Loss Performance and Queue Length Statistics for Multimedia Communication Systems

Shiouming Stanley Wang and John A. Silvester

CENG Technical Report 93-30

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

September 1993

Loss Performance and Queue Length Statistics for Multimedia Communication Systems

Shiouming Stanley Wang and John A. Silvester

Department of Electrical Engineering-Systems
University of Southern California
Los Angeles, California 90089-2562

Abstract

In this paper, we study the performance of integrated services ATM multiplexers in terms of their loss performance. We model voice sources as *ON-OFF processes*, video sources with uniform activity level as a *birth-death process* and the arrival process from the aggregation of data sources as a *Poisson process* with a general bulk size distribution. We show that the integration of these processes belongs to the family of *Batch Markovian Arrival Processes* (BMAP). We investigate an approximation which matches the original model to a two-state *Batch Arrival Markov Modulated Poisson Process* (BMMPP). We investigate the queue length distribution with particular emphasis on tail probabilities for which we develop an easy way to find its asymptote using its Z-transform. One of the major contributions of this research is that the proposed asymptote, which can be very easily obtained, can be used as a very good approximation of the tail probabilities especially for large buffer positions. Another contribution is that we provide an simple way to calculate the loss probability of a single-server queue with correlated arrivals. The results show that the proposed model presents greatly reduced complexity while retaining reasonably accurate performance prediction. We also demonstrate that the approach of estimating loss probability using tail probabilities is valid only for light to moderate system load for the systems under study.

1 Introduction

With the increasing affordability of powerful workstations, which are capable of handling not only text but also sound, graphics and even motion pictures, and recent advances in high-speed communications technologies, which allow high-data-rate services, such as videoconferencing and videotelephony, future communication networks are expected to carry various types of traffic, for example: data, voice, image and video. Due to their complexity, modeling such systems has become one of the major challenges for researchers in this field.

In this paper, voice sources are modelled as *ON-OFF* processes[4]; video sources with uniform activity level are modelled as a birth-death process [12]; and data sources are modelled as a Poisson process with batch (cell) arrivals. We propose to use a two-state *Batch Arrival Markov Modulated Poisson Process* (BMMPP) to approximate the integration of a number of video, voice and data sources which are multiplexed into an *Asynchronous Transfer Mode* (ATM) network. We model the multiplexer as a two-state BMMPP/D/1/K queue and study its loss performance.

A BMMPP, which is a special case of *Batch Markovian Arrival Processes* (BMAP) [11], [14], is a *doubly stochastic Poisson process with batch arrivals* where the arrival rate is determined by the state of the underlying finite-state Markov chain, i.e., the arrivals generated by an BMMPP are highly correlated. BMAP were first introduced by Neuts (called *Neuts' versatile Markov processes* or *N-processes*) in [14]. Lucantoni [11] revises the notation used in the original computational algorithm (presented by Ramaswami in [15]) for BMAP/G/1 (called N/G/1 in [15]) queueing systems, which has both CPU-time and storage burdens, and introduced a new algorithm that is feasible to be implemented in practice.

In [2], Blondia presents the analysis of a finite buffer BMAP/G/1 queueing system and its batch blocking probability (the probability that an arrived batch is completely blocked due to no available buffer space). We present the analysis of the (cell) loss probability of BMMPP/D/1/K queueing systems, i.e., we assume that cells arriving in the same batch are accepted up to the number of available free spaces and that the rest are lost. We prove that even though the arrival process of a BMMPP/D/1/K queueing system is highly correlated, the *Law of Large Numbers* (LLA) can still be applied to find the (cell) loss probability. We also study a related issue, the *tail probabilities* (also called *survivor function*), of the queue length distribution of the BMMPP/D/1 queues with infinite buffers and present an easy-to-implement algorithm to find the asymptote for the tail probabilities. Using this approach, the tail probabilities and the queue length probabilities for large buffer positions can be easily approximated without actually solving the system. This in fact would be a quite useful result in practice, since future ATM networks are expected to have a very low, say 10^{-10}, loss probability. In other words, regardless of the application, tail probabilities and queue length probabilities for small to medium buffer positions are not of interest.

In the next section, we first discuss the individual traffic models used in this research. We then introduce an approximation for the integration of different types of traffic. The analysis of the BMMPP/D/1/K queues, including an algorithm to find the asymptote for the queue length tail probabilities, are presented in section 3. Some numerical examples are given in section 4 with some closing remarks provided in section 5.

2 Traffic Model

In this section, we explain the traffic model and the approximation technique used in this research. Specifically, voice sources are modelled as *ON-OFF* processes; video sources are modelled as a birth-death process; and data sources are modelled as a Poisson process with batch arrivals. We approximate the integration of these sources as a two-state BMMPP. Detailed discussions about the approximation technique are available in [17].

2.1 Model for voice traffic

We model a voice source by an *ON-OFF* process in which the voice source alternates between exponentially distributed *ON* and *OFF* periods. Cells are generated with a constant arrival rate ω during the *ON* periods and no cells are generated during the *OFF* periods [5]. Thus, an *ON-OFF* process can be defined by three parameters, the transition rates from-ON-to-OFF (β) and from-OFF-to-ON (α) and ω. Fig. 1 shows the model for the superposition of N independent voice sources where the states represent the number of voice sources in the *ON* state. (Note that, this is a special case of an (N+1)-state BMAP)

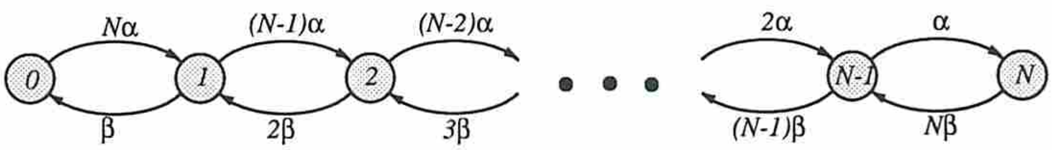

Fig. 1. Model for superposition of N voice sources.

2.2 Model for video traffic

We adopt the model originally proposed by Maglaris *et al.* [12] for video sources with uniform activity level. This model assumes that there are no sudden movements in the video scenes, e.g., a video scene from a videotelephone connection showing a person talking in front of the camera. This model can be viewed as the superposition of M mini-sources each of which has a cell arrival rate of, say, η and transition probabilities of, say, a and b (see [12] for details). Thus, the cell arrival rate from all video sources can be fitted into $M + 1$ equal-distance discrete levels, $0, \eta, 2\eta, \ldots, M\eta$. 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. Fig. 2 shows our model for video sources with the states representing the cell arrival rate levels.

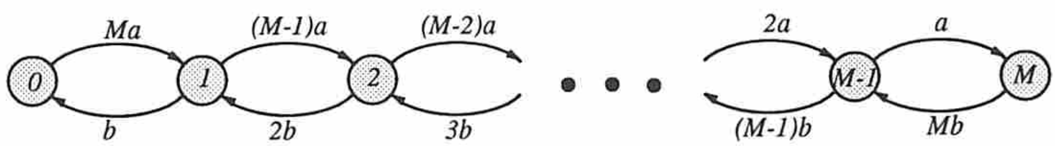

Fig. 2. Model for video sources with M+1 levels of cell arrival rate.

2.3 Model for data traffic

Packet arrivals from aggregated data sources are assumed to form a *Poisson process* (single packet) with a parameter μ, i.e., the packet interarrival time is exponentially distributed with the same parameter. We assume that the size of the data packets has a general probability mass function $s_k = Pr\{\text{data packet is comprised of } k \text{ cells}\}$, $k = 1, 2, 3, \ldots$, with $\bar{s} = \sum_{k=1}^{\infty} k s_k$ being the average number of cells per packet and that it is non-trivial, i.e., $s_k > 0$ for some k.

2.4 Approximation of voice sources using two-state MMPP

Consider an ATM multiplexer loaded with N independent voice sources, which is modeled by the *birth-death process* shown in Fig. 1. Let C denote the capacity of the multiplexer (in cells per unit of time) available to the voice sources (i.e., the total capacity is reduced by the amount consumed by data sources, if any); $L = \lfloor C/\omega \rfloor$ 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); and π_k, $0 \le k \le N$, be the steady state probability that k out of N voice sources are in the ON-state. In [17], we proposed a matching procedure for approximating the superposed arrival process corresponding to N voice sources as a two-state MMPP. The matching procedure differs from the one proposed by Baiocchi *et al.* in [1] in the way γ_2 is calculated, which results in a great saving in the computation time required for parameter matching especially when the number of sources is large (note that, solving an (N-L)-dimentional eigen-system is required to find γ_2 in [1]). Readers are referred to [17] for detailed discussions on the differences between these two models and other models along this line (e.g., [7], [9], [13] and [16]). The matching procedure is given as follows:

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

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

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

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

2.5 Using two-state MMPP as an approximation for voice and video integration

To include video traffic into the model, we combine the models for voice and video sources (Fig. 1 and Fig. 2) which results in a *two-dimensional Markov chain* (note that, this is a special case of (N+1)(M+1)-state BMAP), where the states represent the level of cell arrival rate for video sources and the number of active voice sources. A state (i, j) with $(\eta i + \omega j) > C$ has an arrival rate greater than the system capacity. We call such a state *overloaded*; otherwise, it is *underloaded*. In [17], we proposed the following extension of the matching procedure for the integration of voice and video sources:

$$i.\quad \gamma_2 = \left(\sum_{i=0}^{M} (\sigma_i + 1) / \sum_{i=0}^{M} (N - \sigma_i)\right)(\beta + b)$$

$$ii.\quad \lambda_1 = \sum_{(i,j) \in S_u} (\eta i + \omega j)\left(\frac{\pi_{ij}}{\Pi_u}\right), \text{ where } \Pi_u = \sum_{(x,y) \in S_u} \pi_{xy}$$

$$iii.\quad \lambda_2 = \sum_{(i,j) \in S_o} (\eta i + \omega j)\left(\frac{\pi_{ij}}{\Pi_o}\right), \text{ where } \Pi_o = \sum_{(x,y) \in S_o} \pi_{xy}$$

$$iv.\quad \gamma_1 = \gamma_2 \frac{(\omega\phi_1 + \eta\phi_2) - \lambda_1}{\lambda_2 - (\omega\phi_1 + \eta\phi_2)}, \text{ where } \phi_1 = N\frac{\alpha}{\alpha + \beta} \text{ and } \phi_2 = M\frac{a}{a + b}$$

where S_u and S_o are the sets of the underload states and overload states respectively; π_{ij}, $0 \le i \le M$, $0 \le j \le M$, is the steady state probability of the Markov chain being in state (i, j); and σ_i is the number of voice sources that the system can support given that the video sources are currently in cell arrival rate level i. Note that σ_i's must satisfy the following conditions: $\sigma_i = \lfloor (C - \eta i) / \omega \rfloor$ and $-1 \le \sigma_i \le N$ (where $\sigma_i = -1$ counts the situation when video sources in cell arrival rate level i alone overload the system).

2.6 The integration of video, voice and data sources

It can be clearly seen that the integration of the video and voice sources (modeled by the adopted two-state MMPP approximation) and the data sources (modeled by a batch arrival Poisson process) is a two-state BMMPP with: (for phase i, $i = 1, 2$) $\gamma'_i = \gamma_i$ and $\lambda'_i = \lambda_i + \mu$, and the probability that a batch has a size of k cells given by:

$$P_{ik} = s_k \frac{\mu}{\mu + \lambda_i} + \delta_{k-1} \frac{\lambda_i}{\mu + \lambda_i}, \quad k = 1, 2, 3, \dots \tag{1}$$

where

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

3 Analysis of the BMMPP/D/1/K Queue

In this section, we compute the queue length distribution of a two-state BMMPP/D/1/K queue, which is used to model an ATM multiplexer loaded with the different types of traffic described in the previous section. We also propose an easy way to find the asymptote for the tail probabilities of the queue length distribution. We adopt the following notation:

- $\vec{q} = (q_1, q_2) = (\gamma_2 / (\gamma_1 + \gamma_2), \gamma_1 / (\gamma_1 + \gamma_2))$ is the steady state probability of the underlying Markov chain of the BMMPP.
- $(\boldsymbol{A}_n)_{ij}$, the (i, j) component of matrix $\boldsymbol{A}_n$, $n = 0, 1, 2, \ldots$, $1 \le i, j \le 2$, is the probability that given the system is not empty and the arrival process is in phase i at the beginning of a service period, there are n arrivals during the service period and the phase of the two-state BMMPP is j at the end of the service period.
- $(\boldsymbol{B}_n)_{ij}$, the (i, j) component of matrix $\boldsymbol{B}_n$, $n = 0, 1, 2, \ldots$, $1 \le i, j \le 2$, is the probability that given the system is left empty by a cell departure and the arrival process is in phase i right after the departure, there is a total of n+1 arrivals until the next cell departure instant and the phase of the two-state BMMPP is j at the end of this period.
- x_{ij}, $1 \le i \le 2$, $0 \le j \le K-1$, is the steady state joint probability of the phase being i and the queue length being j at cell departure instants (right after each departure).
- $\tilde{x}_i = x_{1i} + x_{2i}$, $0 \le i \le K-1$, is the steady state probability of the queue length being i.
- $\vec{X} = \{\vec{x}_0, \vec{x}_2, \ldots, \vec{x}_{K-1}\}$, where $\vec{x}_i = (x_{1i}, x_{2i})$, $0 \le i \le K-1$, is the steady state joint probability distribution of the queue length and the phase of the two-state BMMPP at cell departure instants.
- $\vec{Y}$, which has a similar structure as $\vec{X}$, is the steady state joint probability distribution of the queue length (including server) and the phase of the two-state BMMPP at any time instant.

3.1 Queue length distribution at departure instants

We specialize the algorithm presented by Lucantoni [11] to that of BMMPP/D/1/K queues for computing $\boldsymbol{A}_n$'s and $\boldsymbol{B}_n$'s. Once these matrices are known, we can easily see that $\vec{X}$ is the invariant probability vector of the following stochastic matrix:

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

Thus, we can solve $\vec{X}$ directly using the facts that $\vec{X}R = \vec{X}$ and $\sum_{i=0}^{K-1} x_{1i} + x_{2i} = 1$. Specifically,

$$
\vec{X} = \vec{b} \cdot (\tilde{R})^{-1} \tag{3}
$$

where $\vec{b} = (0, 0, \dots, 0, 1)$ and $\tilde{R} = R - I$ with all elements of its last column replaced by 1. In practice, for $K = \infty$ (i.e., a BMMPP/D/1 queue with infinite buffers), the A_n's and B_n's are calculated up to a certain number where the truncation error is small enough to be ignored. According to our experience for an error tolerance of 10^{-20} (which is assumed to be the case in all examples, which require to solve a BMMPP/D/1/∞, presented in this research), around 190 A_n's and B_n's are required to be calculated.

3.2 Loss probability

Let:

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

Making use of the queue length distribution at cell departure instants obtained in section 3.1, it is not difficult to show that:

$$
N_a = \tilde{x}_0 \left(\sum_{n=1}^{\infty} n B_{n-1} \vec{e} \vec{q} \right) + (1 - \tilde{x}_0) \left(\sum_{n=1}^{\infty} n A_n \vec{e} \vec{q} \right) \tag{4}
$$

$$N_l^K = \tilde{x}_0\left(\sum_{n=1}^{\infty}(n-K)^+ B_{n-1}\dot{e}\dot{q}\right) + \left(\sum_{n=1}^{\infty}\sum_{k=1}^{K}\tilde{x}_k(n+k-K)^+ A_n\dot{e}\dot{q}\right) \tag{5}$$

where $\dot{e} = (1, 1)'$ and $(x)^+$ is defined to be $max\{0, x\}$. Note that in (4) and (5), the first term counts the corresponding numbers for the situation where a cell departs leaving an empty system; while the second term counts the corresponding numbers for the situation where a cell departs with at least one cell in the system.

To calculate the cell loss probability, we first point out that the random variables representing the number of cells arriving between consecutive departures are not independent (nor are the numbers of lost cells between consecutive departures). Thus the *Law of Large Numbers* can not be applied directly to calculate the cell loss probability. By making use of the *Law of Large Numbers for Markov Chains* (Theorem 9.4 of [2]) however, we prove (see *Appendix*) that the cell loss probability can be calculated by:

$$l_K = N_l^K / N_a \tag{6}$$

3.3 Asymptote for the tail probabilities of queue length distribution

We specialize the Z-transform of the queue length distribution at any time instants for BMAP/G/1 queues (derived also in [11]) to that for two-state BMMPP/D/1 queues as:

$$Y(z) = \bar{\lambda}(1-z)\dot{x}_0 D_0^{-1} D(z) A(z)\,[zI - A(z)]^{-1}\,[D(z)]^{-1} \tag{7}$$

In (7),

$$D(z) = \begin{bmatrix} -\gamma_1 - \lambda_1(1 - P_1(z)) & \gamma_1 \\ \gamma_2 & -\gamma_2 - \lambda_2(1 - P_2(z)) \end{bmatrix} \tag{8}$$

is the Z-transform of D_k, $k = 0, 1, 2, \ldots$, where

$$D_0 = \begin{bmatrix} -(\lambda_1 + \gamma_1) & \gamma_1 \\ \gamma_2 & -(\lambda_2 + \gamma_2) \end{bmatrix} \text{ and } D_k = \begin{bmatrix} \lambda_1 P_{1k} & 0 \\ 0 & \lambda_2 P_{2k} \end{bmatrix},\ k = 1, 2, 3, \ldots \tag{9}$$

$P_i(z)$ is the Z-transform of P_{ik}'s defined in (1); $\bar{\lambda} = \lambda_1 q_1 + \lambda_2 q_2$ is the average arrival rate of the BMMPP; and $A(z) = e^{D(z)T}$ (see equation (13) of [11]), where T is the constant service time, is the Z-transform of A_n's.

Using (7) with the algorithm recently introduced by Choudhury and Lucantoni [6], we can compute μ_n, the nth moment of the queue length distribution, as follows:

$$\mu_n = \frac{n!}{2n\xi r_n^n f_n^n} \{ W_n(r_n) + (-1)^n W_n(-r_n) + 2 \sum_{k=1}^{n\xi - 1} \mathrm{Re}\,[W_n(r_n e^{k\pi i/n\xi}) e^{-k\pi i/\xi}] \} \tag{10}$$

where

- $i = \sqrt{-1}$
- ξ is either 1 or 2 (the choice for ξ is explained in [6])
- $r_n = 10^{-5.5/n\xi}$ (for an accuracy up to 16-digit place)
- $f_n = (n-1)\,\mu_{n-2}/\mu_{n-1}$ (adaptively changed for μ_1 and μ_2, see [6] for details)
- $W_n(z) = Y(e^{f_n z})$

We then apply two theorems introduced in [6] (page 18-20) by Choudhury and Lucantoni to $\tilde{y}_n$'s:

Theorem 1: For some positive real constants $\sigma < 1$ and C, if $\tilde{y}_n \sigma^{-n} \to C$ as $n \to \infty$, then $C_n = (\mu_n [\ln(1/\sigma)]^{n+1}) / n! \to C$ as $n \to \infty$

Theorem 2: For some positive real constants $\sigma < 1$ and C, if $\tilde{y}_n \sigma^{-n} \to C$ as $n \to \infty$, then $\sigma_n = e^{(-n\mu_{n-1})/\mu_n} \to \sigma$ as $n \to \infty$

to find some integer κ, such that σ_n and C_n are constants (up to a certain degree of accuracy), for all $n \geq \kappa$. Thus,

$$\tilde{y}_n \to C_\kappa \sigma_\kappa^n, \text{ as } n \to \infty \tag{11}$$

Let $t_n = Pr\{\text{the queue length} \geq n\} = \sum_{i=n}^{\infty} \tilde{y}_i$, i.e., the tail probabilities of the queue length distribution. From (11), we then get:

$$t_n \to \sum_{i=n}^{\infty} C_\kappa \sigma_\kappa^i = C_\kappa \left(\sum_{i=0}^{\infty} \sigma_\kappa^i - \sum_{i=0}^{n-1} \sigma_\kappa^i \right) = C_\kappa \left(\frac{\sigma_\kappa^n}{1 - \sigma_\kappa} \right), \text{ as } n \to \infty \tag{12}$$

4 Numerical Results

We present some numerical results in this section. In 4.1, results for voice and data integration obtained from the proposed approximation are compared to that of simulation as well as that of [1] with batch arrivals extension. The approximation of the integration of video, voice and data is presented in 4.2 using simulation to verify its accuracy.

4.1 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 correspondingly (as concluded by [4]). Aggregated data traffic is assumed to have a packet arrival rate of $20N_d/3$ packets/sec, where N_d (a user-defined parameter) is the number of data calls. We assume that the packet size is geometrically distributed with an average of 5 cells/packet. We assume that the system has a total of 20 voice calls and 50 data calls.

In Fig. 3 through Fig. 6, we 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 the previous section. The dot-dashed lines are the tail probability and its asymptote using the approximation model presented in [1] extended to allow batch arrivals (denoted BMMPP-1). Note that in Fig. 3, the dot-dashed lines overlap the dashed lines.

As can be seen from these figures, both approximations perform very well for a moderate system load (defined to be $\rho = (\frac{\omega\alpha N}{\alpha+\beta} + \mu\bar{s})/C$ where C denotes the capacity of the system), 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 approximate model presented in [1] diverge from the simulation for large buffer positions, while the proposed model stays within a reasonably close range.

Fig. 7 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, it clearly shows that the proposed approximation method outperforms the one presented in [1] for all ranges of system loads.

We use Fig. 8 to demonstrate that tail probabilities, which are generally used as an estimation of loss probabilities (see, for examples, [8] and [10]), 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 done on 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.

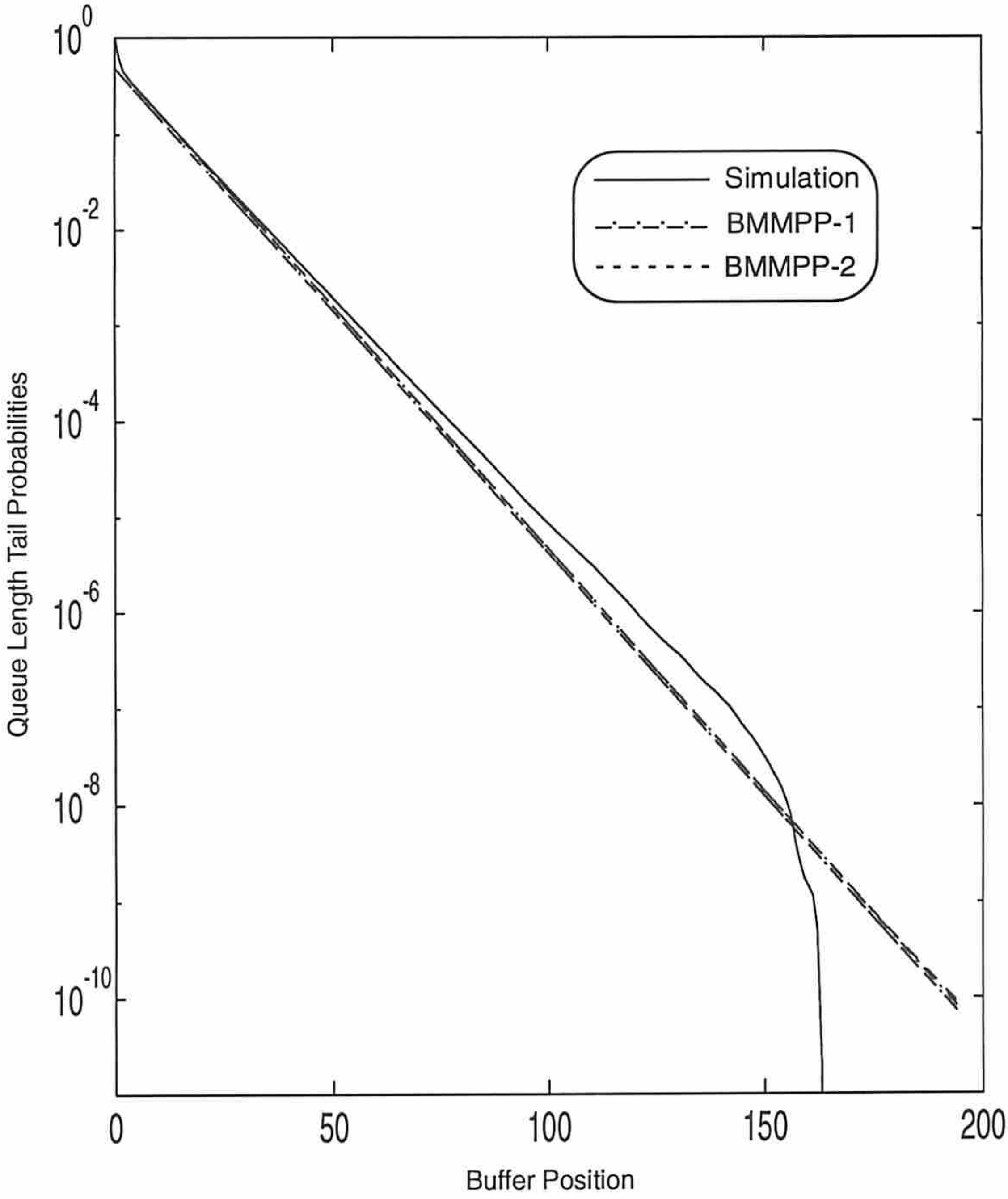

Fig. 3 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.6.

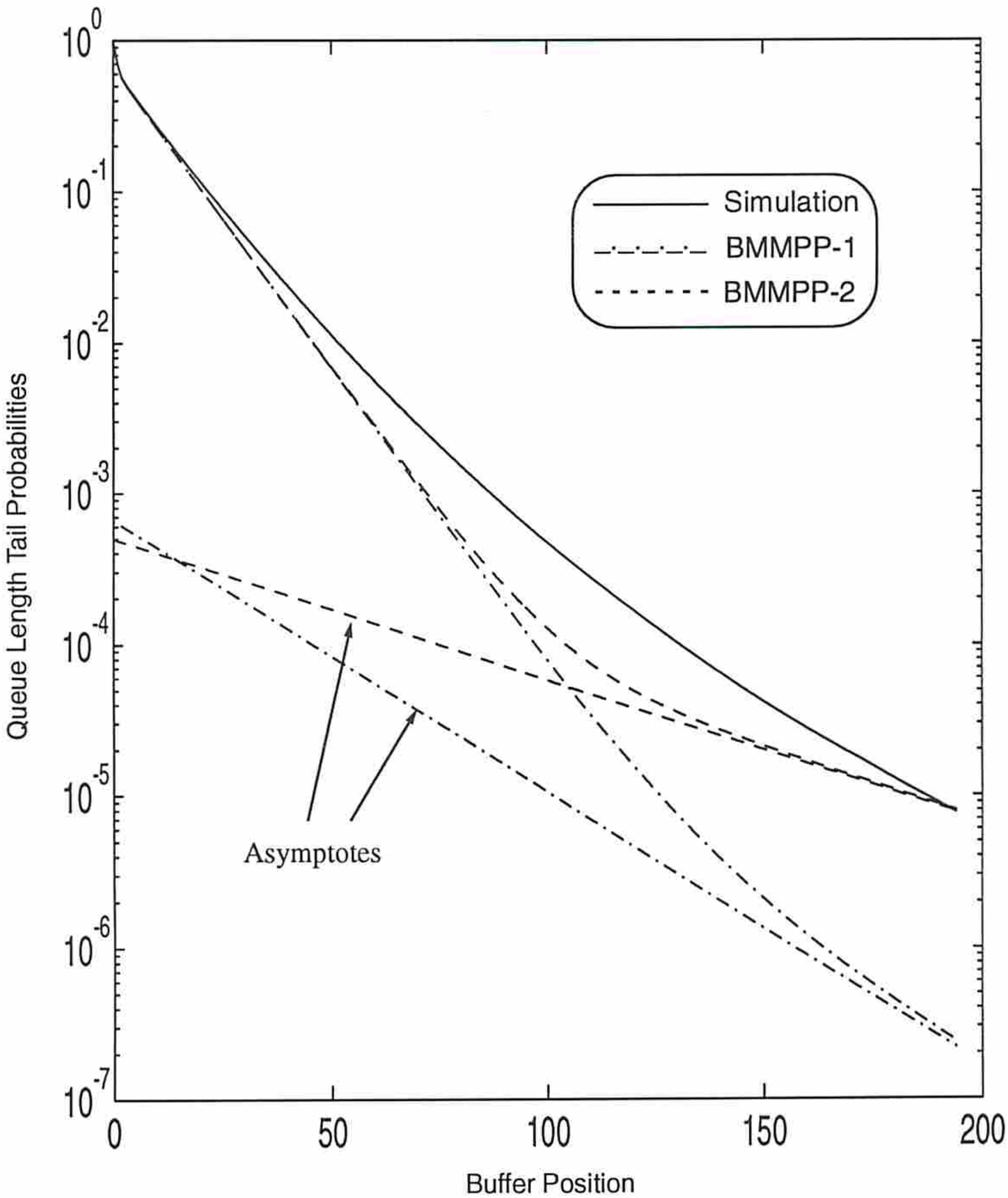

Fig. 4 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.7.

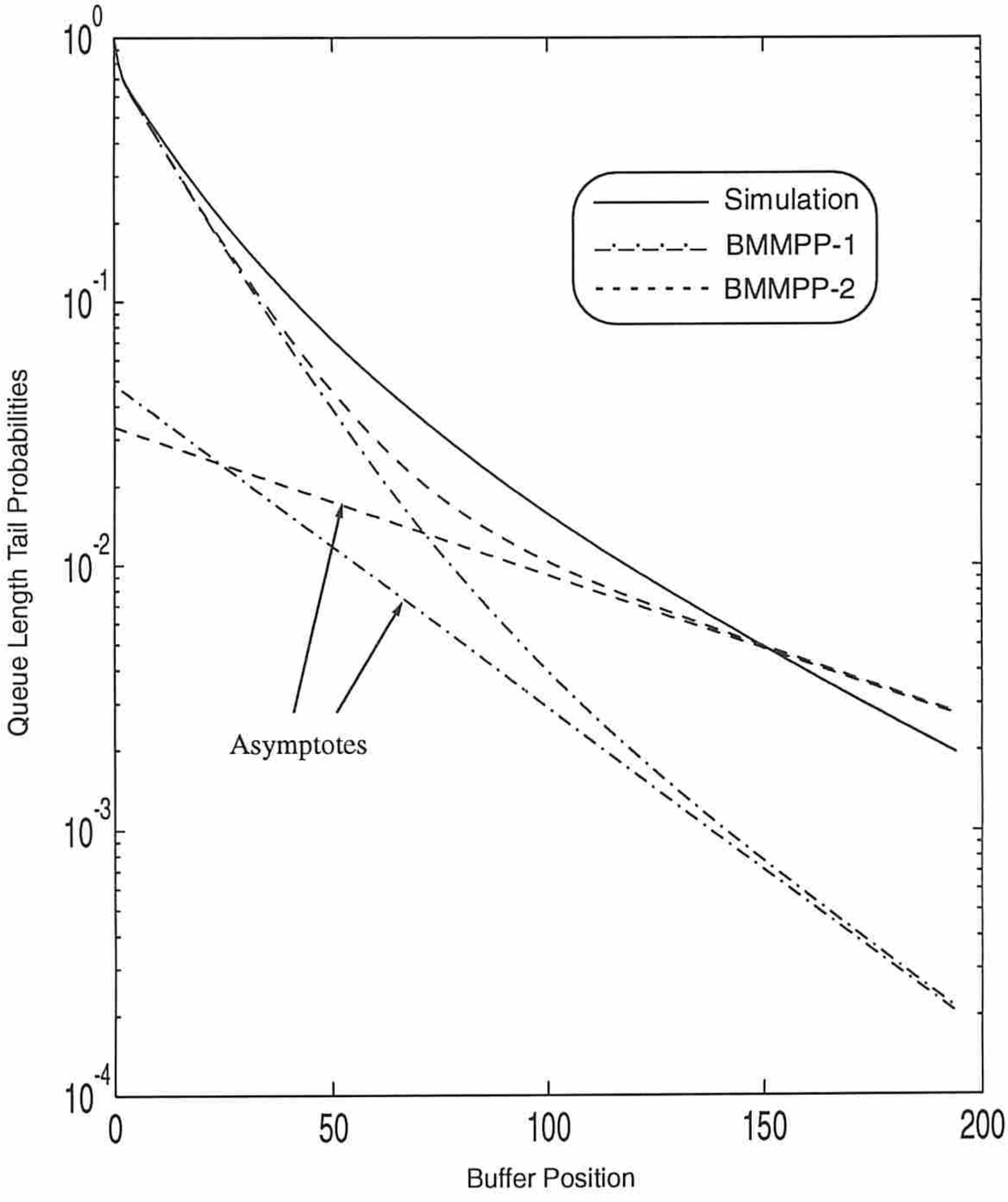

Fig. 5 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.8.

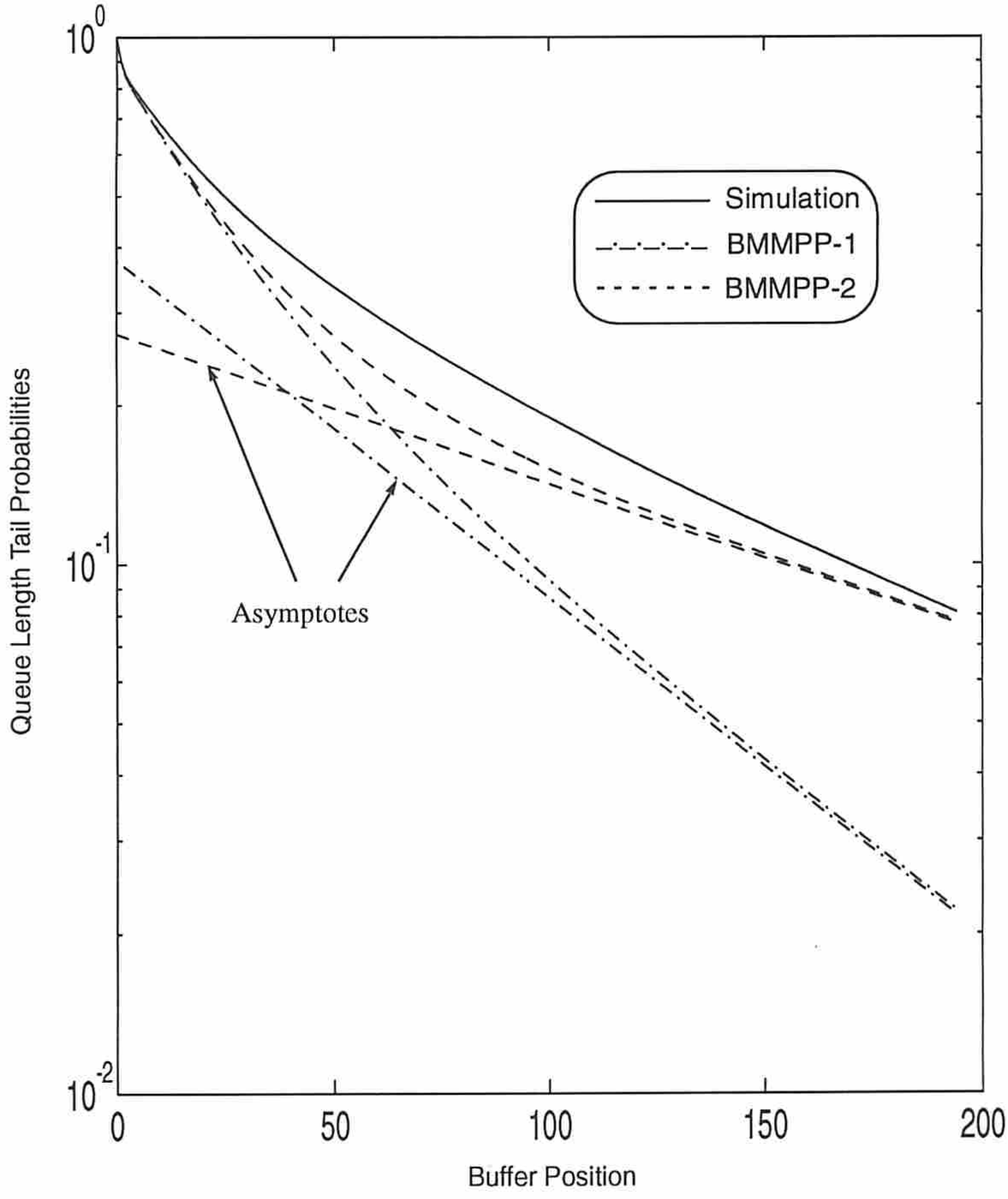

Fig. 6 Survivor function and its asymptote for 20 voice calls and 50 data calls with a system utilization of 0.9.

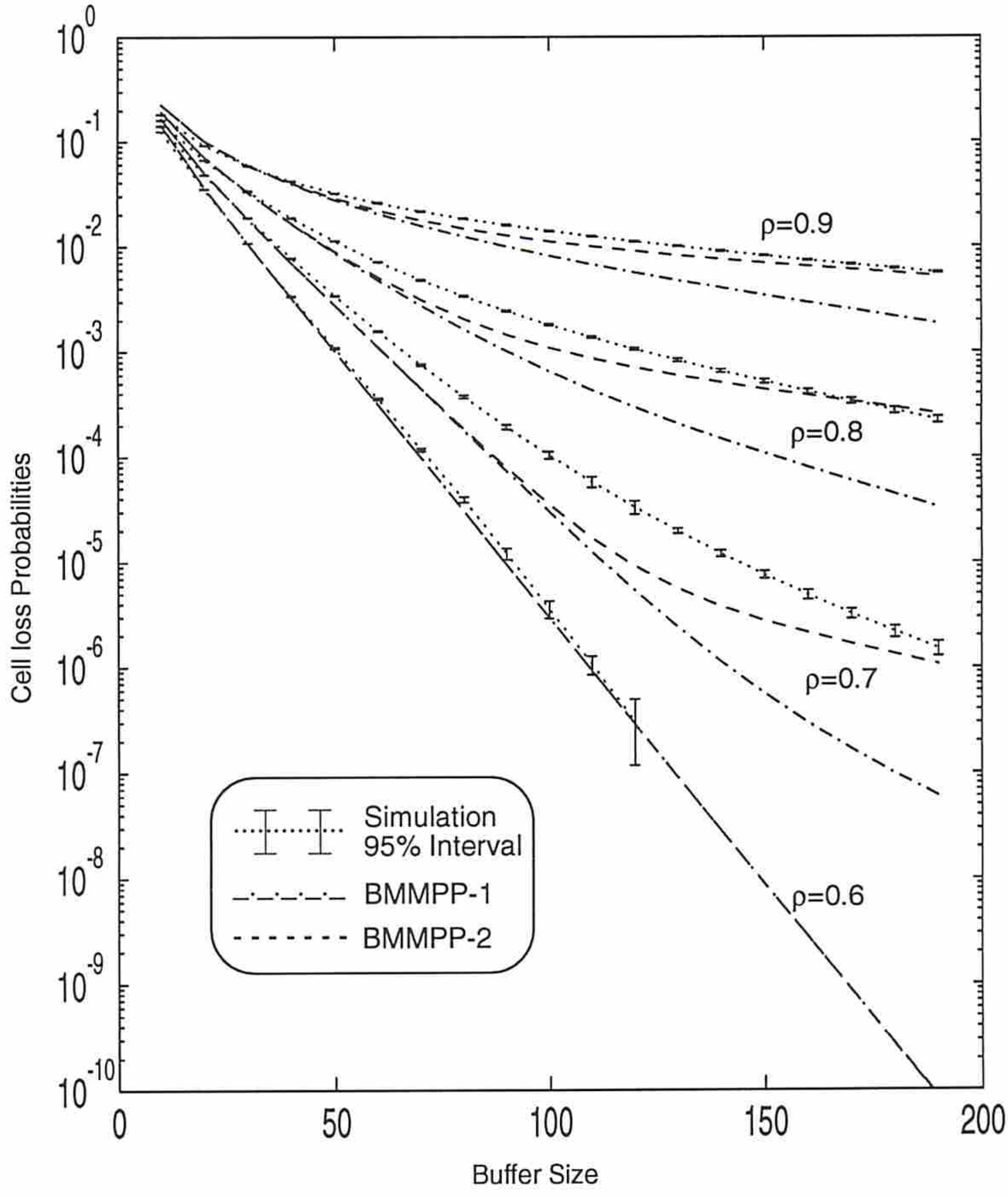

Fig. 7 Cell loss probabilities for 20 voice calls and 50 data calls with different system loads.

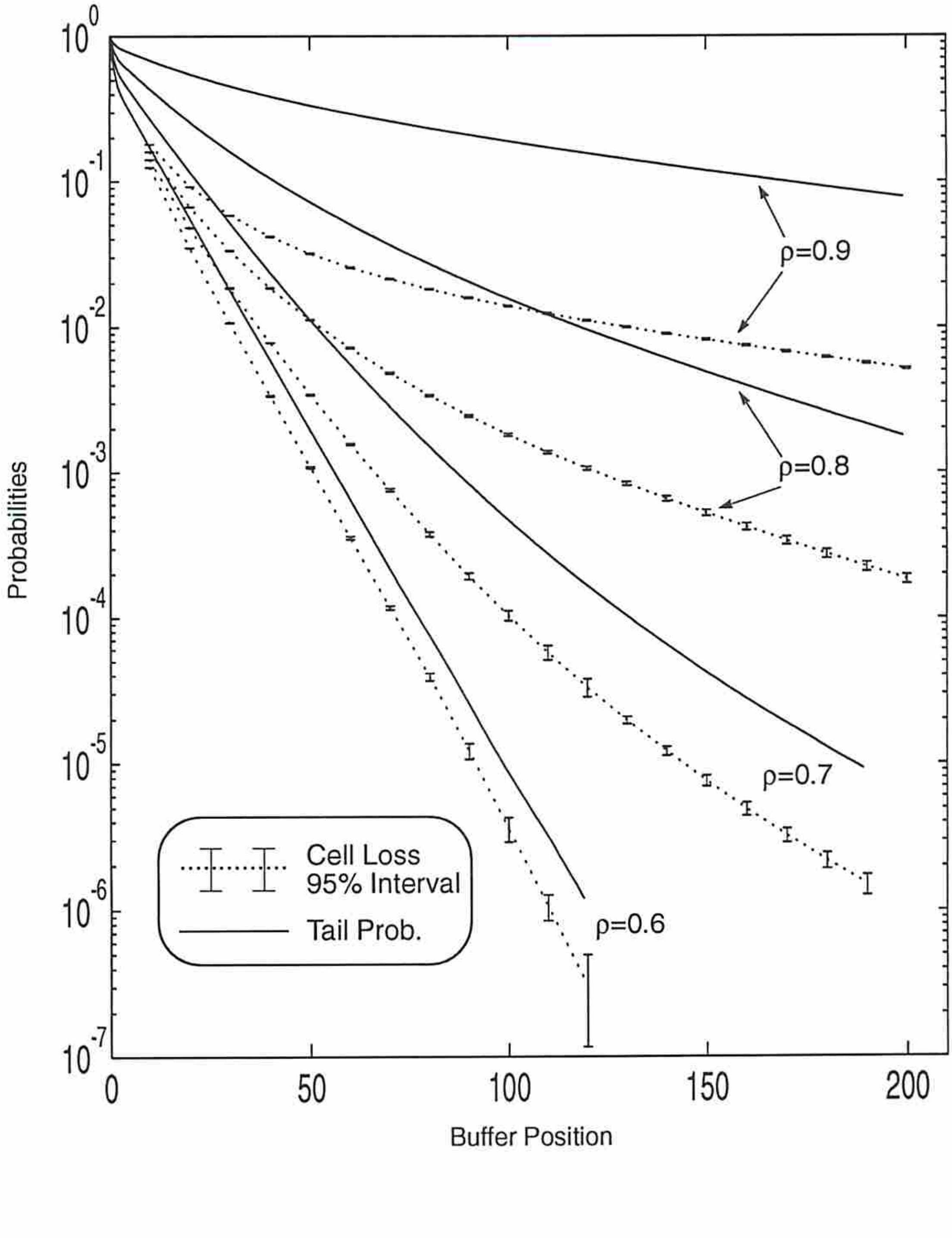

Fig. 8 Tail probabilities versus loss probabilities for 20 voice calls and 50 data calls (simulation results).

4.2 Video, voice and data integration

We use the same voice and data sources as in the previous examples and use the same set of parameters for the video sources as used by [3], [12] and [17], 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 16 times the number of video sources (as suggested by [12]).

Fig. 9 shows the approximation results on the tail probabilities and its asymptote for different system loads (defined to be $\rho = (\frac{\eta a M}{a+b} + \frac{\omega \alpha N}{\alpha+\beta} + \mu \bar{s}) / C)$. 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. 10.

We use Fig. 11 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. 12 is used to show the difference between the tail probabilities and cell loss probabilities. We assume the same number of sources as in Fig. 9 and Fig. 10. This plot confirms again that using tail probabilities is not a good estimate of the corresponding cell loss probabilities, especially for heavy system loads.

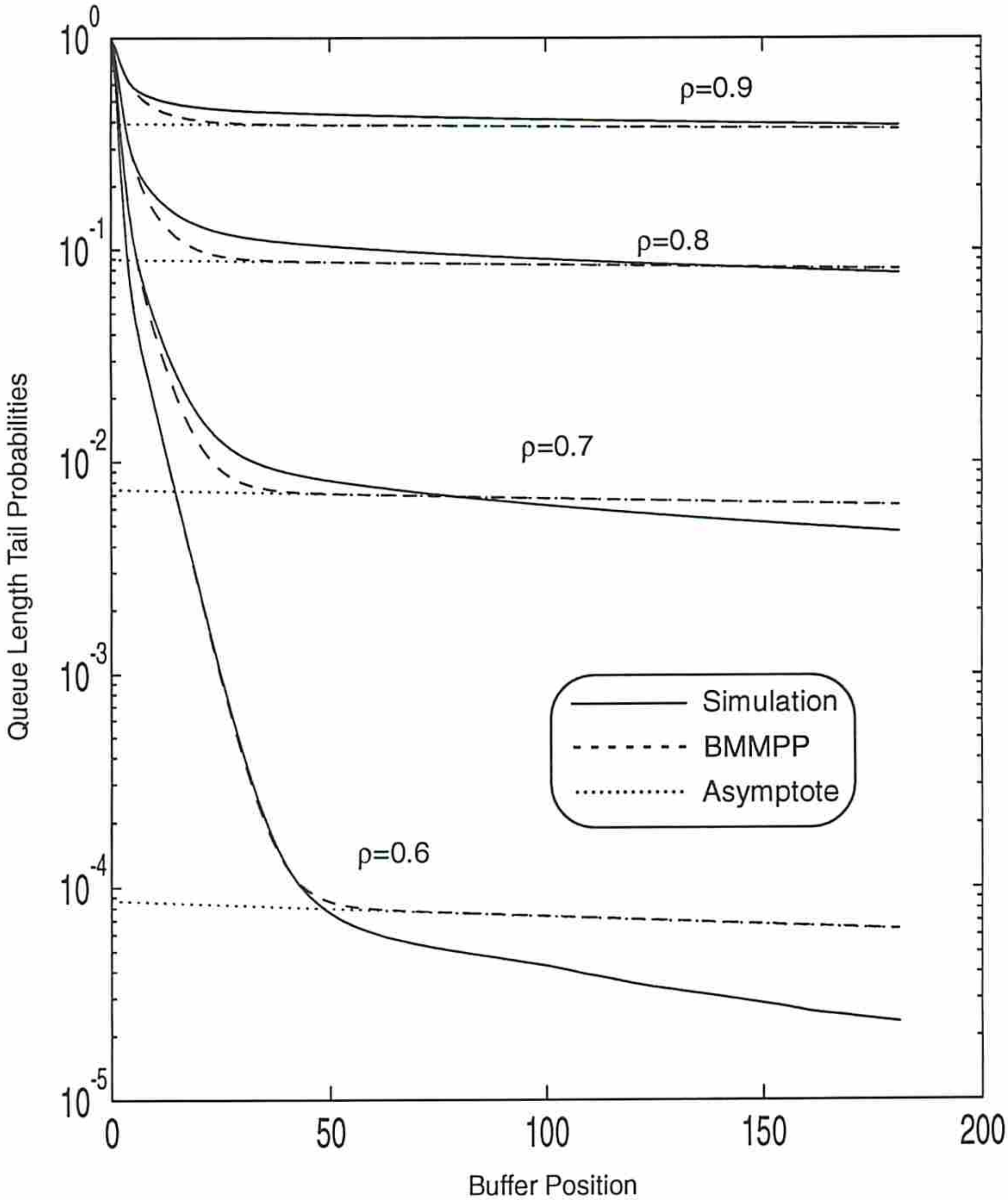

Fig. 9 Survivor function and its asymptote for 5 video calls, 100 voice calls and 100 data calls for different system loads.

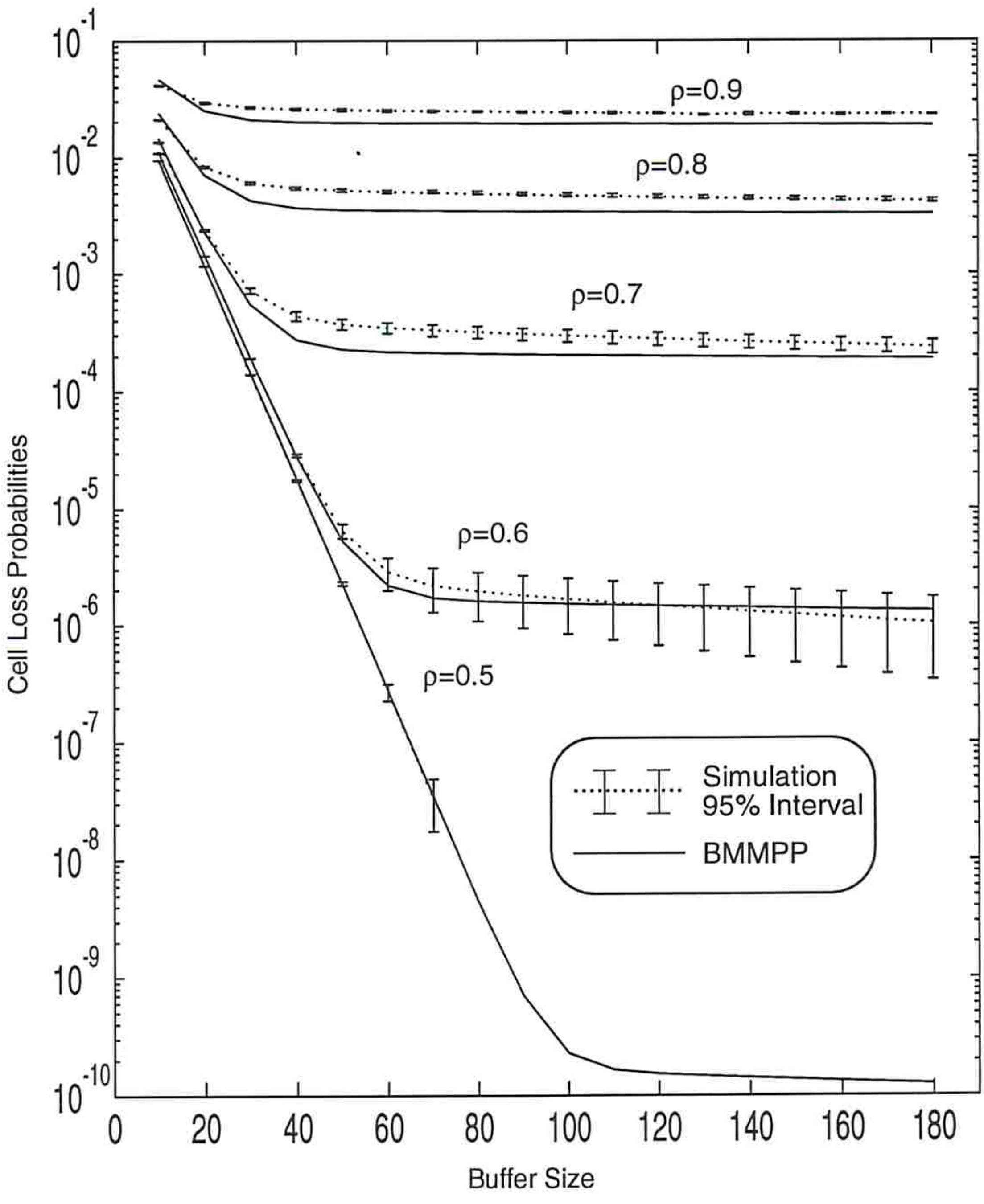

Fig. 10 Cell loss probabilities for 5 video calls, 100 voice calls and 100 data call with different system loads.

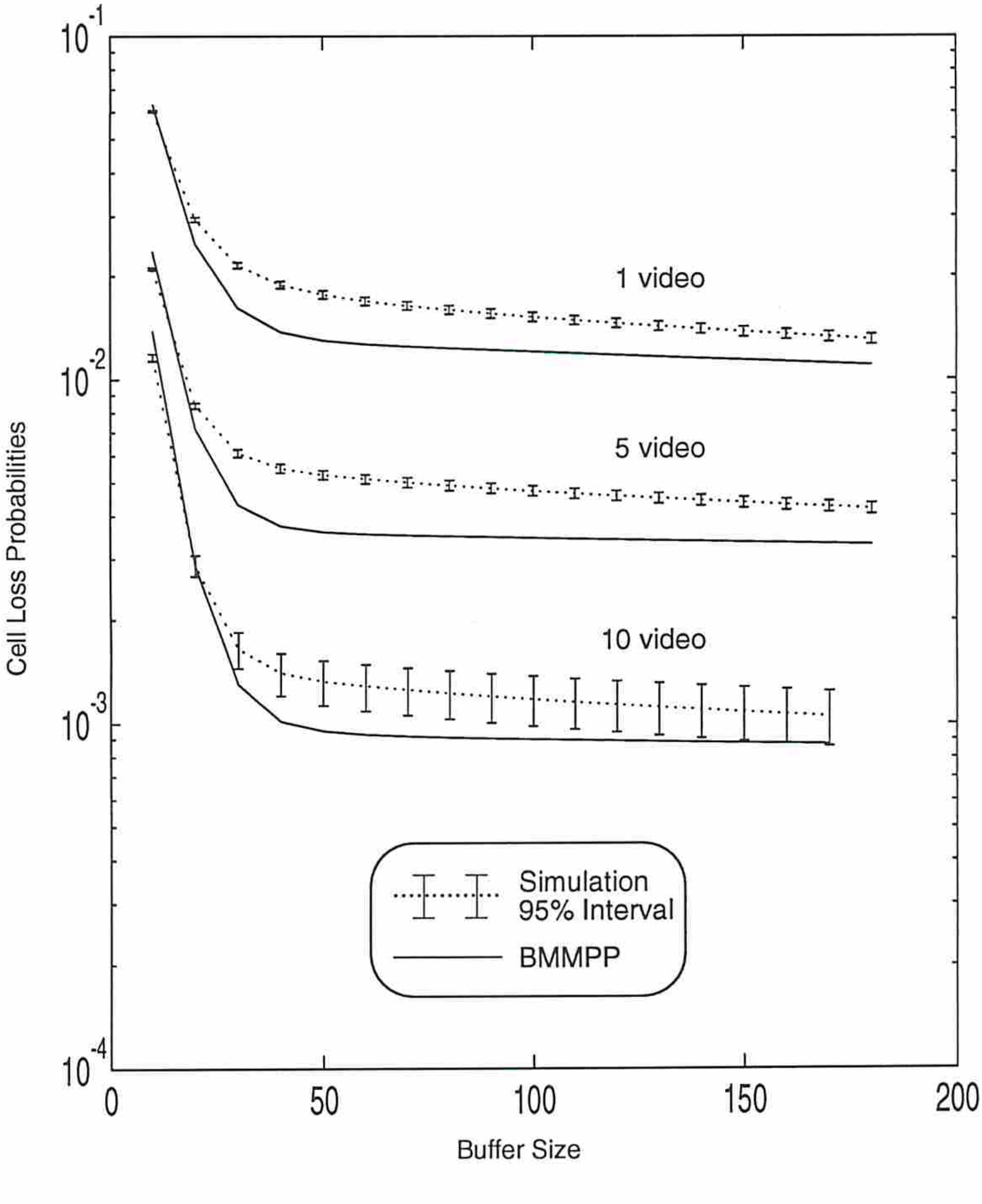

Fig. 11 Cell loss probabilities for a system load of 0.8 and 100 voice calls and 100 data calls with different number of video calls.

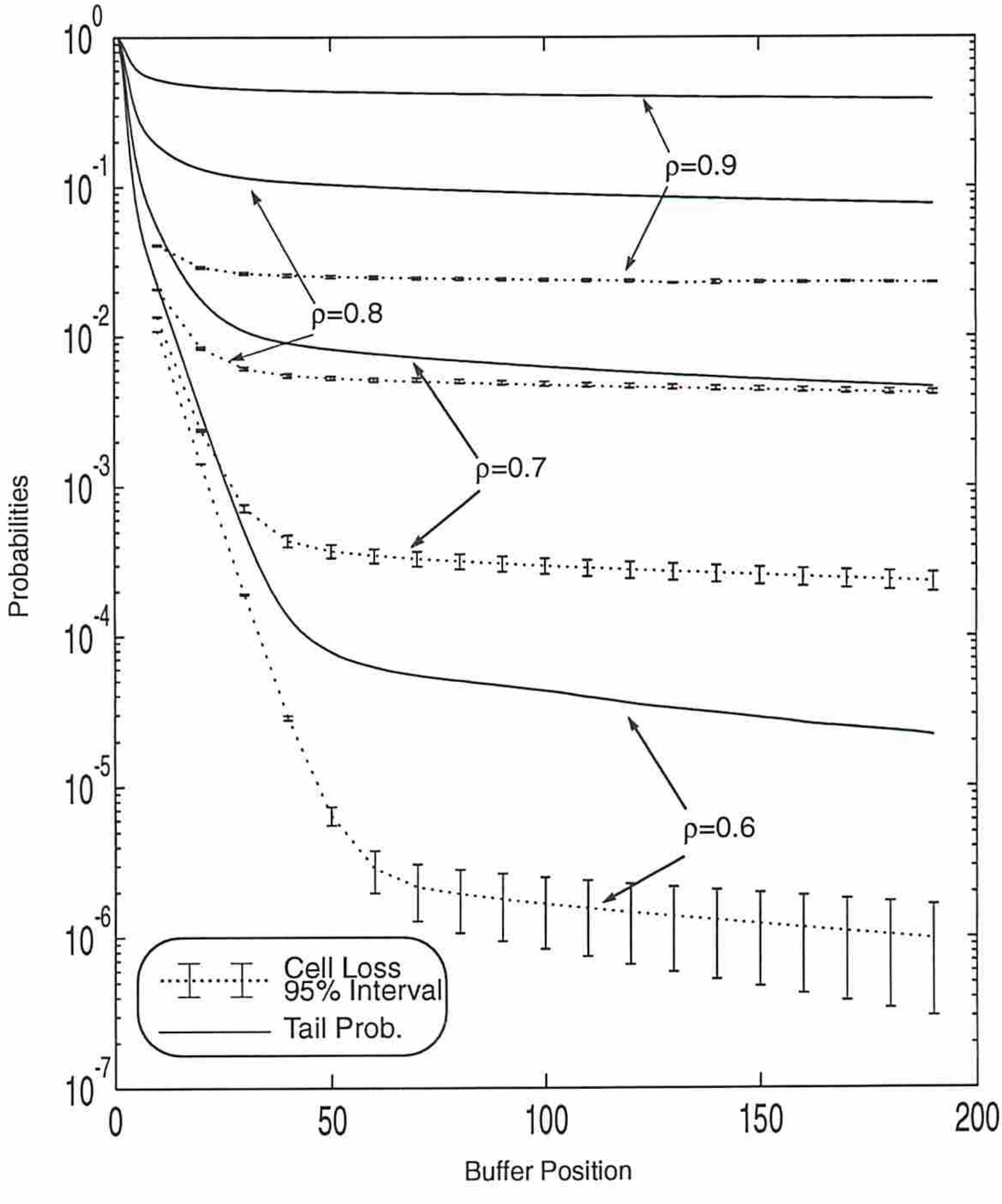

Fig. 12 Tail probabilities versus loss probabilities for 5 video calls, 100 voice calls and 100 data calls (simulation results).

5 Conclusions

In this paper, we have studied the performance of an ATM multiplexer loaded with video, voice and data traffic. We use a two-state BMMPP as an approximation for the integration of voice and data sources and extend this approach to handle video sources also (reducing from an $(M+1)(N+1)$-state arrival process to a two-state one). We then model the multiplexer as a two-state BMMPP/D/1/K queue to study its loss performance.

The size of 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. 10 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).

We carry out the analysis of the cell loss probabilities of the proposed approximation. We also study the tail probabilities and present an algorithm to find its asymptote for the queue length distribution for the approximate model. The analytical results, which are presented in a form specifically for two-state BMMPP/D/1/K queues, can be easily extended to a general (with any number of phases) BMMPP/D/1/K queue by simply changing the size of the vectors and matrices.

One of the contributions of this research 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 (11), instead of solving the system explicitly. This indeed provides a very good approximation as shown by Fig. 3 through Fig. 6 and Fig. 9.

As demonstrated by numerical examples, the proposed approximation performs much better than the one used by Baiocchi *at al.* in [1], especially for larger buffer sizes that we have studied. It also functions as a reasonably good approximation for estimating the cell loss probabilities. Another conclusion drawn from Fig. 8 and Fig. 12 is that in the environment under study one cannot directly estimate loss probability from the corresponding tail probability of an infinite queue.

Appendix

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 [2]) 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:

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

References

[1] 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

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

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

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

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

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

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

[8] 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.

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

[10] 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.

[11] 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.

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

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

[14] 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.

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

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

[17] 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.

[18] S. S. Wang and J. A. Silvester, "A Fast Performance Model for Real-Time Multimedia Communications," in *Proc. Fifth International Conference on Data Communication Systems and Their Performance: High Speed Networks*, 1993.