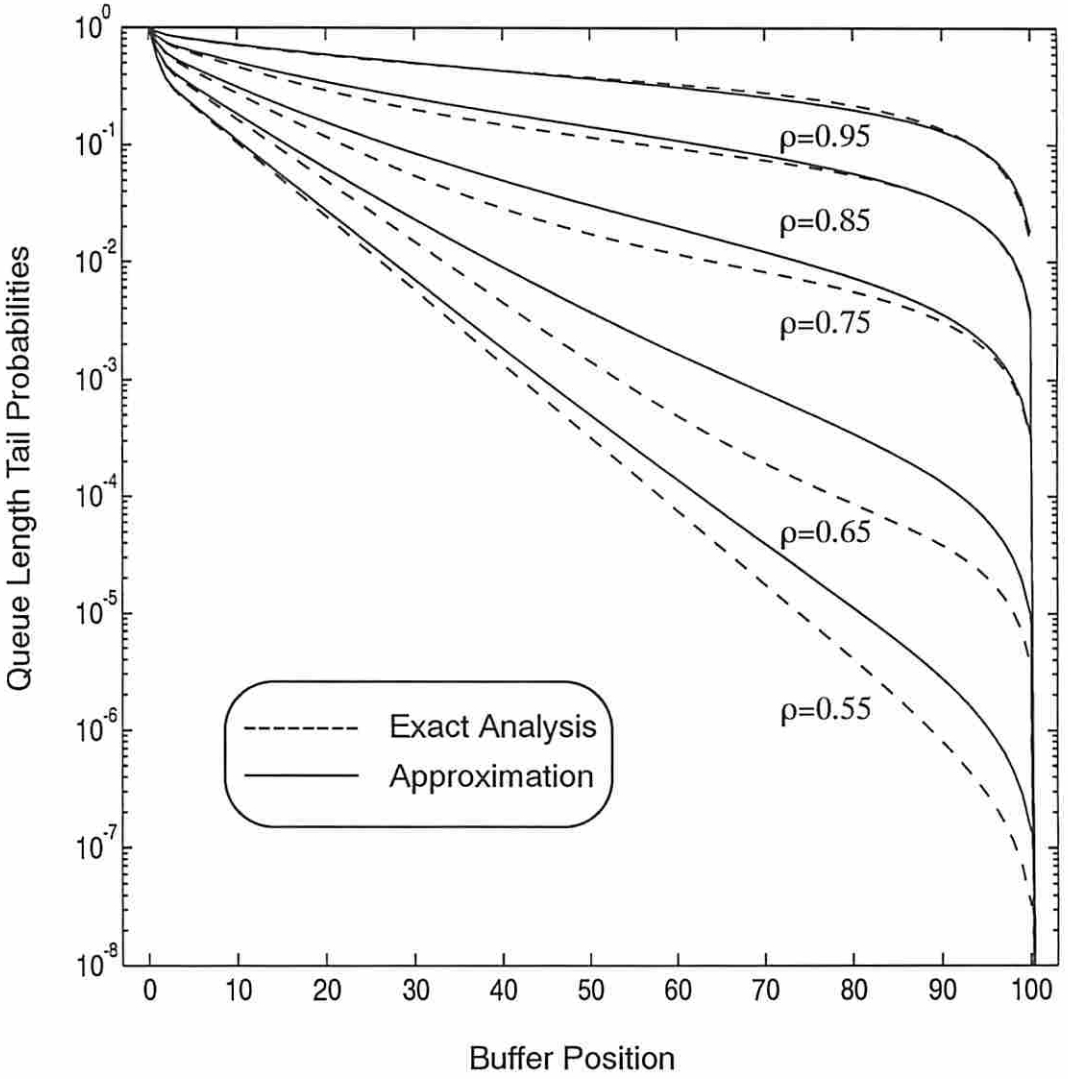

Fig. 5.4 Survivor function for 20 voice calls and 30 data calls.

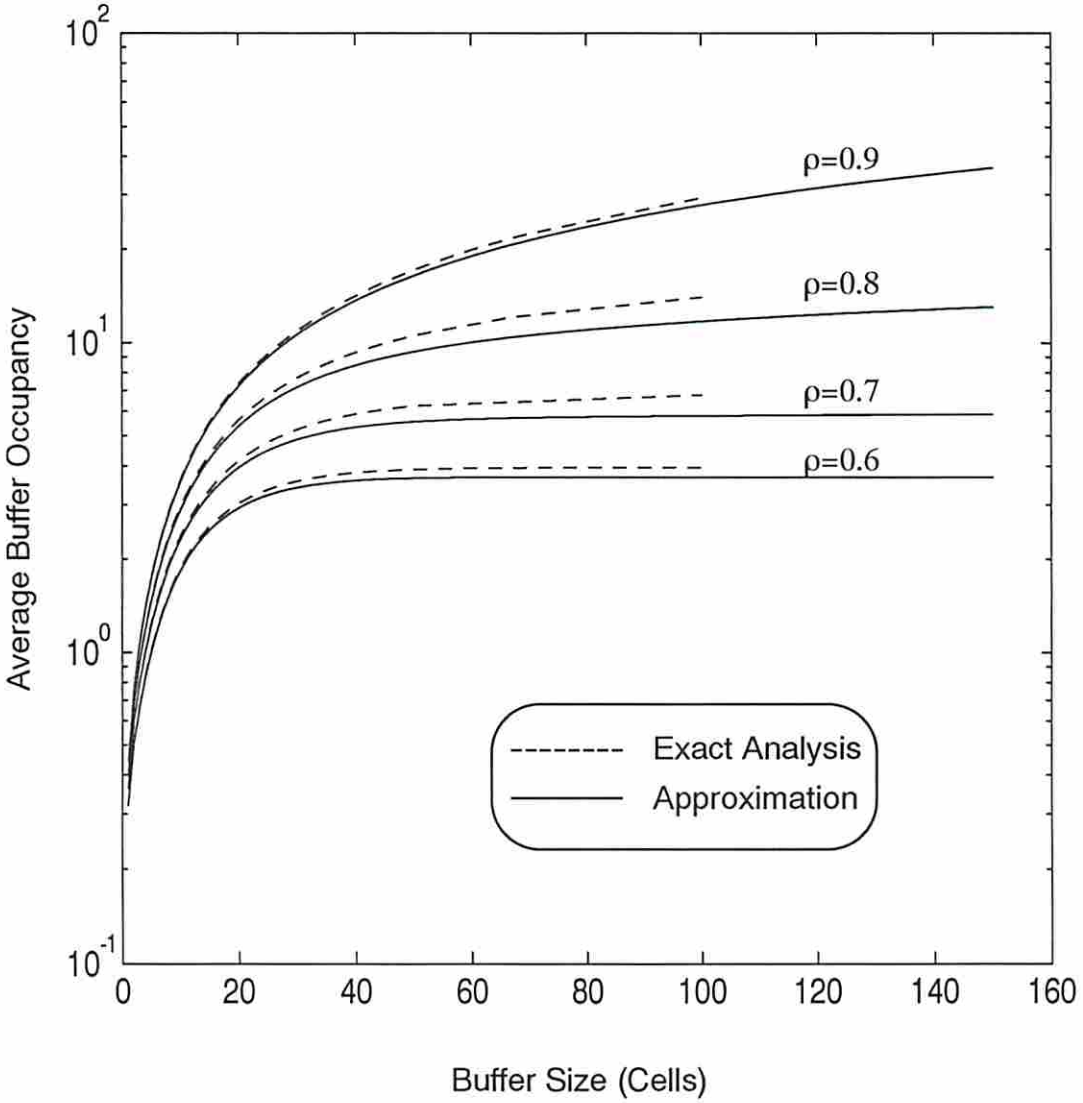

Fig. 5.5 Average buffer occupancy versus different buffer sizes for voice and data integration.

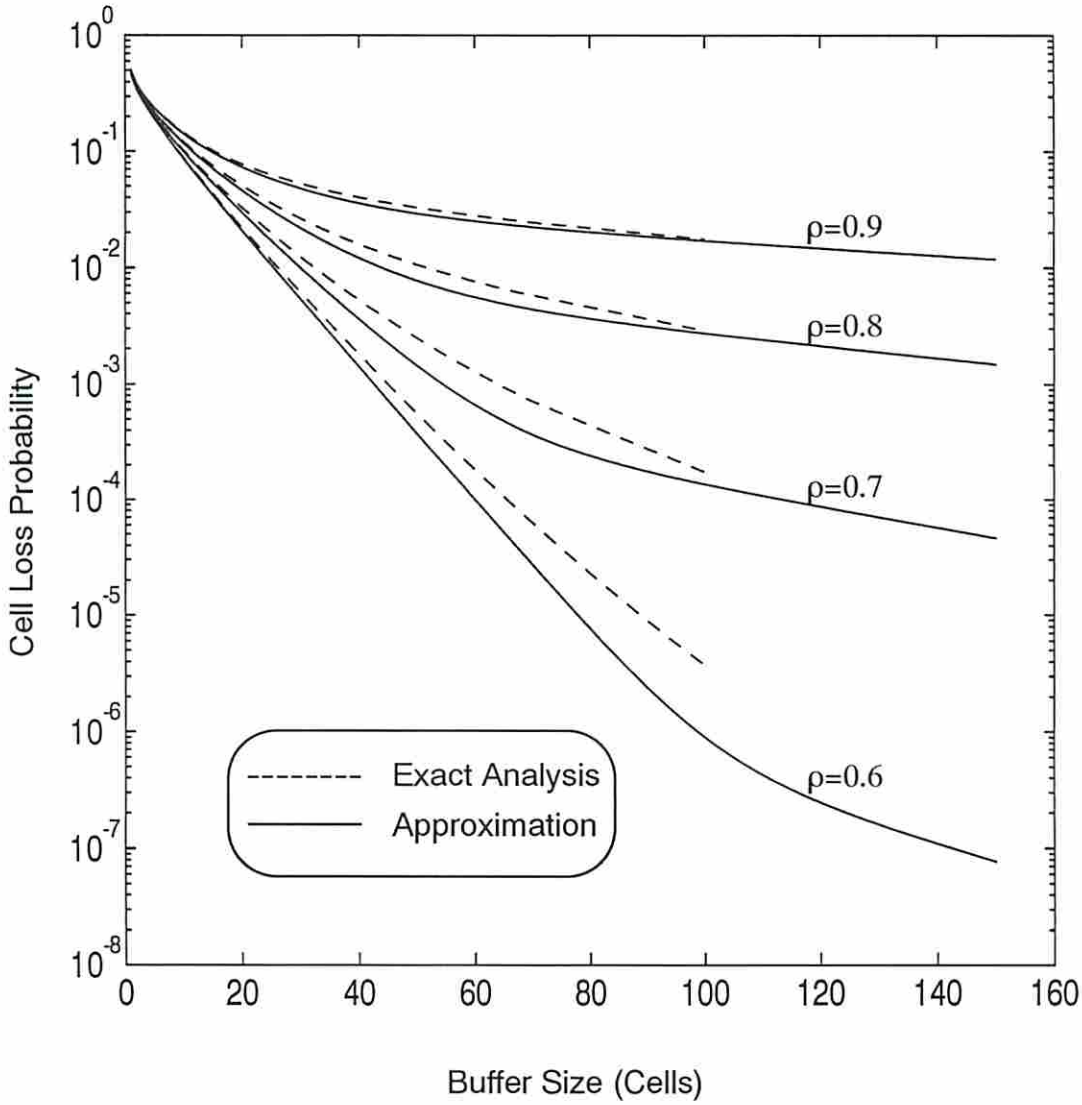

Fig. 5.6 Cell loss probability versus different buffer sizes for voice and data integration.

5.7 Numerical examples for video, voice and data integration

We use the same voice and data source characteristics as in the previous examples and use the same set of parameters for the video sources as used in Chapter 4, i.e., video sources are characterized by: an average bit rate of 3.9 Mbps, a peak bit rate of 10.58 Mbps, a standard deviation of the bit rate of 1.73 Mbps and a parameter for the autocorrelation function of 3.9. The total number of levels for video sources is assumed to be 10 times the number of video sources (as suggested by [53]). We consider a system with 1 video source (otherwise, the size of the exact analytical model would be prohibitively large to be solved using a workstation), 10 voice sources and 50 data sources.

In Fig. 5.7 and Fig. 5.8, we plot the queue length distribution and the corresponding survivor function for the system with a buffer size of 23. Note that in Fig. 5.7, only 2 sets of curves, $\rho = 0.9$ and $\rho = 0.6$, are presented due to their closeness to each other. The average buffer occupancy and the cell loss probability for different buffer sizes are presented in Fig. 5.9 and Fig. 5.10 respectively. In these examples, the exact analytical model can be solved for only a limited buffer size. This is due to the large state space involved in the model. In Fig. 5.7 and Fig. 5.8, for example, the largest buffer size our *SPARC-600*[†] can handle is 23, which has a state space of 2873×2873 and if programmed in *MATLAB*[‡] takes about 300 Mbytes of memory. On the other hand, the approximate model, the state space of which is independent of the number of traffic sources in the system, can be solved for a much larger buffer size (roughly up to a buffer size of well above 1000 running on the same workstation). In order to verify the accuracy of the approximation with an interesting buffer size, we have also include some simulation results in Fig. 5.9 and Fig. 5.10. In these figures, the solid line (representing the analytical results from the exact analysis) stops at a buffer size of 23 due to the reason stated above. Results for larger buffer sizes are presented using simulation results (dotted line). For the average buffer occupancy, a relative error from 7% for high system load to 24% for medium system load is recorded (again, the asymptotic slope matches). For the cell loss probability, even though a relative error of 21% is recorded for ρ=0.6 with small buffer sizes, the results are far more impressive for larger buffer sizes (as confirmed by simulation results).

In order to verify the accuracy of the proposed approximation for more realistic situations, in Fig. 5.11 and Fig. 5.12, we present numerical results for a system with a large (fixed) number of sources under different system loads (10, 1550, and 1380 of video, voice, and data calls respectively are assumed for a channel capacity ranging from 100 Mbps up to 140 Mbps). Similarly, in Fig. 5.13 and Fig. 5.14, we assume a fixed OC-3 channel data rate (155.52 Mbps) with traffic loads from video, voice, and data sources roughly to be 40%, 40%, and 20% respectively. The results show very good agreement in predicting the average buffer size. Due to the large state space (in the order of 10^5) and low loss probability (in the range of 10^{-3} to 10^{-12}) involved in these examples, however, our simulation cannot create enough events to complete the curves for the loss probability in some cases (for example, the curves for ρ=0.75 and 0.8 in Fig. 5.12 and the curves for 12 and 13 video sources in Fig. 5.14). However, from the cases where the loss proba-

[†] SPARC is a trademark of SPARC International, Inc.

[‡] MATLAB is a registered trademark of The MathWorks, Inc.

bility curves are completely drawn and from Fig. 5.10, we expect the agreement on predicting loss probability in these cases to be very good for larger buffer sizes. Another observation which can be made from Fig. 5.10 and Fig. 5.12 is that with a larger number of sources the statistical multiplexing gains become much more significant, i.e., with the same system load, a larger traffic source population creates a significant lower loss probability (see curves for ρ=0.8 on Fig. 5.10 and Fig. 5.12, for example). The potential implication on this observation is that using a fast packet switching technique such as ATM technique on high-speed networks enables the networks to support traffic with an aggregate peak date rate that is much higher than the system capacity. For example, in Fig. 5.10 with ρ=0.8, the aggregate peak data rate is 184% of the system capacity; while in Fig. 5.12 with the same system load, the aggregate peak data rate is 169% of the system capacity. With the aggregate peak data rate to system capacity ratio roughly the same, the later case provides a much lower loss probability (and actually a smaller average buffer size as well). For the same buffer size of, say, 200, the loss probability of the later case is roughly 5 orders of magnitude lower than that of the first case.

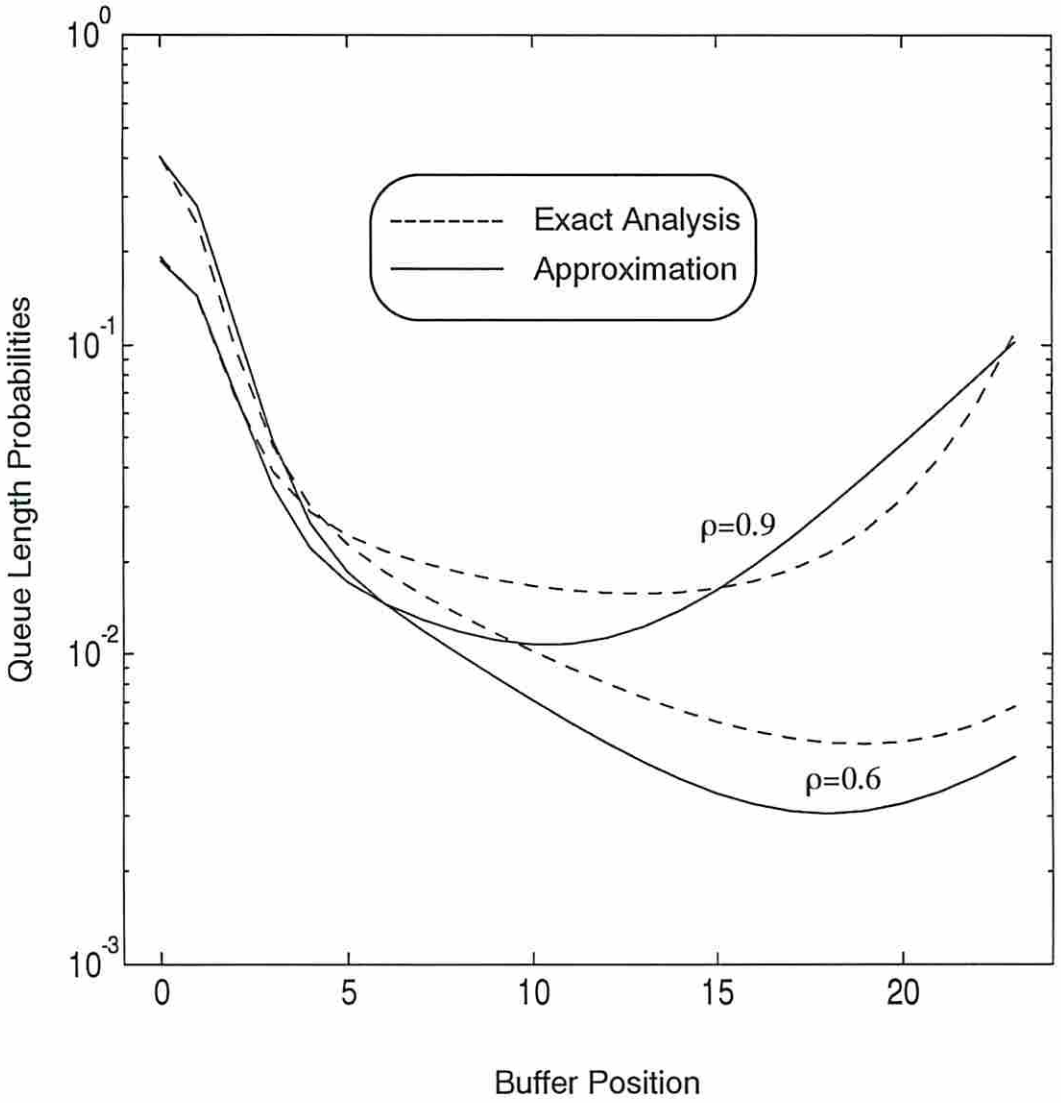

Fig. 5.7 Queue length distribution for 1 video, 10 voice and 50 data calls.

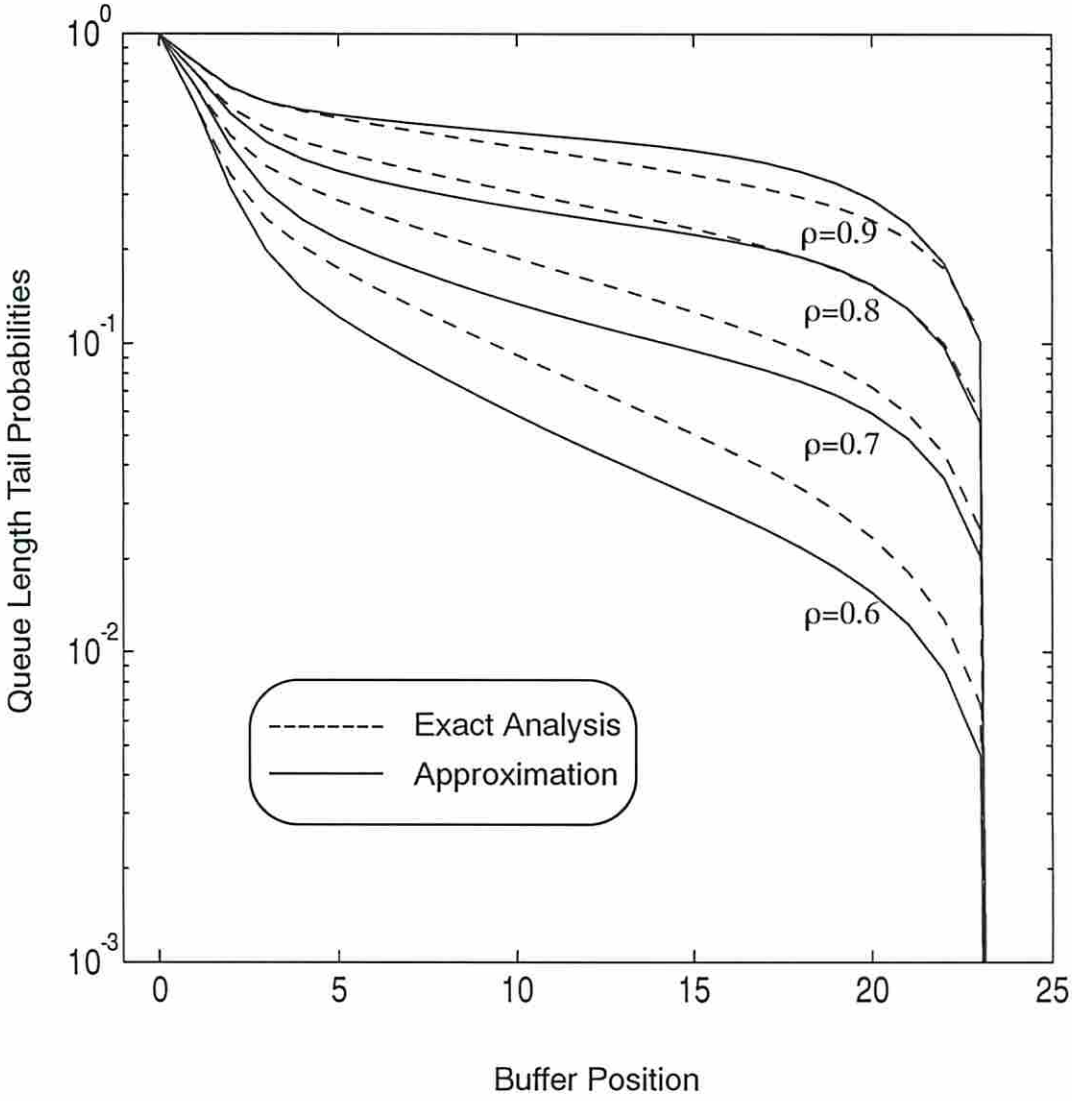

Fig. 5.8 Survivor function for 1 video, 10 voice and 50 data calls.

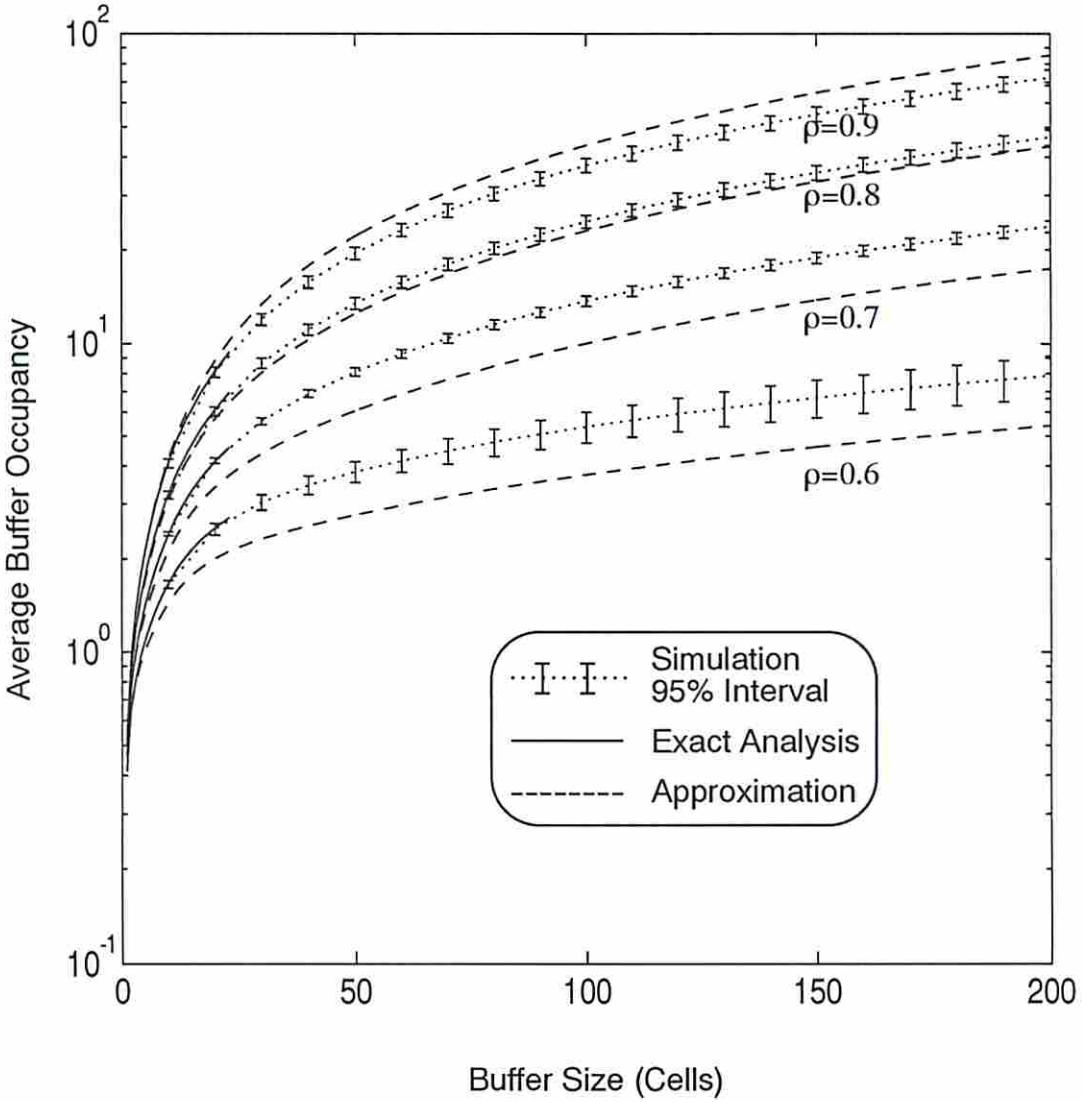

Fig. 5.9 Average buffer occupancy versus different buffer sizes for 1 video, 10 voice and 50 data calls.

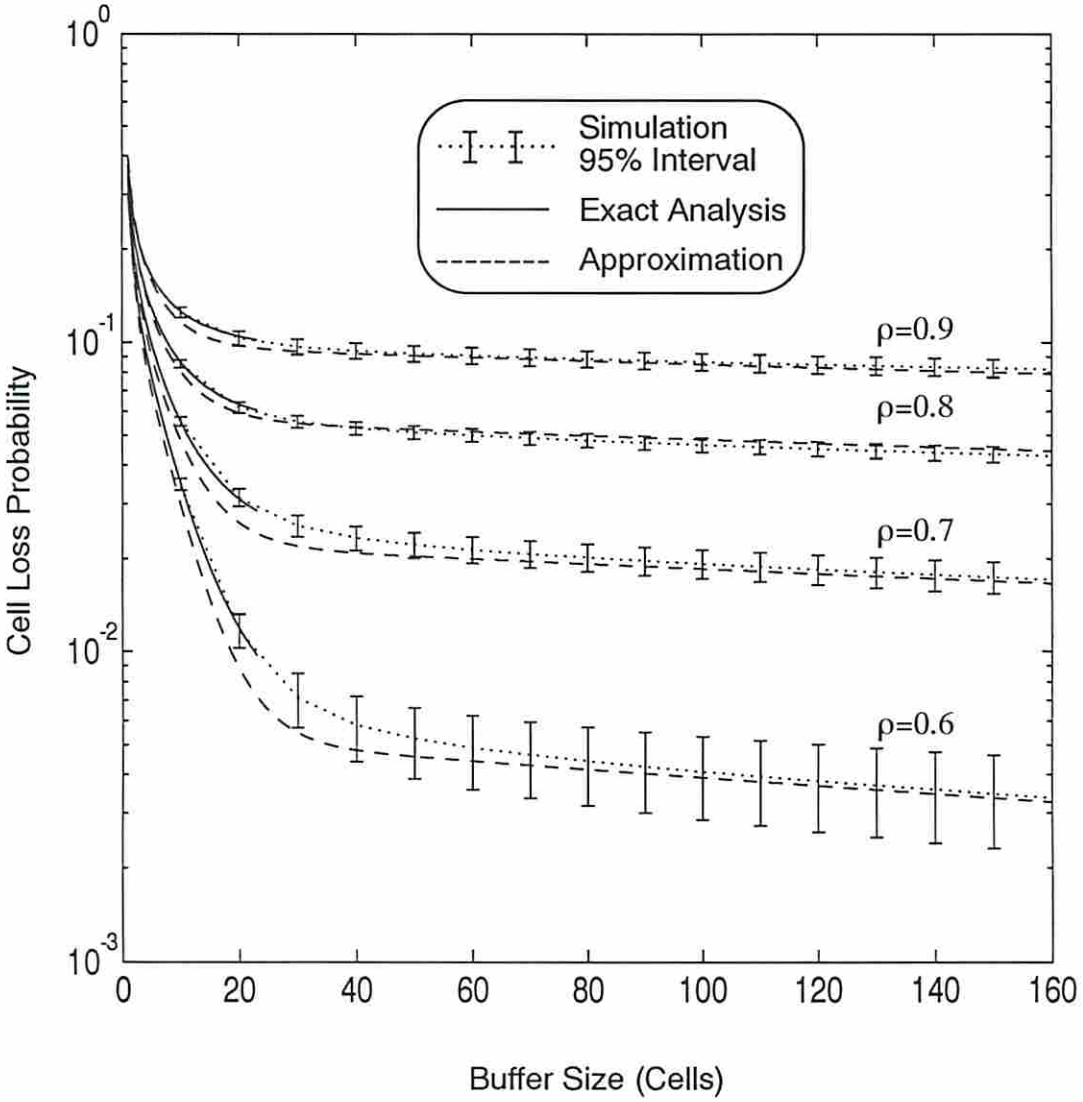

Fig. 5.10 Cell loss probability versus different buffer sizes for 1 video, 10 voice and 50 data calls.

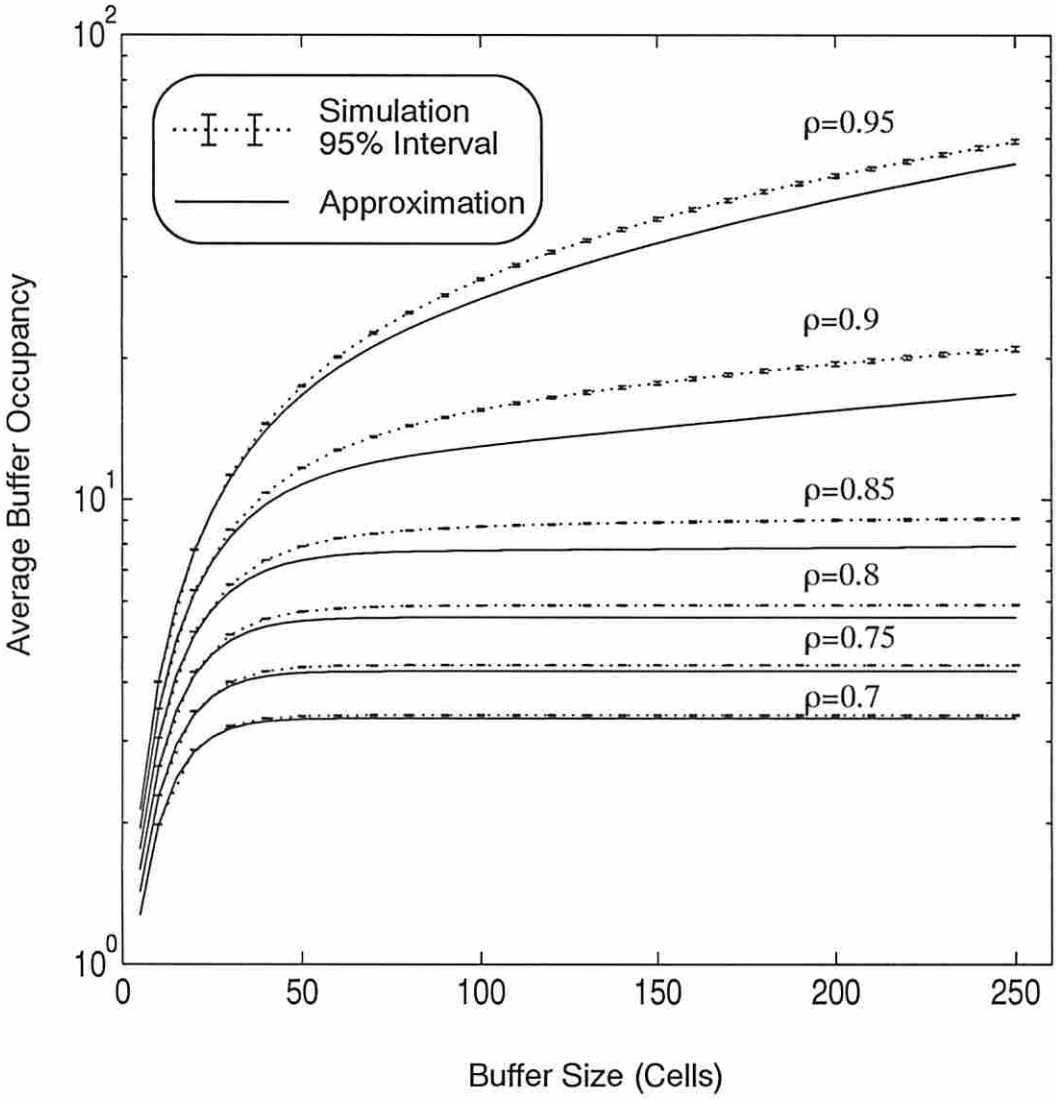

Fig. 5.11 Average buffer occupancy versus different buffer sizes for 10 video, 1550 voice and 1380 data calls.

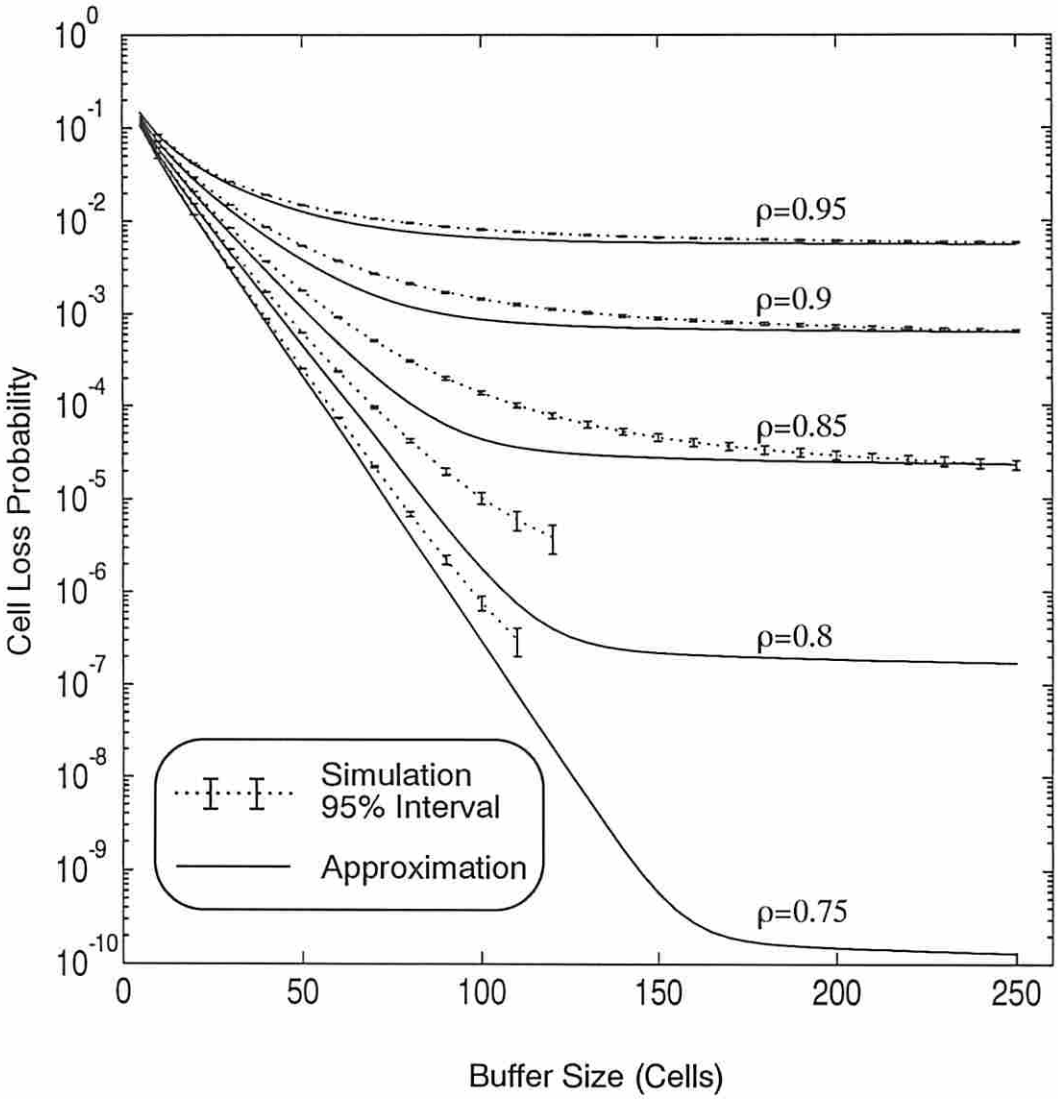

Fig. 5.12 Cell loss probability versus different buffer sizes for 10 video, 1550 voice and 1380 data calls.

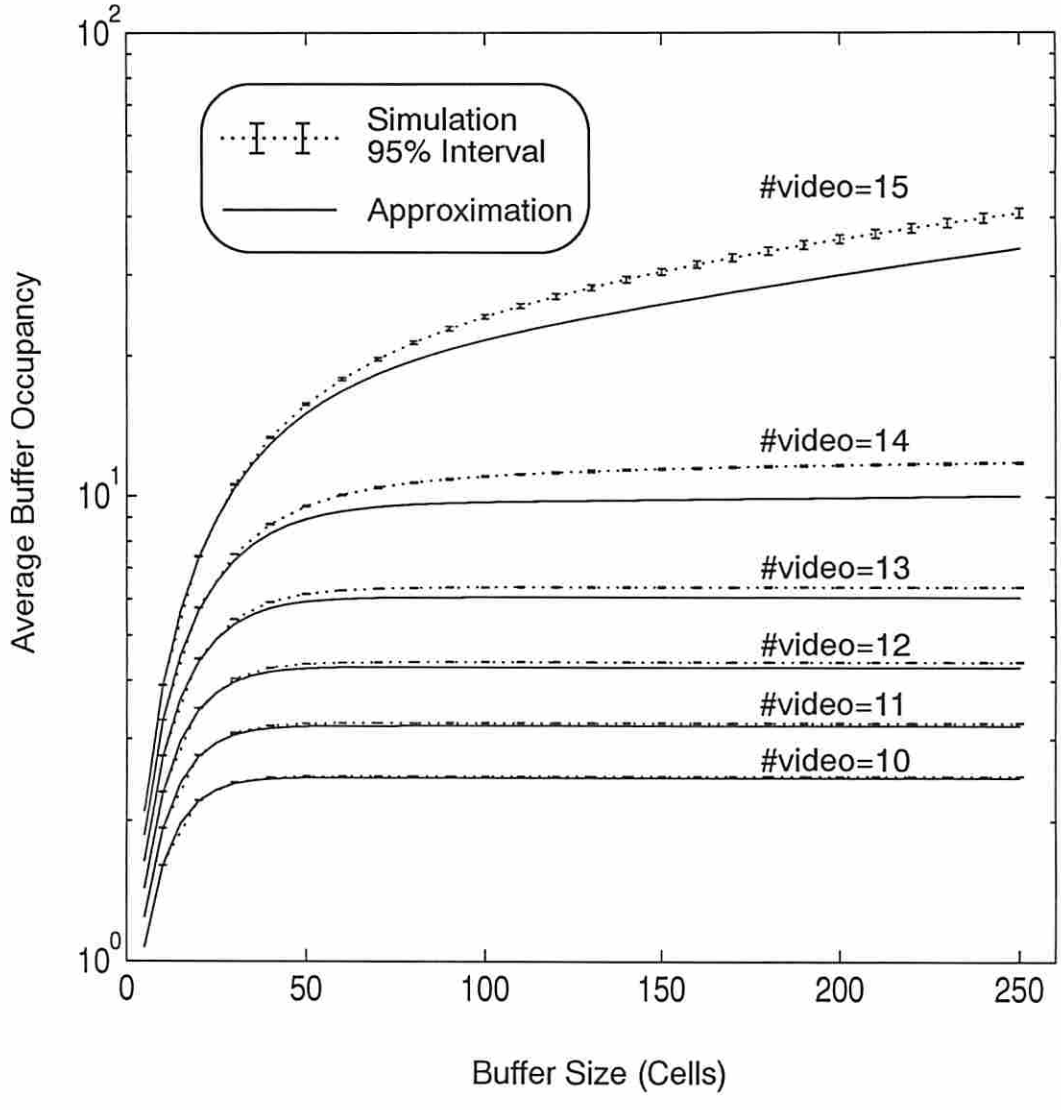

Fig. 5.13 Average buffer occupancy versus different buffer sizes for a standard OC-3 rate, 155.52 Mbps.

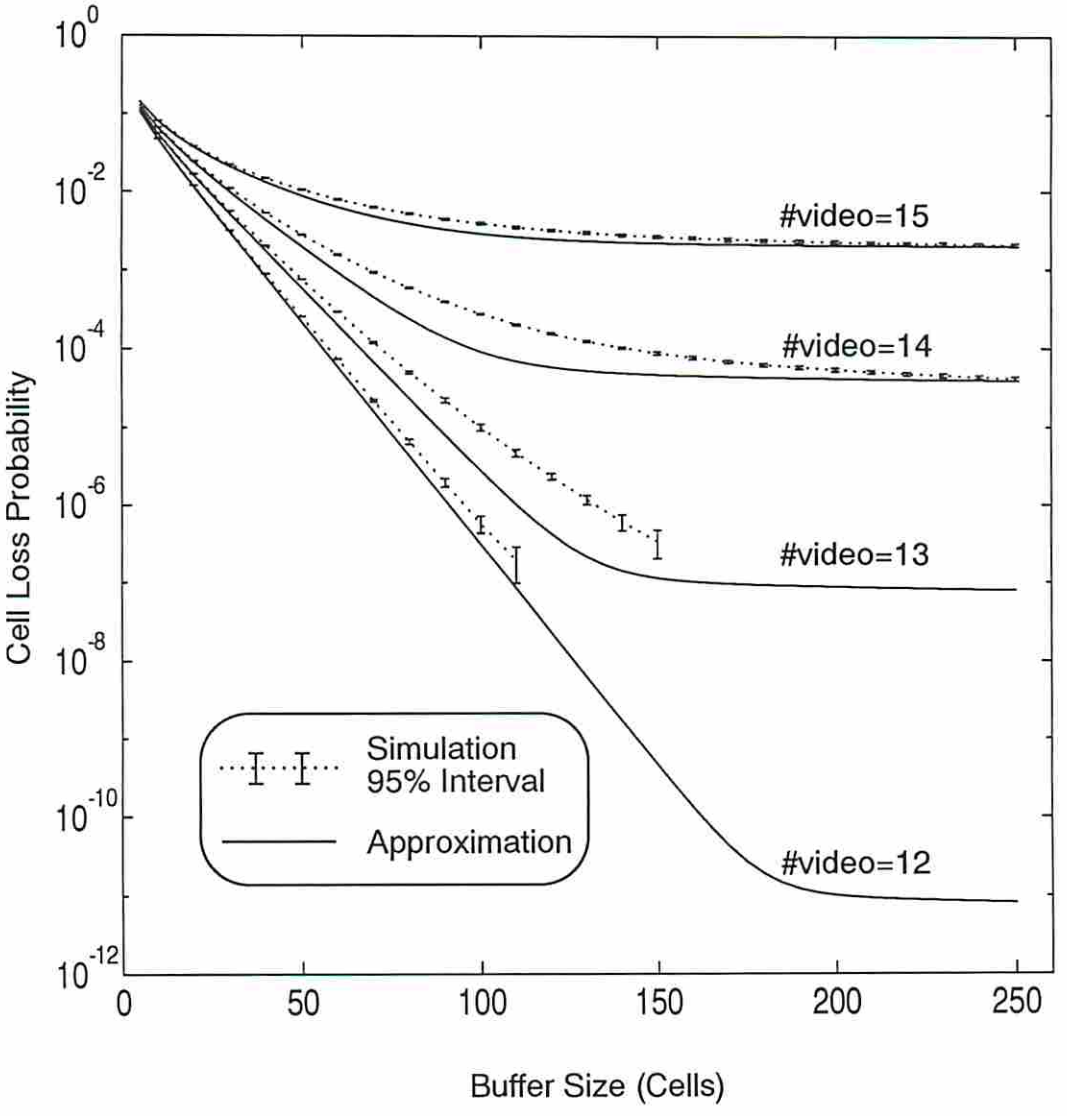

Fig. 5.14 Cell loss probability versus different buffer sizes for a standard OC-3 rate, 155.52 Mbps.

5.8 Application to traffic control

Because of the fast computation of the proposed approximation, one of its potential applications is real-time traffic management such as admission control. Even though developing the details of such a *Call Admission Control* (CAC) algorithm is not within the scope of this research, the basic idea can be described as follows: given a specific number of each type of traffic, the corresponding performance figures (the cell loss probability, for example) can be calculated rather quickly. Depending on the QOS requirement of the connections (new connection as well as existing connections), a decision can then be made to either accept or reject another connection request. We could even adapt the model to measured traffic characteristics of the current connections.

In the following examples, we assume a fixed buffer size of 200 and a fixed channel capacity of 44.736 Mbps, i.e., standard DS-3 rate, and show (in Fig. 5.15) the trade-off between the number of voice sources and video sources that the system can support given a fixed number of data sources (500 data sources in this case) in the background. Similarly, Fig. 5.16 shows the trade-off between the number of data sources and video sources that the system can support given a certain number of voice sources (600 voice sources for this example) in the background. Using these figures, we can easily find, given a specific cell loss probability requirement, the maximum number of voice sources, for example, that the system can support with a certain number of video sources.

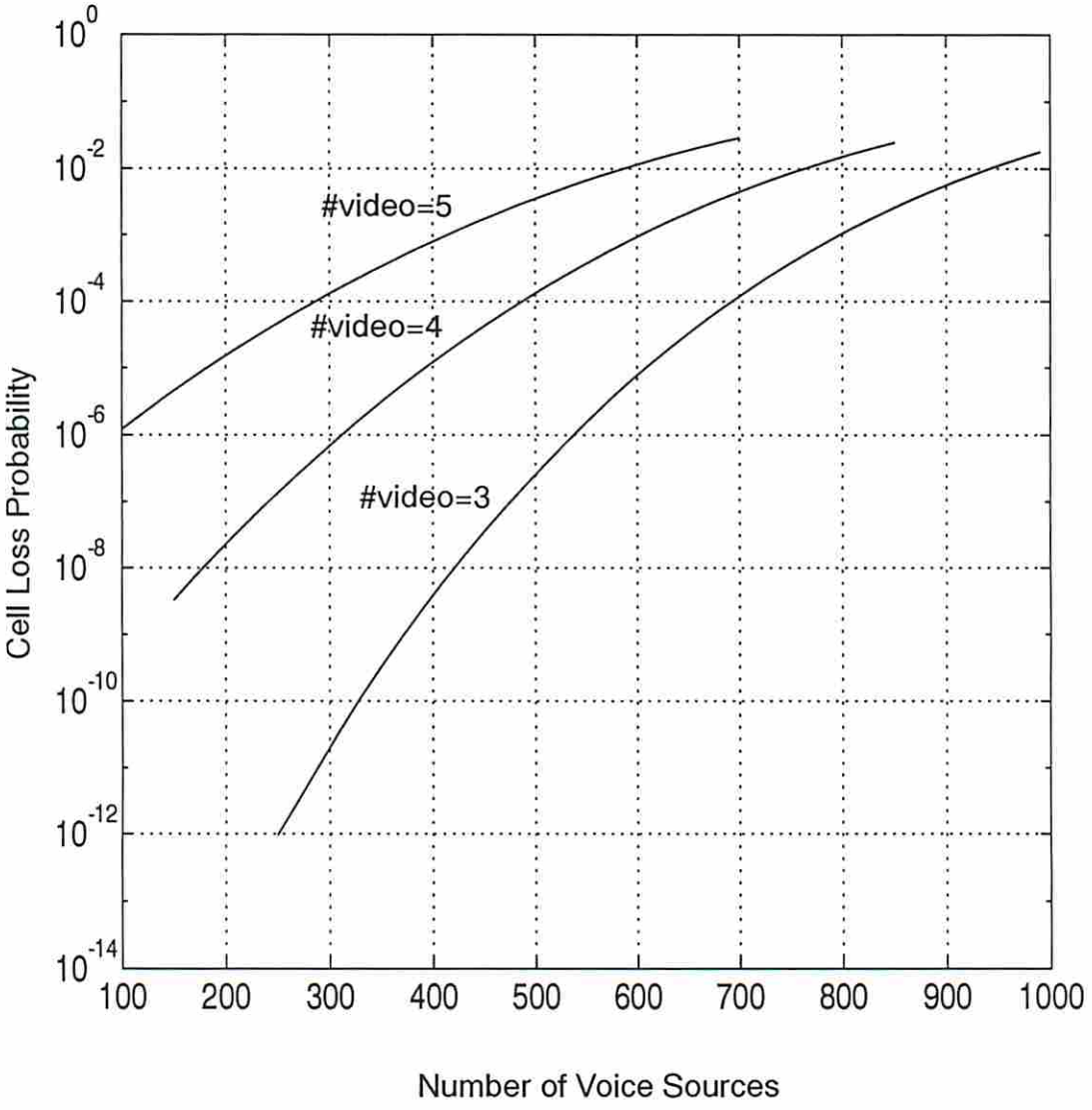

Fig. 5.15 Cell loss probability for different number of video calls with 500 data calls as a function of the number of voice sources.

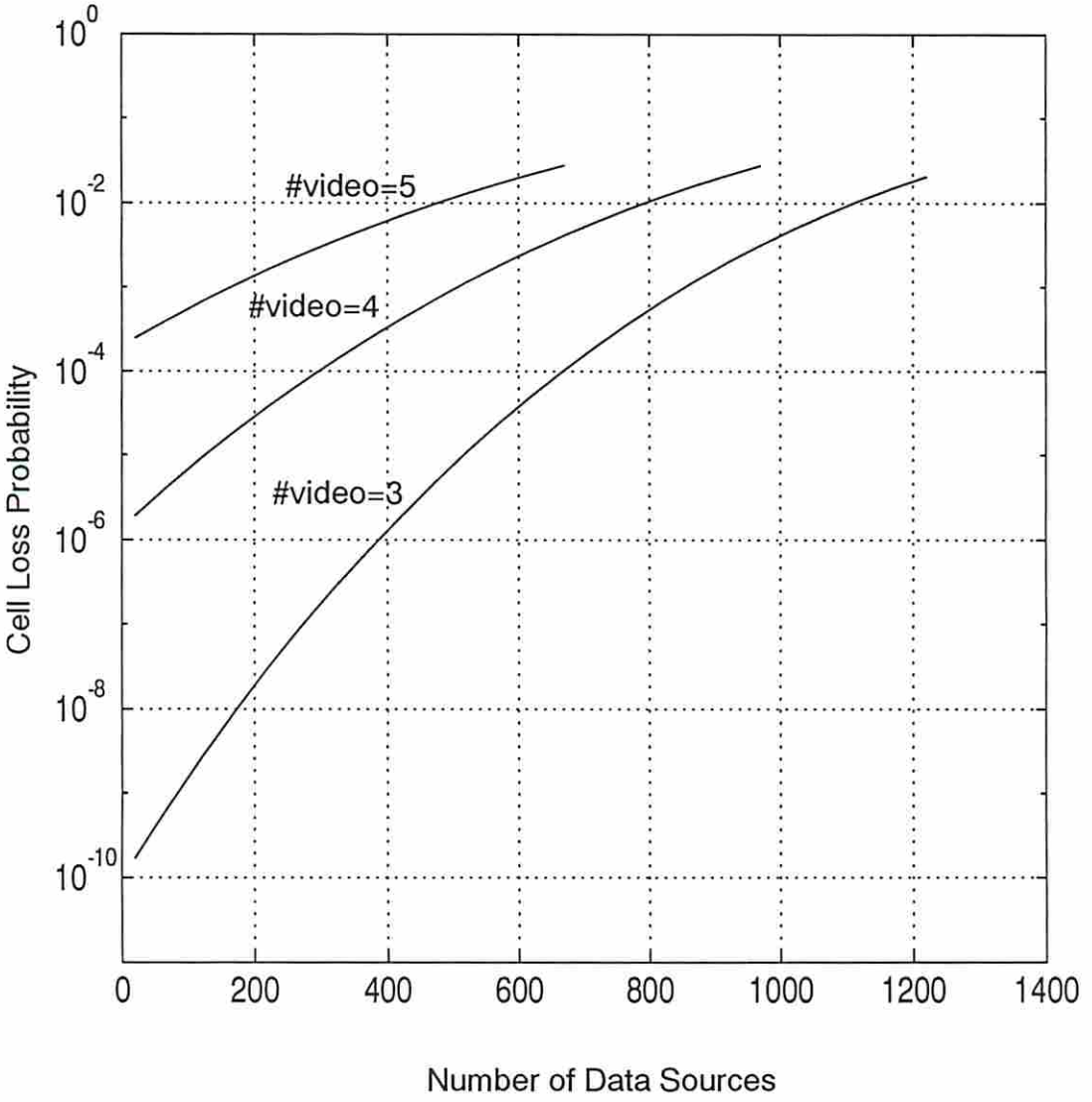

Fig. 5.16 Cell loss probability for different number of video calls with 600 voice calls as a function of the number of data sources.

Chapter 6

Conclusions and Future Research

In this dissertation, we present our research investigating the use of two-state *batch Markov modulated models* as an approximation to represent a complex mixture of different traffic sources. We summarize our results for both continuous-time and discrete-time domains in this chapter followed by some suggestions for the direction of future research.

6.1 Discussions and Conclusions

We specialize part of the analysis for BMAP/G/1 queues presented in [60] and [67] to that for BMMPP/G/1 queues with much simpler and more comprehensible results which are also specialized to some well-know special cases such as $M^{[X]}/G/1$ and MMPP/G/1 queues. We present algorithms to find the tail probability of the queue length distribution, the asymptote for the tail probabilities of the queue length distribution, the expected system delay, and the loss probability. Using these results, we study an ATM multiplexer with a complex combination of different sources by approximating the traffic as a two-state BMMPP.

For systems loaded with voice and data sources, we use a two-state BMMPP and present a parameter matching procedure to model the integrated voice/data traffic. Note that for a similar model presented in [3], the solution of an (*N-L*)-dimentional eigen system is required in order to find the parameters for the MMPP. This requires a significant amount of computational effort especially when $N-L$ is large, i.e., when the load of the system is heavy. Yet, as demonstrated by Fig. 3.2 through Fig. 3.4, the model fails to make a good delay estimation of the modeled system under medium to high system load. Fig. 3.6 through Fig. 3.9 also prove that the model works poorly to predict both loss probability and tail probabilities for these situations. Compared to the above mentioned model, the computational requirement of our approach is insignificant and is independent of the system size. More importantly, our model provides better delay and loss performance predictions (agrees with the simulation better) for a wide range of system loads (the improvement is more noticeable for higher system load).

The approach is extended to cover video, voice and data sources in Chapter 4. Although we provide no comparison to other analytical models (since no similar work exists), simulation

results show that the approximation has a very good delay prediction (see Fig. 4.3 and Fig. 4.4) and loss estimation (see Fig. 4.6 and Fig. 4.7) even under heavy system loads. One can also use Sen's model (Fig. 1.8) to model video sources with scene changes which results in a three-dimensional Markov chain for the integrated multimedia traffic. A similar approximation can then be used to analyze the performance of such systems.

One contribution of this research is the reduction of the size of the state space. The model's state space is reduced from $(N+1)$ to 2 for voice and data integration and from $(M+1) \times (N+1)$ to 2 for video, voice and data integration. For example, the system used in Fig. 4.6 has a state space of 90×100, which is prohibitively large to be solved explicitly (inverting a $(90 \times 100 \times 100)$-dimensional matrix would be required to solve the system with a buffer size of 100). Another contribution is that if only the probabilities of queue occupancy for large buffer positions are of interest, one can use the proposed algorithm to find the corresponding asymptotic values, using (2.23), instead of solving the system explicitly. This indeed provides a very good approximation as shown by Fig. 3.5 through Fig. 3.8 and Fig. 4.5. This research has also shown that (conclusion drawn from Fig. 3.10 and Fig. 4.8) in the environment under study one cannot directly estimate loss probability from the corresponding tail probability of an infinite queue.

Due to the discrete-time nature of ATM networks, we have also developed a similar discrete-time model in Chapter 5 which also has good accuracy. The size of the model's state space is reduced by a factor of $O(M^2 \cdot N^2)$ in our approximation. As an example, consider a system with 20 voice calls, 2 video calls (represented by 21 different levels) and 20 buffer spaces. The exact model would have a state space of size of $9,216 \times 9,216$, where as in our approximation only 60×60 is needed. The time complexity for solving the system drops by a factor of $O(M^3 \cdot N^3)$. Consider a very small system which has only 1 video source, 10 voice sources and 10 buffer spaces, for example. It takes approximately 4.8×10^8 floating point operations to solve the exact system as compared to only about 1.6×10^4 for the approximation. Thus, the complexity of the model is greatly reduced while, as our results have shown, a good accuracy is retained. The model performs better for high loads and in the case of moderate loads, it gives much better agreement for larger buffers (see Fig. 5.10, for example).

One important feature about the proposed model is that we use a set of traffic descriptors which are readily available for each type of traffic, e.g., α, β and ω for voice sources. Therefore, each type of traffic can be specified by a set of user definable parameters. This is particular important, since traffic control (especially admission control) should be based on user-defined parameters instead of system-measured parameters. Thus, network users can be in control as well as responsible for the cost and quality of using the network. Also, several advantages of our approximations can be identified. First, as discussed above, *much less computational overhead is involved.* Second and more importantly, *it provides very good performance predictions.* Third, *a real-time traffic control algorithm can be developed based on the results presented here.*

Another observation which can be made from this research is that the statistical multiplexing gains are much more significant when the systems capacity and the population of traffic sources are significantly larger. We have included a detailed discussion and examples of this observation in Chapter 5.

6.2 Self-similar nature of the traffic

The renewal traffic models used in this research to represent data traffic are expected to work reasonably well for traffic generated by a collection of, say, distributed terminals. Since ATM multiplexers are likely to serve traffic forwarded by *Local Area Networks* (LAN's) as well, it would be quite useful to understand and be able to model the distinct characteristics of this type of traffic.

Recent studies of high-quality, high-resolution LAN traffic measurements [44] have revealed *self-similar* (or *fractal*) property with potentially important implications to the modeling and design of B-ISDN. From a packet arrival process point of view, the self-similar property exhibits structural similarities across a wide range of time scales, i.e, the duration of a burst is no longer well defined. In other words, renewal traffic models have a property that as the time scale increases the number of arrivals in each time unit tends to "smooth" out. Whereas in self-similar traffic models, as the time scale increases the structure of the number of arrivals in each time unit is either indistinguishable or with a small distinction from that of a smaller time scale.

On an attempt to construct a self-similar traffic model, one can aggregate many ON-OFF processes similar to the one shown in Fig. 1.2 with a significant probability of having an arbitrarily long duration of ON periods and OFF periods. (Note that the duration of an ON period and OFF period in an ON-OFF process is exponentially distributed for continuous-time case or geometrically distributed for discrete-time case.) This in fact creates a renewal reward process (with a reward of 0 and 1 for the OFF state and ON state respectively) with an infinite variance for the inter-renewal times (called heavy-tails in [44]).

The questions remaining to be answered are how to match the proposed approximation in this research to the one described above and how well the resulting approximation performs. One thing can be observed from Fig. 4.6 and Fig. 4.7, though, is that using BMMPP as a traffic model can have a similar impact as a self-similar traffic model does on the corresponding loss probability. That is the loss probability cannot be improved significantly by increasing buffer size alone (after a certain size), one of the implications of having a heavy-tail.

6.3 Network traffic analysis

It is not yet clear how the burstiness of ATM network traffic propagates through the network. Even though some studies have suggested that the traffic tends to become smoother at it traverses the network (see [13] for an example), others have concluded that it is the other way around (see [21], [39] and the self-similar traffic discussed in the previous section for examples). Given any specific traffic model for an ATM network, it would be worthwhile to quantify

the burstiness of the traffic as it propagates through the network.

Some efforts have been made towards the analysis of the traffic in an ATM network. As part of the initial step, Saito in [73] and Takine *el al.* in [78] studied the output process of a BMAP/G/1 and D-BMAP/D/1/K queue respectively. More specifically, Takine *et al.* in [78] proved that the output process of a D-BMAP/D/1/K queue is also a D-BMAP. Fonseca and Silvester [19] used a two-state *Markov Modulated Bernoulli Process* (MMBP, a special case of two-state D-BMAP) to approximate the output process of a D-BMAP/D/1/K queue and introduce a framework for the queueing networks with Markov modulated traffic flow. Simply speaking, having a uniform representation for the network traffic (note that joining several D-BMAP's or probabilistically splitting a D-BMAP simply results in another D-BMAP or other D-BMAP's), the network traffic can be analyzed in a similar way as we did for a single node. However, the overall network model is extremely complicated and approximations seem to be inevitable. Thus, it is desirable to have an independence assumption like the well-known *Klenrock's Independence Assumption* for conventional packet switching networks [35].

6.4 Priority systems

We have only studied the overall performance of the system. In some cases, it may be required to estimate the corresponding performance figures for each type of traffic. Also it may be necessary to provide priority to some traffic over others due to QOS requirements. This priority could be in terms of service priority (non-FIFO service) for delay sensitive traffic such as real-time voice traffic, or in terms of space priority (at the waiting buffer) for loss sensitive traffic such as data traffic, or both for compressed video signals, for example. Even though M/G/1 systems with service priorities have been well studied, there have been little work on BMAP/G/1 (or even BMMPP/G/1) queueing systems with multiple priorities. Nevertheless, some related work exists in the literature.

In [28], Herrmann studied the loss probability of each traffic class (two traffic classes were considered) of a D-BMAP+D-MAP/D/1/K queueing system without priorities. Bae *et al.* [2] examined a single server (with an exponentially distributed service time) loaded with several traffic classes each of which is modeled by an MMPP. They studied individual packet loss for each traffic stream under a (space) priority discarding scheme. Kofman and Korezlioglu [38] extended the results presented by Bae *et al.* in [2] to that of BMAP/M/1 queues. It would be interesting to investigate BMAP/G/1 queues that provide both space and service priorities. A priority queueing system with a reduced state space as the ones proposed in this research to make the results well-suited for real-time implementation would also be very useful.

6.5 Admission control algorithm

One can use the results presented in this research as the framework to develop an admission control algorithm for high-speed packet-switching networks. Using a set of user-defined parameters (the traffic descriptors) for each traffic class, different performance measures, such as loss probability and expected buffer occupancy, can be computed in real-time. These measures can be used in conjunction with other factors such as QOS requirements of the traffic as the basis for

admission control. If the user characterization is not reliable, we could measure traffic characteristics to determine parameter for our model.

Some details remaining to be worked out are: (a) how to describe (or estimate) the traffic descriptors? Are we holding the network subscribers responsible for specifying their traffic descriptors or the network providers for making different choices of traffic descriptor sets available? The point is that network subscribers may not know exactly how to describe their traffic. On the other hand, limiting the choice of traffic descriptor sets may not satisfy all users. (b) How to specify the QOS requirements? Some QOS measures including loss probability, average delay, delay bound, and delay jitter, have been considered in the literature some of which are end-to-end performance measures. For example, the delay jitter distribution is expected to be highly depended on the flow and congestion control algorithm (such as the *Virtual Clock* proposed in [85]) used by the network. The algorithm must be able to compute (or provide) not only the QOS to be received by the new connection request, but also determine the performance impact on the existing connections. (c) What kind of interaction between different functionalities of the traffic control algorithm should be considered? For example, given that the system is using a specific flow and congestion control algorithm, how does the admission control algorithm adjust itself to react to a certain undesired network condition such as network congestion? Thus, it may be necessary to consider the consequences of the interactions with the lower-level traffic control algorithm such as flow and congestion control and packet discarding policy. In other words, the function of admission control may be different from that of the lower-level controls, but, at least in some respects, it depends on the lower-level controls as well.

Appendix A

Loss Probability for BMMPP/D/1/*K* queues

In this appendix we assume that the system under study is *stable* (meaning that the average system load is less than the average system capacity, i.e., with probability one the time between two occurrences of an empty system is less then infinite) and both the average arrival rate and average batch size are finite. We then prove that (under steady state) the loss probability can be obtained directly from the expected number of losses versus the expected number of arrivals between two consecutive departures.

We use the following notation for the BMMPP/D/1/K system:

- Q_i is the queue length of the BMMPP/D/1/K queue right after the ith departure;
- F_i is the phase of the underlying Markov chain of the BMMPP right after the ith departure;
- A_i is the number of arrivals between the ith and the $(i+1)$th departures;
- L_i is the number of losses between the ith and the $(i+1)$th departures.

From the definitions above it can be clearly seen that A_i depends only on F_i and that the relations between these random variables are given by:

$$Q_{i+1} = [Q_i + A_i - L_i - 1]^+ \text{ and } L_i = [A_i - (K - Q_i)]^+ \tag{A.1}$$

where $[x]^+$ is defined to be $max\{0, x\}$. Thus, if we let $X_i = (Q_i, F_i, A_i)$ and $Y_i = (Q_i, F_i, L_i)$, it can be readily seen that X_{i+1} depends only on X_i and Y_{i+1} depends only on Y_i and that both $\{X_i\}$ and $\{Y_i\}$ are irreducible, positive recurrent (since the system is assumed to be *stable*) Markov chains with some steady state distribution, say, π_X and π_Y respectively.

If we let the function $f(X_i) = A_i$ and observe that $E_{\pi_X}|f(X_1)| = E_{\pi_X}[A_1] < \infty$, by applying the *Strong Law of Large Numbers for Markov Chains* (Theorem 9.4 of [5]) we get the following expression:

$$\lim_{n \to \infty}\left(\frac{1}{n}\sum_{i=1}^{n} f(X_i)\right) = \lim_{n \to \infty}\left(\frac{1}{n}\sum_{i=1}^{n} A_i\right) = E_{\pi_X}[f(X_1)] = E_{\pi_X}[A_1] \tag{A.2}$$

Similarly, we can prove that:

$$\lim_{n \to \infty}\left(\frac{1}{n}\sum_{i=1}^{n} L_i\right) = E_{\pi_Y}[L_1] \tag{A.3}$$

From (A.2) and (A.3), we conclude that the cell loss probability can be readily obtained using:

$$\lim_{n \to \infty}\left(\frac{\sum_{i=1}^{n} L_i}{\sum_{i=1}^{n} A_i}\right) = \frac{E_{\pi_Y}[L_1]}{E_{\pi_X}[A_1]} \tag{A.4}$$

Bibliography

[1] J. J. Bae and T. Suda, "Survey of Traffic Control Schemes and Protocols in ATM Networks," *Proc. IEEE*, pp. 170-189, Feb. 1991.

[2] J. J. Bae *et al.*, "Analysis of Individual Packet Loss in a Finite Buffer Queue with Heterogeneous Markov Modulated Arrival Process: A study of Traffic Burstiness and Priority Packet Discarding," in *Proc. IEEE INFOCOM '92*, pp. 2C.1.1-2C.1.12, 1992.

[3] A. Baiocchi *et al.*, "Loss Performance Analysis of an ATM Multiplexer Loaded with High-Speed On-Off Processes," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 388-393, Apr. 1991.

[4] Bell Communications Research, "Generic System Requirements in Support of Switched Multi-megabit Data Service," *Technical Reference*, TR-TSV-000772, Issue 1, May 1991.

[5] R. N. Bhattacharya and E. C. Waymire, *Stochastic Process with Applications*, New York, NY: John Wiley & Sons, 1990.

[6] C. Blondia, "The N/G/1 Finite Capacity Queue," *Stochastic Models*, vol. 5, no. 2, pp. 273-294, 1989.

[7] C. Blondia and O. Casals, "Performance Analysis of Statistical Multiplexing of VBR Sources," in *Proc. IEEE INFOCOM '92*, pp. 6C.2.1-6C.2.11.

[8] P. T. Brady, "A Statistical Analysis of On-Off pattern in 16 Conversations," *Bell Systems Technical Journal*, pp. 73-91, Jan. 1968.

[9] P. T. Brady, "A Model for Generating On-Off Speech Patterns in Two-Way Conversation," *Bell Systems Technical Journal*, pp. 2445-2472, Sep. 1969.

[10] G. L. Choudhury and D. M. Lucantoni, "Numerical Computation of Large Number of Moments with Application to Asymptotic Analysis," to appear in *Operations Research.*

[11] I. Cidon *et al.*, "Bandwidth Management and Congestion Control in plaNET," *IEEE Commun. Mag.*, vol. 29, pp. 54-64, Oct. 1991.

[12] J. N. Daigle and J. D. Langford, "Models for Analysis of Packet Voice Communications Systems," *IEEE J. Select. Areas Commun.*, vol. 4, pp. 847-855, Sep. 1986.

[13] B. S. Davie and A. Desclouz, "The Effects of Bursty Traffic on ATM Switching Systems," *Technical Report*, TM-ARH-016175, Bell Communications Research, Feb. 1990.

[14] L. Dittmann *et al.*, "Flow Enforcement Algorithms for ATM Networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 343-350, Apr. 1991.

[15] S. S. Dixit and P. Skelly, "Video Traffic Smoothing and ATM Multiplexer Performance," in *Proc. IEEE GLOBECOM '91*, pp. 8B3.1-8B3.4.

[16] A. E. Eckberg, "The Single-Server Queue with Periodic Arrival Process and Deterministic Service Time," *IEEE Trans. Commun.*, vol. 27, pp. 556-562, Mar. 1979.

[17] A. E. Eckberg *et al.*, "Controlling Congestion in B-ISDN/ATM: Issues and Strategies," *IEEE Commun. Mag.*, vol. 29, pp. 64-70, Sep. 1991.

[18] J. M. Ferrandiz and A. A. Lazar, "Admission Control for Real-Time Packet Sessions," in *Proc. IEEE INFOCOM '91*, pp. 6A.2.1-6A.2.7.

[19] N. L. S. Fonseca and J. A. Silvester, "Modeling the Out Put Process of an ATM Multiplexer with Markov Modulated Arrivals," in Proc. *IEEE ICC '94*, pp. 721-725, 1994.

[20] H. Gilbert *et al.*, "Developing a Cohesive Traffic Managements Strategy for ATM Networks," *IEEE Commun. Mag.*, vol. 29, pp. 36-45, Oct. 1991.

[21] S. J. Golestani, "A Framing Strategy for Congestion Management," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1064-1077, Sep. 1991.

[22] R. L. Graham, *Concrete Mathematics: A Foundation for Computer Science*, Menlo Park, CA: Addison-Wesley, 1989.

[23] R. Grunenfelder *et al.*, "Characterization of Video Codecs as Autoregressive Moving Average Processes and Related Queueing System Performance," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 284-293, Apr. 1991.

[24] R. Guerin *et al.*, "Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 968-981, Sep. 1991.

[25] I. W. Habib and T. N. Saadawi, "Controlling Flow and Avoiding Congestion in Broadband Networks," *IEEE Commun. Mag.*, vol. 29, pp. 46-53, Oct. 1991.

[26] R. Handel, "Evolution of ISDN Toward Broadband ISDN," *IEEE Network*, vol. 3, pp. 7-13, Jan. 1989.

[27] H. Heffes and D. M. Lucantoni, "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance," *IEEE J. Select. Areas Commun.*, vol. 4, pp. 856-868, Sep. 1986.

[28] C. Herrmann, "Discrete-Time Models for Connection Admission Control (CAC) in ATM," in *Proc. 2nd Workshop on Performance Modeling and Evaluation of ATM Networks*, pp. 25.1-25.12, Jul. 1994.

[29] D. Hong and T. Suda, "Congestion Control and Prevention in ATM Networks," *IEEE Network Mag.*, pp. 10-16, Jul. 1991.

[30] D. Hong et at., "Survey of Techniques for Prevention and Control of Congestion in an ATM Network," in *Proc. ICC '91*, pp. 6.5.1-6.5.7.

[31] J. Y. Hui, "Resource Allocation for Broadband Networks," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 1598-1608, Dec. 1988.

[32] H. E. Hurst, "Long-Term Storage Capacity of Reservoirs," *Trans. Amer. Soc. Civil Engineers*, vol. 116, pp.770-799, 1951.

[33] I. Ide, "Superposition of Interrupted Poisson Processes and Its Application to Packetized Voice Multiplexers," in *Proc. ITC '88*, pp. 3.1B2.

[34] I. Khan and V. O. K. Li, "Performance Analysis at an ATM Statistical Multiplexer Serving a Superposition of Bursty Traffic Sources," in *Proc. ISCA/ACM International Conference on Computer Communications and Networks*, San Diego, June 1993, pp. 421-425.

[35] L. Kleinrock, *Queueing Systems, Vol II: Computer Applications*, John Wiley & Sons, New York, NY, 1976.

[36] L. Kleinrock, "ISDN-The Path to Broadband Networks," *Proc. IEEE*, vol. 79, pp. 112-117, Feb. 1991.

[37] C. Knessl and B. J. Matkowsky, "A Markov Modulated M/G/1 Queue I: Stationary Distribution," *Queueing Systems*, vol. 1, pp. 355-374, 1987.

[38] D. Kofman and H. Korezlioglu, "Loss Probabilities and Delay and Jitter Distributions in a Finite Buffer Queue with Heterogeneous Batch Markovian Arrival Processes," in *Proc. IEEE GLOBECOM '93*, pp. 830-834, 1993.

[39] R. Krishnan *et al.*, "Jitter at an ATM Multiplexer in the Presence of Correlated Traffic," *Technical Report*, CEng 94-14, EE-Systems, USC, Apr. 1994.

[40] S. S. Lam *et al.*, "An Algorithm for Lossless Smoothing of MPEG Video," to appear in *Proc. ACM SIGCOMM '94*, 1994.

[41] A. A. Lazar and G. Pacifici, "Control of Resources in Broadband Networks with Quality of Service Guarantees," *IEEE Commun. Mag.*, pp. 66-73, Oct. 1991.

[42] A. A. Lazar *et al.*, "Real-Time Traffic measurements on MAGNET II," *IEEE J. Select. Areas Commun.*, Vol. 8. pp. 467-483, Apr. 1990.

[43] D. Le Gall, "MPEG: A Video Compress for Multimedia Applications," *Commun. ACM*, vol. 34, pp. 46-58, Apr. 1991.

[44] W. E. Leland *et al.*, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," *IEEE/ACM Trans. Networking*, vol. 2, pp. 1-15, Feb. 1994.

[45] S.-Q. Li, "A General Solution Technique for Discrete Queueing Analysis of Multimedia Traffic on ATM," *IEEE Trans. Commun.*, vol. 39, pp. 1115-1132, Jul. 1991.

[46] S.-Q. Li, "Study of Information Loss in Packet Voice Systems," *IEEE Trans. Commun.*, vol. 37, pp. 1192-1202, Nov. 1989.

[47] M. Liebhold and E. M. Hoffert, "Toward an Open Environment for Digital Video," *Commun. ACM*, vol. 34, pp. 103-112, Apr. 1991.

[48] A. Y.-M. Lin and J. A. Silvester, "Priority Queueing Strategies and Buffer Allocation Protocols for Traffic Control at an ATM Integrated Broadband Switching System," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1524-1536, Dec. 1991.

[49] M. L. Liou, "Overview of the pX64 kbps Video Coding Standard," *Commun. ACM*, Apr. 1991.

[50] D. M. Lucantoni, "New Results on the Single Server Queue with a Batch Markov Arrival Process," *Stochastic Models*, vol. 7, no. 1, pp. 1-46, 1991.

[51] D. M. Lucantoni, "The BMAP/G/1 Queue: A Tutorial," in *Models and Techniques for Performance Evaluation of Computer and Communications Systems*, Eds: L. Donatiello and R. Nelson, Springer Verlag, 1993.

[52] D. M. Lucantoni and M. F. Neuts, "The Customer Delay in a Single Server Queue with a Batch Markovian Arrival Process," *submitted for publication.*

[53] B. Maglaris *et al.*, "Performance Models of Statistical Multiplexing in Packet Video Communications," *IEEE Trans. Commun.*, vol. 36, pp. 834-844, Jul. 1988.

[54] S. E. Minzer, "Broadband ISDN and Asynchronous Transfer Mode (ATM)," *IEEE Commun. Mag.*, vol. 27, pp. 17-24, Sep. 1989.

[55] B. Mukherjee, "Integrated Voice-Data Communications Over High-Speed Fiber Optic Networks," *Computer*, pp. 49-58, Feb. 1991.

[56] T. Murase *et al.*, "A Call Admission Control Scheme for ATM Networks Using a Simple Quality Estimate," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1461-1470, Dec. 1991.

[57] T. Murase *et al.*, "A Call Admission Control for ATM Networks Based on Individual Multiplexed Traffic Characteristics," in *Proc. IEEE INFOCOM '91*, pp. 6.3.1-6.3.6

[58] R. Nagarajan *et al.*, "Approximation Techniques for Computing Packet Loss in Finite-Buffered Voice Multiplexers," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 368-377, Apr. 1991.

[59] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*, Baltimore, MD: Johns Hopkins Univ. Press, 1981.

[60] M. F. Neuts, "A Versatile Markovian Point Process," *J. Applied Prob.*, vol. 16, pp.764-779, Dec. 1979.

[61] M. Nomura *et al.*, "Basic Characteristics of Variable Rate Video Coding in ATM Environment," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 752-760, Jun. 1989.

[62] T. Okada *et al.*, "Traffic Control in Asynchronous Transfer Mode," *IEEE Commun. Mag.*, vol. 29, pp. 58-62, Sep. 1991.

[63] N. U. Prabhu and Y. Zhu, "Markov Modulated Queueing Systems," *Queueing Systems*, vol. 5, pp. 215-246, 1989.

[64] G. Ramamurthy and R. S. Dighe, "A Multidimensional Framework for Congestion Control in B-ISDN," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1440-1451, Dec. 1991.

[65] G. Ramamurthy and R. S. Dighe, "Distributed Source Control: A Network Access Control for Integrated Broadband Packet Networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 990-1002, Sep. 1991.

[66] G. Ramamurthy and B. Sengupta, "Delay Analysis of a Packet Voice Multiplexer by the $\Sigma D_i/D/1$ Queue," *IEEE Trans. Commun.*, vol. 39, pp. 1107-1114, Jul. 1991.

[67] V. Ramaswami, "The N/G/1 Queue and Its Detailed Analysis," *Adv. Appl. Prob.*, vol. 12, pp. 222-261, Mar. 1980.

[68] C. Rasmussen *et al.*, "Source-Independent Call Acceptance Procedures in ATM Networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 351-358, Apr. 1991.

[69] E. P. Rathgeb, "Modeling and Performance Comparison of Policing Mechanisms for ATM Networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 325-334, Apr. 1991.

[70] K. R. Raymond nd E. A. Walvick, "Broadband, An Evolution," in *Proc. IEEE ICC '91*, pp. 39.6.1-39.6.6.

[71] J. W. Roberts, "Variable-Bit-Rate Traffic Control in B-ISDN," *IEEE Commun. Mag.*, vol. 29, pp. 50-56, Sep. 1991.

[72] I. Rubin and T. Cheng, "Admission Control for Multi-Layer Management of High-Speed Packet-Switched Networks under Observation Noise," in *Proc. IEEE INFOCOM '91*, pp. 6A.4.1-6A.4.9.

[73] H. Saito, "The Departure Process of an N/G/1 Queue," *Performance Evaluation*, vol 11. pp. 241-251, 1990.

[74] H. Saito and K. Shiomoto, "Dynamic Call Admission Control in ATM Networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 982-989, Sep. 1991.

[75] P. Sen *et al.*, "Models for Packet Switching of Variable-Bit-Rate Video Sources," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 865-869, Jun. 1989.

[76] C. Shim *et al.*, "Modeling and Call Admission Control Algorithm of Variable Bit Rate Video in ATM Networks," *IEEE J. Select. Areas Commun.*, vol. 12, pp. 332-344, Feb. 1994.

[77] K. Sriram and W. Whitt, "Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data," *IEEE J. Select. Areas Commun.*, vol. 4, pp. 833-846, Sep. 1986.

[78] T. Takine *et al.*, "Cell Loss and Output Process Analysis of Finite Buffer Discrete Time Queueing System with Correlated Arrivals," in *Proc. IEEE INFOCOM '93*, pp. 1259-1268, 1993.

[79] R. C. F. Tucker, "Accurate Method for Analysis of a Packet-Speech Multiplexer with Limited Delay," *IEEE Trans. Commun.*, vol. 36, pp. 479-483, Apr. 1988.

[80] W. Verbiest, "The Impact of the ATM Concept in Video Coding," *IEEE J. Select. Areas Commun.*, vol. 6, pp. 1623-1632, Dec. 1989.

[81] G. K. Wallace, "The JPEG: A Video Compression Standard for Multimedia Applications," *Commun. ACM*, vol. 34, pp. 46-58, Apr. 1991.

[82] G. M. Woodruff and R. Kositpaiboon, "Multimedia Traffic Management Principles for Guaranteed ATM Network Performance," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 437-446, Apr. 1990.

[83] J. Ye and S.-Q. Li, "Analysis of Multimedia Traffic Queues with Finite Buffer and Overload Control-Part 1: Algorithm," in *Proc. IEEE INFOCOM '91*, pp. 12c.3.1-12c.3.11.

[84] C. Yuan and J. A. Silvester, "Queueing Analysis of Delay Constrained Voice Traffic in a Packet Switching System," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 729-738, Jun. 1989.

[85] L. Zhang, "Virtual Clock: A New Traffic Control Algorithm for Packet-Switched Networks," *ACM Trans. Computer Systems*, vol. 9, no. 2, pp. 101-124, May 1991.

[86] Y. Zhu, "A Markov Modulated M/M/1 Queue with Group Arrival," *Queueing Systems*, vol. 8, pp. 255-264, Apr. 1991.