Submitted to the *International Conference on Communications, ICC '96*

EFFICIENT SIMULATION OF CELL LOSS PROBABILITY IN ATM NETWORKS WITH HETEROGENEOUS CONNECTION TRAFFIC DESCRIPTORS

JAMES A. FREEBERSYSER¹ J. KEITH TOWNSEND²

Center for Advanced Computing and Communication
Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, North Carolina 27695-7914

ABSTRACT

In ATM networks, the purpose of connection admission control is to decide if allowing a new connection into the network violates a quality of service measure of the new connection or the existing connections. Instead of using statistical traffic models, we present the connection entering the network by the connection traffic descriptors standardized by the ATM Forum and used by the connection admission control algorithms. In this paper, we examine traffic sources described by heterogeneous connection traffic descriptors in order to analyze constant bit rate sources versus variable bit rate sources, as well as different classes of variable bit rate sources. We use Monte Carlo simulation to estimate the cell loss probability of the combined traffic stream, as well as the individual classes using both random buffer filling and priority buffer filling schemes. Importance sampling is used to increase the efficiency of the simulation. For the experimental examples considered here, the improvement in simulation efficiency compared to conventional Monte Carlo simulation is inversely proportional to the estimate. The efficient simulation method developed here is suitable for the design and testing of the switches and connection admission control algorithms planned for use in ATM networks.³

¹James A. Freebersyser is supported in part by the U.S. Air Force Palace Knight Program.

²The authors would like to thank Brad Makrucki of BellSouth for suggesting this problem and his helpful feedback while researching this topic, as well as Ahmet A. Akyamac of the Center for Advanced Computing and Communication at North Carolina State University for his helpful discussions of this material and thorough review of this paper.

³This work was supported in part by the Center for Advanced Computing and Communication, North Carolina State University.

1 Introduction

The forthcoming Broadband-Integrated Services Digital Network (B-ISDN) has provoked great interest in the modeling and performance analysis of Asynchronous Transfer Mode (ATM) switches and networks. One of the many problems that faces the designers of ATM networks is how to guarantee that each customer will receive the agreed upon Quality of Service (QoS). The purpose of Connection Admission Control (CAC) algorithms in ATM networks is to decide in real time whether a new connection can be admitted into the network without violating QoS measures, such as the cell loss probability, for the new connection or the existing connections. The decisions made by the CAC algorithm in implementing a given CAC policy are specific to the ATM switch design, and are also used in the negotiation of the connection traffic descriptors and QoS measures that result in the traffic contract between the user and the network [1].

Simple, approximate CAC algorithms must be used in actual ATM networks to perform, in real time, the negotiation and decision functions. However, some method of testing CAC algorithms is necessary to verify that the resulting CAC decisions are correct, or at least conservative, since the network must guarantee the QoS. Closed form solutions for QoS measures such as the cell loss probability are often unavailable by analytical methods for realistic network conditions because of the high dimensionality that results from the complexity of the ATM switches and traffic, in particular the large buffer sizes and the variety of multimedia traffic expected in ATM networks.

Monte Carlo (MC) simulation can be used to obtain accurate estimates of performance measures of complex ATM networks, but is too slow to be used for CAC in real time. Even in non-real time environments, such as CAC algorithm testing, MC simulation can be too slow for estimating the rare events associated with QoS measures (for example, cell loss probabilities in the range of 10^{-6} to 10^{-12}) because of the long run times required to obtain accurate estimates. Importance Sampling (IS) is a technique that, under the proper conditions, can speed up simulations involving rare events of network (queueing) systems (see for example, [2]-[4]). By carefully choosing the modification or bias of the underlying probability measures, large speed-up factors in simulation run time can be obtained.

Most commonly, statistical traffic models have been used in the analysis and simulation of ATM switches and networks. Several generations of statistical traffic models have been developed as more insight has been gained into the burstiness and correlation of ATM traffic. These statistical traffic models have become increasingly accurate, but they have also become more complicated and therefore less tractable to analysis. In addition, it is difficult to establish a relationship between the parameters of the statistical models of the ATM traffic and the connection traffic descriptors used by the CAC algorithm. [5]

We overcome this problem by characterizing the traffic entering the ATM network by the connection traffic descriptors standardized by the ATM Forum. Using traffic descriptors in the analysis of a performance measure was first described as the operational approach in [6]. Several studies have attempted to determine the cell loss probability based on various traffic descriptors [7]-[13]. All of these analysis based studies for determining the cell loss probability using the operational approach have drawbacks. Either low cell loss probabilities were not obtained, or restrictions on the system parameters were made in order to arrive at approximations or upper bounds of the cell loss probability, and only [9] considered heterogeneous connection traffic descriptors. Jabbari et al in [14] present a simulation method based on the operational approach, but did not provide any results for low cell loss probabilities.

Previously in [15], we presented a simulation method using the ATM Forum standardized connection traffic descriptors to characterize the traffic entering the network. We used a multinomial formulation for the Monte Carlo simulation to remove the correlation associated with estimating bursty events, and showed how IS is applied, using the method of interval reduction, to the problem of estimating the cell loss probability due to buffer overflow for traffic sources described by homogeneous connection traffic descriptors.

In this paper, we examine traffic sources described by heterogeneous connection traffic descriptors in order to examine the interaction between constant bit rate (CBR) connections and variable bit rate (VBR) connections, as well as the interaction between different classes of VBR connections. We use MC simulation to estimate the cell loss probability of the combined traffic streams, as well as the cell loss probability of the individual classes, in both random buffer filling and priority buffer filling schemes. IS is used to increase the efficiency

of the MC simulation based on an adaptation of the method of interval reduction [15]. The improvement in simulation efficiency is a result of dramatically increasing the number of samples drawn from the important region. Compared to the conventional MC simulation used to estimate the cell loss probability, the improvement in simulation efficiency observed is inversely proportional to the estimate. Our method of efficient simulation is applicable to the design and testing of ATM switches, as well as the CAC algorithms planned for use in ATM networks.

2 System Description

We first discuss the connection traffic descriptors standardized by the ATM Forum that will be used to characterize the traffic entering the network. We then examine the ATM switch considered and the resulting simulation model.

2.1 Connection Traffic Descriptors

We characterize a connection requesting admission into the ATM network by the following connection traffic descriptors standardized by the ATM Forum UNI Specification [1]:

$$\hat{\lambda}, \text{ peak cell rate (PCR) in bits per second (bps)} \quad (1)$$

$$\bar{\lambda}, \text{ sustained (average) cell rate (SCR) in bps} \quad (2)$$

$$\hat{B}, \text{ maximum burst length in cells at the peak cell rate} \quad (3)$$

The $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplet contains optional connection traffic descriptors which can be used to derive the remaining required connection traffic descriptors, specifically the cell delay variation and burst tolerance, that define the traffic contract at the output of the Usage Parameter Control (UPC) device at the User-Network Interface (UNI) [1]. Each connection, which may be an individual source or multiplexed sources, negotiates with the CAC algorithm for admission into the network using the $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplet to describe the connection traffic.

Figure 1: ATM switch model.

Figure 2: Connection admission control block diagram.

2.2 ATM Switch

The ATM switch considered here is shown in Figure 1. The ATM switch has N_P input ports, and each of the N_C connections in the network routed by the non-blocking switch matrix to the output cell buffer of one of the N_P output ports is described by a $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplet. An example of a non-blocking switch matrix is a shared bus architecture where the shared bus operates at a speed of $N_P \hat{\lambda}$ so that all N_P input ports are given access to the shared bus. The buffers use a first-in, first-out (FIFO) queueing discipline. Each buffer has a finite size of K cells and cells exit the buffer at a service rate of $\hat{\mu}$ bps.

2.3 Simulation Model

As shown in Figure 2, the CAC algorithm uses knowledge of the $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplets of the

new connection and the existing N_C connections, as well as the switch parameters $(\hat{\mu}, K)$ to decide if the new connection should be allowed into the network. Our goal is to determine the cell loss probability due to output cell buffer overflow resulting from the worst-case UPC compliant traffic pattern for the combined traffic stream, and for each class of connection traffic descriptors in both random buffer filling and priority buffer filling schemes as a function of the number of classes and connections in the network and the switch parameters.

The $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplet defines the worst-case traffic that the source can input to the network. ATM policing measures such as the leaky-bucket algorithm will insure this is so. The worst-case traffic pattern which is still UPC compliant is when the traffic enters the network at the peak rate, $\hat{\lambda}$, such that the maximum burst length of $\hat{B}$ cells is generated. The CAC algorithm must consider the worst-case traffic pattern in determining whether or not a QoS measure for cell loss probability is violated by admitting a new connection into the network, and the switch designer must consider this pattern in order to properly dimension the buffer size in the switch. Of course, due to the statistical nature of the input traffic, a source may at times input traffic to the network that is less than the worst-case. But the negotiated $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplet is what the connection is “paying for” and therefore, the network must guarantee the QoS assuming traffic at the worst-case level. Thus, the key feature of this system model is that the traffic source is derived from the standardized connection traffic descriptors planned for use in ATM networks [1], rather than a stochastic model for the traffic source.

In this paper, we examine traffic sources described by one of two classes of connection traffic descriptors. Since peak cell rate allocation can be done for the case of multiple classes of CBR sources, the classes of traffic will be either CBR/VBR or VBR/VBR. We also assume that the VBR classes are sufficiently dissimilar so that the different classes can not be lumped together to form a single homogeneous class, a case which has already been analyzed in [15].

We operate with a slotted-time simulation model where an arrival slot is the amount of time that one cell takes to arrive at the maximum peak rate over all the connections routed through the same buffer. Because of the discrete time nature of the simulation, the peak cell rates of the different classes of traffic source must be integer factors of the peak cell rate,

Figure 3: Examples of traffic generated by CBR and VBR sources.

thus we examine only combinations of connection traffic descriptors such that the maximum peak cell rate is an integer multiple of all of the peak cell rates, $\hat{\lambda}_{\max} = n\hat{\lambda}_l$, where n is a positive integer and l indexes the class of traffic.

The period of the on/off cycle of the l -th of the S classes of arriving traffic, T_l , is the time in arrival slots that the $\hat{B}_l$ length cell burst at the peak rate $\hat{\lambda}_l$ must be averaged over so that the sustained rate, $\bar{\lambda}_l$, is not exceeded, thus

$$T_l = \hat{B}_l \hat{\lambda}_l / \bar{\lambda}_l, \text{ arrival slots} \quad (4)$$

For CBR traffic, where $\hat{\lambda}_l = \bar{\lambda}_l$, the period is defined to be $T_{\text{CBR}} = 1$. The overall period, or super-period, in the simulation model is $T_{SP} = \text{lcm}(T_1, T_2, \dots, T_S)$ where lcm is the lowest common multiplier operator.

Traffic examples of sources with three different $(\hat{\lambda}, \bar{\lambda}, \hat{B})$ -triplets are shown in Figure 3. A VBR source with a burst of length $\hat{B} = 4$ cells is shown with the dark grey cells. Shown with the light grey cells is another VBR source with the same average cell rate, but half the peak cell rate of the first VBR source. Since the arrival slot length is a result of the maximum peak cell rate, the cell bursts of the slower peak cell rate are not contiguous, thus the cells in the burst of length $\hat{B} = 2$ cells of this source do not occupy adjacent slots. In addition, two periods of this source fall within the super-period defined by the first VBR source. The plaid cells depict a CBR source with a peak cell rate one quarter of the first VBR source.

We assume uniform traffic routing through each of the N_P buffers, so that by examining one buffer we obtain the performance characteristic of all of the N_P buffers. We also assume that the buffer is initially empty, since the steady-state behavior is equivalent whether it

is reached by starting $N + 1$ connections with an empty buffer, or by adding an additional connection to the existing N connections and a non-empty buffer.

We let the number of ports on the ATM switch be equal to the number of connections routed through the buffer, $N_P = N_C$, since this condition results in an upper bound of the cell loss probability for any size switch, in terms of the number of input and output ports. In the example of the shared bus architecture, the data rate of the shared bus is then N_C cells per arrival slot, so that all N_C input ports have access to the shared bus in each arrival slot, which results in N_C service slots per arrival slot. When $N < N_C$ input ports are active, we make the worst-case assumption that the cells of the active ports occupy the first N of N_C service slots. In the random buffer filling scheme, we assume that each connection occupies every service slot position with equal probability for each buffer. In the priority buffer filling scheme, the cells of one class enter the buffer before the cells of other classes in each arrival slot.

The effective service rate, μ , of the server is

$$\mu = \hat{\mu} / \hat{\lambda}_{\max}, \text{ cells/arrival slot} \quad (5)$$

In each service slot, one cell can be loaded into the buffer if there are arriving cells and μ/N_C cells can be removed from the buffer if the buffer contains greater than μ/N_C cells. If the buffer contains less than μ/N_C cells, the remaining cells in the buffer are serviced. Cells in the buffer are removed according to this policy even if there are no cell arrivals. Cell losses occur when there is no space in the buffer for the arriving cells.

The randomness in the simulation model is found in the connection starting-slot position. A connection can start in any arrival slot within the T_l length period, so we describe the connection starting-slot position with a uniform probability density. Once the connection starting-slot positions of the N_C connections are known, the system model behaves deterministically. We need only observe the buffer size for three super-periods to determine whether or not a cell loss occurs. The onset of the steady-state in the system occurs when the buffer behavior is repeated over consecutive super-periods.

2.4 Deterministic Results

The maximum number of cell losses in a period, $L_{\max}$, corresponds to the case where all the $N_C = \sum_{l=1}^S N_{C_l}$ connections start in the same arrival slot. We define this case as the aligned at zero (AAZ) case. An upper bound for the maximum number of cell losses that can occur in a steady-state super-period, $\overline{L_{\max}}$, is then

$$\overline{L_{\max}}(N_C) = \lceil \left(\sum_{l=1}^S \lceil \frac{N_{C_l} \hat{B}_l}{\hat{\lambda}_{\max} / \hat{\lambda}_l} \rceil \right) - \hat{B}(\hat{\lambda}_{\max}) \mu - K \rceil + 1 \quad (6)$$

where $\lceil x \rceil$ is the ceiling operation that rounds x up to the next highest integer if x is not an integer. The exact value of $L_{\max}$ must be obtained by simulation of the AAZ case.

Since the AAZ case results in the maximum number of cell losses, the minimum number of connections, $N_{C_0} = \sum_{l=1}^S N_{C_l}$, that result in a cell loss requires that

$$\sum_{l=1}^S \lceil \frac{N_{C_l} \hat{B}_l}{\hat{\lambda}_{\max} / \hat{\lambda}_l} \rceil > K + \hat{B}(\hat{\lambda}_{\max}) \mu \quad (7)$$

For sets of connections that do not meet this condition, cell losses at the buffer can never occur.

A cell loss is guaranteed to occur at the buffer *a priori* for a number of connections, $N_{C_g} = \sum_{l=1}^S N_{C_l}$, such that the sum of the average cell rates exceeds the service rate

$$\sum_{l=1}^S N_{C_l} \bar{\lambda}_l > \hat{\mu} \quad (8)$$

The performance region we are interested in is over the set of connections such that $N_{C_0} \leq N_C \leq N_{C_g}$. There are combinations of system parameters where $N_{C_0} \geq N_{C_g}$, such as when $K \gg T_{SP}$, but such combinations result in high cell loss probability and therefore are not of interest.

3 Closed-Form Analysis

Consider the set V' of all unique connection starting-slot vectors $\underline{v}'$. A connection starting-slot vector $\underline{v}'$ is composed of N_C elements, where the j -th element in the vector represents

the arrival slot position, from 1 to T_l arrival slots in a period for the l -th of S classes, where the j -th connection starts its $\hat{B}_l$ length cell burst. Each connection starting-slot vector corresponds to the loss of i cells in a steady-state super-period. Let n_{CL_i} be the total number of connection starting-slot vectors that map to i cell losses in a steady-state super-period, where $|V'| = \sum_{i=0}^{L_{\max}} n_{\text{CL}_i}$ and $L_{\max}$ is the maximum number of cell losses that occurs in a steady-state super-period. The probability of i cell losses in a steady-state super-period is

$$p_i = n_{\text{CL}_i} / |V'| \quad (9)$$

where $\sum_{i=0}^{L_{\max}} p_i = 1$. Since each of the N_{C_l} connections generates a burst of length $\hat{B}_l$ cells, the average number of arrivals per steady-state super-period is

$$\bar{n}_{\text{CA}} = \sum_{l=1}^S N_{C_l} \hat{B}_l \left(\frac{T_{\text{SP}}}{T_l} \right) \quad (10)$$

Define the cell loss ratio (CLR) or cell loss probability, P_{CL} , due to buffer overflow as

$$P_{\text{CL}} = \frac{\text{Total Number of Cell Losses}}{\text{Total Number of Arriving Cells}} = \frac{N_{\text{CL}}}{N_{\text{CA}}} = \frac{\sum_{i=1}^{L_{\max}} i n_{\text{CL}_i}}{|V'| \bar{n}_{\text{CA}}} = \frac{\bar{n}_{\text{CL}}}{\bar{n}_{\text{CA}}} = \sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right) p_i \quad (11)$$

Thus, P_{CL} is a weighted sum of probabilities from the multinomial distribution of cell loss.

Define V as the set of connection starting-slot vectors $\underline{v}$ such that the first connection of the class with the maximum period, $T_{l_{\max}}$, is fixed to start at the beginning of a super-period. Each connection starting-slot vector in V is translationally invariant to $T_{l_{\max}} - 1$ connection starting-slot vectors in V' . The exhaustive method of determining P_{CL} sums the cell losses over every connection starting-slot vector in V' in every arrival slot in a steady-state super-period for each connection.

However, for even small values of T_l and $N_C = \sum_{l=1}^S N_{C_l}$, the exponential order of complexity of the exhaustive method, $O\left(\frac{N_C T_{\text{SP}}}{T_{l_{\max}}} \prod_{l=1}^S T_l^{N_{C_l}}\right)$, quickly becomes unmanageable. So far, this high dimensionality, as well as the non-linearity of the cell loss mechanism, has made this problem intractable to analysis for realistic system parameters without the use of simplifying assumptions. Therefore, we are interested in using MC simulation to estimate P_{CL} .

4 Multinomial Formulation for MC Simulation

Conventional MC simulation estimates the numerator in (11), the average number of cell losses per steady-state period, $\bar{n}_{\text{CL}}$. By estimating $\bar{n}_{\text{CL}}$ and thus P_{CL} we are implicitly estimating the p_i of the multinomial distribution of cell loss. We use the multinomial formulation from [15] where the multinomial indicator function of a cell loss, I_L , maps each of the N_D starting-slot vectors drawn independently from V to L cell losses in a steady-state period for all N_C connections

$$I_L = \begin{cases} 1, & \text{if } L \text{ cell losses occur for } 1 \leq L \leq L_{\max} \\ 0, & \text{if a cell loss does not occur} \end{cases} \quad (12)$$

as

$$\hat{P}_{\text{CL}} = \frac{1}{N_D} \sum_{k=1}^{N_D} \sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right) I_i(\underline{v}(k)) = \sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right) \hat{p}_i \quad (13)$$

The multinomial formulation used here reduces to the more familiar binomial formulation when $L_{\max} = 1$.

Since the $\hat{p}_0$ term dominates the other $\hat{p}_i$ terms but is not included in the summation, each $\hat{p}_i$ is effectively independent, which was verified experimentally for known systems. The variance of the $\hat{p}_i$ in (13) is then closely approximated by

$$\sigma^2(\hat{p}_i) = \frac{p_i(1-p_i)}{N_D} \quad (14)$$

and the variance of $\hat{P}_{\text{CL}}$ is closely approximated by

$$\sigma^2(\hat{P}_{\text{CL}}) = \sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 \sigma^2(\hat{p}_i) = \frac{1}{N_D} \left(\sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 (p_i(1-p_i)) \right) \quad (15)$$

The estimate of the variance of $\hat{p}_i$ is the sample variance

$$\hat{\sigma}^2(\hat{p}_i) = \frac{1}{N_D(N_D-1)} \sum_{k=1}^{N_D} (I_i(\underline{v}(k)) - \hat{p}_i)^2 \quad (16)$$

and the estimate of the variance of $\hat{P}_{\text{CL}}$ is

$$\hat{\sigma}^2(\hat{P}_{\text{CL}}) = \sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 \hat{\sigma}^2(\hat{p}_i) \quad (17)$$

In [16], it was determined that the confidence interval of a weighted sum of a multinomial distribution follows a χ^2 distribution. We state the theorem from [16] with the original notation slightly altered for our application:

Theorem [16]: As $N_D \rightarrow \infty$, the lower limit of the probability that P_{CL} satisfies

$$\hat{P}_{\text{CL}} - Z\hat{\sigma}(\hat{P}_{\text{CL}}) \leq P_{\text{CL}} \leq \hat{P}_{\text{CL}} + Z\hat{\sigma}(\hat{P}_{\text{CL}}) \quad (18)$$

is at least $1 - \alpha$ where Z is the positive square root of the upper $(1 - \alpha)$ -th percentage point of the χ^2 distribution with L_{max} degrees of freedom.

5 Efficient Simulation Using Importance Sampling

The basic notion behind IS is to modify or bias the original joint probability density function (pdf) for the connection starting-slot vectors, $f_{\mathbf{v}}(\underline{v})$, to a new joint pdf, $f_{\mathbf{v}}^*(\underline{v})$, such that the variance of the estimate of P_{CL} is reduced. Each cell loss event is weighted so that $\hat{P}_{\text{CL}}^*$ remains an unbiased estimate of P_{CL} . The only required condition on the joint pdf $f_{\mathbf{v}}^*(\underline{v})$ is that $f_{\mathbf{v}}^*(\underline{v}) > 0$ when $I_L(\underline{v} \in V_{\text{CL}}) > 0$, where $V_{\text{CL}} \subset V$ is the subset of connections starting-slot vectors which cause a cell loss. Hence, the set V_{CL} is the important region. The joint pdf $f_{\mathbf{v}}(\underline{v})$ we consider here is the product of i.i.d. uniform probability density functions.

The key to using IS is knowing how to bias the pdf $f_{\mathbf{v}}(\underline{v})$. A necessary condition for a good bias scheme is that the number of cell losses that occur must be increased under the biased distribution. The connection starting-slot has a uniform probability distribution over a period T_l . We use *a priori* information about how cell losses occur to arrive at the biased joint pdf by reducing the support of the unbiased joint pdf. The method used is an example of the analytical form of biasing called splitting [17], which takes its name from how the biased pdf is obtained by splitting the original pdf into two regions, one of which contains the important region. We have previously presented how this is done for homogeneous connections traffic descriptors [15].

5.1 Theory

The important region V_{CL} includes every connection starting-slot vector that causes one or more cell losses, so we can write P_{CL} as

$$P_{\text{CL}} = \frac{1}{|V|} \sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right) |V_{\text{CL}}(i)| \quad (19)$$

where $V_{\text{CL}}(i)$ is the subset of connection starting-slot vectors in V_{CL} that cause i cell losses such that $V_{\text{CL}} = \cup_{i=1}^{L_{\max}} V_{\text{CL}}(i)$.

As presented in [15], IS is applied by reducing the sampled set of connection starting-slot vectors from $|V|$ to $|V^*| < |V|$ such that $V_{\text{CL}} \subseteq V^* \subset V$. By maintaining all the connection starting-slot vectors that cause cell losses in V^* , we do not violate the condition necessary to apply IS that $f_{\mathbf{v}}^*(\underline{v}) > 0$ when $I_L(\underline{v} \in V_{\text{CL}}) > 0$. Ideally we would like $V^* = V_{\text{CL}}$, but as long as $|V^*| \ll |V|$ and $V_{\text{CL}} \subseteq V^*$, the efficiency of the simulation can be increased. Furthermore, we define the set $V_{\text{CL}}^{(k)} = \cup_{i=k}^{L_{\max}} V_{\text{CL}}(i)$ which contains all the connection starting slot vectors that cause k or more cell losses and $V^{*(k)}$ such that $V_{\text{CL}}^{(k)} \subseteq V^{*(k)}$. This further partitioning of V_{CL} allows specific cell loss cases to be targeted by sampling from the set $V^{*(k)}$.

The MC simulation estimates the probabilities p_i and forms the overall estimate of P_{CL} in (13). Each p_i can be written as

$$p_i = \frac{|V_{\text{CL}}(i)|}{|V|} = \sum_{k=1}^{|V_{\text{CL}}(i)|} p_V \quad (20)$$

where $p_V = \frac{1}{|V|}$ is the probability of each connection starting slot vector in V and thus V_{CL} . Our IS algorithm biases the probability p_V to a probability $p^*(\underline{v}(k))$ which is dependent on the connection starting-slot vector $\underline{v}(k)$. The simulation using IS estimates each $\hat{p}_i^*$ by independently drawing $N_D(i)$ connection starting-slot vectors out of the set $V^{*(i)}$

$$\hat{p}_i^* = \frac{1}{N_D(i)} \sum_{k=1}^{N_D(i)} I_i(\underline{v}(k)) w(\underline{v}(k)) \quad (21)$$

where each cell loss event is weighted by an amount $w(\underline{v}(k)) = p_V / p^*(\underline{v}(k))$ so that $\hat{p}_i^*$ remains an unbiased estimate of p_i , as shown in the following:

$$E_* \{\hat{p}_i^*\} = \frac{1}{N_D(i)} \sum_{k=1}^{N_D(i)} E_* \{I_i(\underline{v}(k)) w(\underline{v}(k))\}$$

$$= \sum_{k=1}^{|V_{\text{CL}}(i)|} w(\underline{v}(k)) p^*(\underline{v}(k)) = \sum_{k=1}^{|V_{\text{CL}}(i)|} p_V = p_i \quad (22)$$

The estimate of P_{CL} using IS is

$$\hat{P}_{\text{CL}}^* = \sum_{i=1}^{L_{\text{max}}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right) \hat{p}_i^* \quad (23)$$

and the variance of $\hat{P}_{\text{CL}}^*$ is

$$\begin{aligned} \sigma^2(\hat{P}_{\text{CL}}^*) &= \sum_{i=1}^{L_{\text{max}}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 \sigma^2(\hat{p}_i^*) \\ &= \sum_{i=1}^{L_{\text{max}}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 \frac{p_V \sum_{k=1}^{|V_{\text{CL}}(i)|} w_i(\underline{v}(k)) - p_i^2}{N_D(i)} \end{aligned} \quad (24)$$

Since we are able to target the simulation for a specific number of cell losses, different biased distributions can be used in the simulation of each case, resulting in a more accurate estimate of the variance than the method in [18]. In (24), we see that the number of draws $N_D(i)$ is now dependent on i , rather than being equal for all i as with the Monte Carlo experiment in (15).

While this development applies specifically to the cell loss probability of the combined traffic stream, the same IS simulation can be used to estimate the cell loss probability of the individual classes with random buffer filling and priority buffer filling. Consider the set of vectors V_{CL_l} that result in cell losses of class l for the random buffer filling scheme. We know that $V_{\text{CL}_l} \subseteq V_{\text{CL}}$. If during our IS simulation we obtain estimates of $\bar{n}_{\text{CL}_l}(i)$ in each $V_{\text{CL}}(i)$, the average number of cell losses for class l is

$$\bar{n}_{\text{CL}_l} = \sum_{i=1}^{L_{\text{max}}} \bar{n}_{\text{CL}_l}(i) \quad (25)$$

The estimate of the cell loss probability of the l -th class of traffic is then

$$P_{\text{CL}}^{(l)} = \sum_{i=1}^{L_{\text{max}}} \frac{\bar{n}_{\text{CL}_l}}{N_{C_l} \hat{B}_l T_{SP} / T_l} \quad (26)$$

The average cell loss of the combined traffic stream is composed of the average cell loss of the individual classes

$$\bar{n}_{\text{CL}} = \sum_{l=1}^S \bar{n}_{\text{CL}_l} \quad (27)$$

Since cell losses of the individual classes are positively correlated (more cell losses of one class implies more cell losses of another class), the variance of the average number of cell losses of the combined traffic stream is an upper bound for the variance of the average number of cell losses for an individual class

$$\sigma^2(\bar{n}_{\text{CL}}) \geq \sigma^2(\bar{n}_{\text{CL}_l}) \quad (28)$$

and $\hat{\sigma}^2(\bar{n}_{\text{CL}})$ can be used to upper bound the confidence intervals of the cell loss probability for the individual classes. A similar analysis can also be applied to the case of priority buffer filling.

5.2 Application

As in [15], with the system model we consider, cell losses occur when the connections start close enough together, over a number of slots less than $T_{l_{\max}}$, that all the arriving cells can not be served and a buffer overflow results in a cell loss. For the minimum number of connections that causes a cell loss, N_{C_0} , we can determine an interval of c_0^* arrival slots in which all the connections must start for a cell loss to result. If any connection starts more than c_0^* arrival slots away from the beginning of another connection, then a cell loss can never result. The multinomial indicator function $I_L(\cdot)$ is always zero outside this c_0^* length arrival slot interval and the support of the uniform pdf that originally covered the entire interval T_l can be reduced, hence in [15], we named this technique the method of interval reduction.

Define c_r^* , where $r = N_C - N_{C_0}$, as the distance in arrival slots between connections that still cause a cell loss involving the interaction of $N_{C_0} + r$ connections. Interaction between connections is defined to be when the buffer is non-empty between the start of each adjacent connection.

Previously in [15], we presented the equations used to determine c distances and described the method of interval reduction for the case of homogeneous connection traffic descriptors. As shown in Table 1, the equations used to determine c distances are a function of the connection traffic descriptors, the number of connections and the switch parameters. For the

Type	I	II	III
	$\mu \geq 1.0$ and $N_{C_0} - 1 < \mu$	$\mu \geq 1.0$ and $N_{C_0} - 1 \geq \mu$	$\mu < 1.0$
c_0	$\lfloor \hat{B} - \frac{K}{N_{C_0} - \mu} \rfloor + 1$ unless $\hat{B} - \frac{K}{N_{C_0} - \mu} = 0$ then $c_0 = 0$	$\lfloor \hat{B} (N_{C_0} - \mu) - K \rfloor + 1$ unless $\hat{B} (N_{C_0} - \mu) - K = 0$ then $c_0 = 0$	$\lfloor \hat{B}/\mu - \hat{B} \rfloor + 1$
c_r	$r\hat{B} + c_0$	$r\hat{B} + c_0$	$\lfloor (r+1)\hat{B}/\mu - \hat{B} \rfloor + 1$

Table 1: Calculation of c distances as a function of the dominant VBR connection traffic descriptors and the switch parameters.

case of traffic sources described by heterogeneous connection traffic descriptors, we consider cases with only 2 classes. We call the VBR class with the highest peak cell rate the dominant VBR class. We assume that the super-period is the period of the dominant VBR class.

The c^* distances for the CBR/VBR case are increased over the c distances from Table 1, which are generated using the dominant VBR connection traffic descriptors and the number of dominant VBR connections, according to the amount of CBR cells included in the c distances

$$c_r^* (\text{CBR/VBR}) = c_r + \lceil N_{\text{CBR}} \left(c_r + \lceil \left(\hat{B}_{\text{VBR}} + 1 \right) / \left(T_{\text{VBR}} \hat{\lambda}_{\text{VBR}} / \hat{\lambda}_{\text{CBR}} \right) \rceil \right) / (\min(\mu, 1)) \rceil \quad (29)$$

The starting slots of the dominant VBR connections are then selected using the method of interval reduction. The CBR connections are chosen randomly over $[0, T_{\text{CBR}})$, because there is little to be gained, in terms of the improvement in simulation efficiency, by restricting the starting slot of the CBR sources since CBR periods are short relative to the dominant VBR period. For the CBR/VBR case, the number of CBR connections is limited to the condition that $\sum_{l=1}^S N_{C_l} \bar{\lambda}_l < \hat{\mu}$.

For the VBR/VBR case where $\mu < 1$, the c^* distances are

$$c_r^* = \lfloor \hat{B} (\hat{\lambda}_{\max}) / \mu + r\hat{B} (\hat{\lambda}_{\min}) - \hat{B} (\hat{\lambda}_{\max}) \rfloor + 1 \quad (30)$$

For this case, the number of non-dominant VBR connections is limited to the condition $N_C (\hat{\lambda}_{\min}) \hat{B} (\hat{\lambda}_{\min}) \leq \hat{B} (\hat{\lambda}_{\max})$. For the VBR/VBR case where $\mu \geq 1$, the c^* distances are determined using the dominant VBR connection traffic descriptors and the total number of

connections. The starting slots of both classes of VBR connections are then selected using the method of interval reduction.

Examples of the method of interval reduction are given in [15]. The algorithm in Figure 4 uses the method of interval reduction to cause a threshold number of cell losses for each of the $L_{\max}$ cell loss cases of the combined traffic stream for both the CBR/VBR and VBR/VBR cases. The simulation is able to target all i cell losses or greater by reducing the c^* distances using (31).

$$c_r^*(i) = c_r^* - \lfloor (i - 1) / \min(\mu, 1) \rfloor \quad (31)$$

The method of interval reduction is limited to a range of connections N_C such that $N_{C_0} \leq N_C \leq 2N_{C_0} - 1$ or until $c_r^* \geq \min(T(\text{VBR}))/2$ because improvement results only when the restriction of the support is less than the minimum period, and when $N_C < 2N_{C_0}$, since when this occurs there can be two separate cases of cell losses caused by N_{C_0} connections, a condition which is not yet accounted for by this method.

The improvement in simulation efficiency is a result of reducing the support interval over which the next connection is randomly drawn. The method of interval reduction guides the selection of the connection starting slots so that a cell loss occurs while reducing the support over which the next connection starting-slot can be randomly drawn. For each $\underline{v}$, the pdf $f_{\underline{v}}^*(\underline{v})$ changes conditionally as each connection starting-slot is randomly drawn.

5.3 Improvement In Simulation Efficiency Using IS

The improvement in simulation efficiency for the cell loss probability of the combined traffic stream, R_{net} , that results from using IS describes the factor by which the variance of an IS estimator is reduced when compared to the variance of the conventional MC estimator for a fixed sample size, or, equivalently, how many fewer steady-state periods must be simulated to obtain a given accuracy. The value of R_{net} is found analytically by taking the ratio of the variances $\sigma^2(\hat{P}_{\text{CL}})$ and $\sigma^2(\hat{P}_{\text{CL}}^*)$ from (15) and (24), respectively.

$$R_{\text{net}} = \frac{\sigma^2(\hat{P}_{\text{CL}})}{\sigma^2(\hat{P}_{\text{CL}}^*)} = \frac{\frac{1}{N_{MC}} \left(\sum_{i=1}^{L_{\max}} i^2 (p_i - p_i^2) \right)}{\sum_{i=1}^{L_{\max}} i^2 \left(\frac{p_V \sum_{k=1}^{|V_{\text{CL}}^{(i)}|} w_i(\underline{v}(k)) - p_i^2}{N_D(i)} \right)} \quad (32)$$

```

 $i = 1$                                 /* Initialize cell loss target */
while ( $i \leq L_{\max}$ ) {
    while (lost count [ $i$ ] < maximum lost count) {
        < Compute  $c_r^*[i]$  values using  $c_r^*$  >
        < If  $c_r^*[i] < 0$ , then  $c_r^*[i] = c_{r+1}^*[i]$  >
        delay [1] = 0                    /* Fix first connection starting slot at the beginning of a period */
         $j = 2$                             /* Initialize connection counter */
        while ( $j \leq N_C(\text{VBR}) - N_{C_0} + 1$ ) {
            delay [ $j$ ] = int (rand [ $0, T_{\text{VBR}}$ ])          /* Choose connection starting slot over  $[0, T_{\text{VBR}})$  */
             $j = j + 1$                                     /* Increment connection counter */
        }
        < Create 'valid' intervals around each connection >
        while ( ( a 'natural' cell loss can still occur ) and ( $j \leq N_C(\text{VBR})$ ) ) {
            delay [ $j$ ] = int (rand [ $0, T_{\text{VBR}}$ ])          /* Choose connection starting slot over  $[0, T_{\text{VBR}})$  */
            if (delay [ $j$ ] is within an existing interval) {
                < If possible, reduce the size of the existing intervals >
            }
            else {
                < Create 'valid' intervals around connection  $j$  >
            }
             $j = j + 1$                                     /* Increment connection counter */
        }
        while ( $j \leq N_C(\text{VBR})$ ) {
            < Compute  $vT$ , the number of valid slots >
            delay [ $j$ ] = int (rand [ $0, vT$ ])              /* Choose connection starting slot over  $[0, vT)$  */
            < Map delay [ $j$ ] into int ( $[0, T_{\text{VBR}})$ ) >
            < Retain only those intervals which contain delay [ $j$ ] >
            < If possible, reduce the size of the remaining intervals >
             $j = j + 1$                                     /* Increment connection counter */
        }
        for ( $k = 0; k < N_C(\text{CBR}); k++$ ) {
            delay [ $j$ ] = int (rand [ $0, T_{\text{CBR}}$ ])          /* Choose connection starting slot over  $[0, T_{\text{CBR}})$  */
             $j = j + 1$                                     /* Increment connection counter */
        }
        < Run network simulation >
    }
    while (lost count [ $i$ ]  $\geq$  minimum lost count) {
        < Collect statistics >
         $i = i + 1$                                         /* Increment cell loss target */
    }
}
< Compute  $\hat{P}_{\text{CL}}$ ,  $\hat{P}_{\text{CL}}^{(l)}$ ,  $\hat{P}_{\text{CL}}^{(lP)}$ ,  $\hat{R}_{\text{net}}$ , and confidence intervals >

```

Figure 4: Pseudo-code describing the method of interval reduction for traffic sources described by heterogeneous connection traffic descriptors.

Connection	$\hat{\lambda}$, Mbps	$\bar{\lambda}$, Mbps	$\hat{B}$, cells	T, a.s.
CBR-1	10	10	1	1
VBR-1	50	5	25	250
VBR-2	500	2	50	12,500

Table 2: Connection traffic descriptors for the experimental examples (a.s. = arrival slots).

We see from (32) that the simulation efficiency is improved when $\sigma^2(\hat{P}_{\text{CL}}^*) < \sigma^2(\hat{P}_{\text{CL}})$ which results when $w_i(\underline{v}(k)) < 1$. Using the method of interval reduction, we are guaranteed that $w_i(\underline{v}(k)) \leq 1$. In fact, when $w_i(\underline{v}(k)) = 1$, $R_{\text{net}} = 1$ for an equal number of draws as expected.

The method for estimating the improvement $\hat{R}_{\text{net}}$ uses the estimate of the variance of the estimate, $\hat{\sigma}_{IS}^2(\hat{P}_{\text{CL}}^*)$ from (17), and the total number of random draws, N_{IS} , used in the importance sampling experiment. The number of random draws required to obtain a given accuracy using conventional Monte Carlo simulation is closely approximated by

$$N_{MC} = \frac{\sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 p_i (1 - p_i)}{\sigma^2(\hat{P}_{\text{CL}})} \quad (33)$$

By substituting $\hat{\sigma}_{IS}^2(\hat{P}_{\text{CL}}^*)$ for $\sigma^2(\hat{P}_{\text{CL}})$ and $\hat{p}_i^*$ for p_i , the estimated improvement $\hat{R}_{\text{net}}$ is then closely approximated by

$$\hat{R}_{\text{net}} = \frac{N_{MC}}{N_{IS}} = \frac{\sum_{i=1}^{L_{\max}} \left(\frac{i}{\bar{n}_{\text{CA}}} \right)^2 \hat{p}_i^* (1 - \hat{p}_i^*)}{\hat{\sigma}_{IS}^2(\hat{P}_{\text{CL}}^*) N_{IS}} \quad (34)$$

In using IS, the pdf must be modified to increase the efficiency of the simulation associated with a particular estimator, in this case the cell loss probability of the combined traffic stream. However, the simulation efficiency for the estimates of the cell loss probability for the individual classes with random buffer filling and priority buffer filling will also be improved, although we are not able to quantify this improvement exactly.

6 Experimental Examples

We considered the experimental examples with the connection traffic descriptors shown in Table 2 in the combinations shown in Table 3. We selected the peak cell rate of the CBR class to be comparable to that required by video traffic. The peak cell rate of the dominant

System	$\hat{\mu}$, Mbps	K , c.	T_{SP} , a.s.	μ , c./a.s.	N_{C_0}	VBR-2	c_0 , a.s.	c_1 , a.s.	c_2 , a.s.	$L_{\max}(N_{C_0})$, c.
CBR-1 vs. VBR-2	120	100	12,500	0.24	5		159	184	217	40
VBR-1 vs. VBR-2	120	200	12,500	0.24	5		159	264	369	45

Table 3: Switch parameters and derived simulation parameters for the experimental examples (a.s. = arrival slots, c. = cells).

Figure 5: Cell loss probability of the combined traffic stream and the individual classes with random buffer filling and priority buffer filling as a function of the number of CBR-1 connections and five VBR-2 connections.

VBR class, VBR-2, was chosen to be a factor of 10 higher than the peak cell rate of class VBR-1. In the algorithm in Figure 4, we set the minimum loss count equal to 10 cell losses and the maximum loss count equal to 20 cell losses.

For the experimental examples, Figures 5 and 6 show the curves of the estimates of the cell loss probability of the combined traffic stream, as well as the estimates of the cell loss probabilities for the individual classes with random buffer filling and priority buffer filling.

For each performance curve, the bottom points were obtained by simulation using IS, and the upper points were obtained by conventional MC simulation. The accuracy of the conventional MC estimates is measured using the precision, ϵ , defined as $\epsilon = \hat{\sigma}(\hat{P}_{CL}) / \hat{P}_{CL}$

Figure 6: Cell loss probability of the combined traffic stream and the individual classes with random buffer filling and priority buffer filling as a function of the number of VBR-1 connections and five VBR-2 connections.

and is typically $\epsilon \leq 0.1$ for the experimental examples shown.

Two types of discontinuities can occur in the performance curves. In (7), we determined the minimum number of connections, N_{C_0} , needed to cause at the least one cell loss. Thus, for $N_C < N_{C_0}$, $P_{CL} = 0$. The downward pointing arrow at the bottom of each performance curve is meant to signify that P_{CL} is zero below N_{C_0} connections, thus there is a point on the performance curve where cell losses begin to occur and below this point cell losses do not occur. This is an important result because the CAC algorithm can quickly determine that a new connection can be admitted to the network since no cell losses will occur, thus satisfying any QoS standard. The guaranteed cell loss condition in (8) also creates a discontinuity in that there is a point on the curve where a ‘jump’ in the cell loss probability is possible. This point can be seen in Figure 5 between 11 and 12 CBR-1 connections as a result of N_{C_g} occurring at 12 CBR-1 connections.

For the CBR/VBR case shown in Figure 5, the cell loss probability initially decreases as a result of increasing the number of cell arrivals while the cell losses do not increase

accordingly. However, the cell loss probabilities of the individual classes continue to increase monotonically. In contrast to the performance curves in [15] which are concave down, the performance curves here are concave up in the CBR/VBR case and nearly linear in the VBR/VBR case, at least until the point is approached where cell losses are guaranteed to occur. This difference in shape is a result of the CBR-1 and VBR-1 classes being less bursty than the VBR-2 class. The difference in burstiness for the CBR/VBR case results in the cell loss probability of the CBR-1 traffic in Figure 5 being one or two orders of magnitude below the cell loss probability of the VBR-2 traffic. In the VBR/VBR case in Figure 6, the cell loss probability of the VBR-1 traffic is roughly one order of magnitude below the cell loss probability of the VBR-2 traffic. For both the CBR/VBR and VBR/VBR cases, there is little difference in the cell loss probability between the random buffer filling scheme and the priority buffer filling scheme.

A ‘gap’ between the points found by simulation using IS and the points found by conventional MC simulation can occur because of the restrictions on the number of connections that can be simulated using IS (See Section 5.2). For most cases, this gap can be bridged using cubic spline interpolation, since we expect a smooth curve in the gap region.

For the experimental examples, the confidence intervals for the estimates of P_{CL} are obtained by running a total of N_{IS} steady-state periods shown in Table 4 and applying the method from [16] given in (18). In Table 4, we give the estimates and confidence intervals of the cell loss probability of the combined traffic stream and the individual classes, with random buffer filling and priority buffer filling, for the N_{C_0} and $N_{C_0} + 1$ points. The lower confidence interval of the CBR-1 estimates is zero, which is a result of using the upper bound for the estimate of the variance in (28). The estimate of the improvement in efficiency over conventional MC simulation obtained by simulation using IS, $\hat{R}_{net}$, for the experimental examples is also shown in Table 4, along with the simulation time required for $N_D = 100$ steady-state periods using MC simulation for N_{C_0} connections on a DECStation 5000/25. The improvement that we observed in the simulation using IS is inversely proportional to the estimate, $\bar{n}_{CL}$. The use of importance sampling added less than 0.1% overhead to the effort required for conventional Monte Carlo simulation.

System	N_C	N_{IS}	$\bar{n}_{CL}$	$\hat{P}_{CL}$	95% CI	$\hat{R}_{net}$	$t_{N_D=100}(N_C)$, secs.
VBR-2	5	9,434	250	4.28×10^{-9}	$(3.49 \times 10^{-9}, 5.08 \times 10^{-9})$	2.1×10^6	61.12
CBR-1/VBR-2	1/5	6,739	250/250	3.44×10^{-9}	$(2.51 \times 10^{-9}, 4.39 \times 10^{-9})$	8.7×10^5	66.63
CBR-1				1.42×10^{-10}	$(0, 2.02 \times 10^{-9})$		
VBR-2				6.76×10^{-9}	$(4.88 \times 10^{-9}, 8.63 \times 10^{-9})$		
CBR-1 P				1.17×10^{-10}	$(0, 1.99 \times 10^{-9})$		
VBR-2 P				6.73×10^{-9}	$(4.86 \times 10^{-9}, 8.61 \times 10^{-9})$		
VBR-1/VBR-2	1/5	412,448	125/250	2.32×10^{-10}	$(5.62 \times 10^{-11}, 4.07 \times 10^{-10})$	1.8×10^5	65.98
VBR-1				5.63×10^{-11}	$(0.0, 5.83 \times 10^{-10})$		
VBR-2				3.13×10^{-11}	$(4.99 \times 10^{-11}, 5.76 \times 10^{-10})$		
VBR-1 P				4.38×10^{-11}	$(0.0, 5.70 \times 10^{-10})$		
VBR-2 P				3.07×10^{-11}	$(4.37 \times 10^{-11}, 5.70 \times 10^{-10})$		

Table 4: Estimates of the cell loss probability, confidence intervals and the estimate of the improvement factor obtained by simulation using IS, and the simulation time required for $N_R = 100$ steady-state periods using MC simulation on a DECStation 5000/25 for the experimental examples.

7 Conclusion

In contrast to previous system models which used stochastic models to describe the traffic entering the network, we have utilized a system model with a traffic source derived from the ATM Forum standardized connection traffic descriptors which are also used by the connection admission control algorithms. This system model takes advantage of the deterministic behavior that occurs after the connections starting-slot vector has been randomly drawn to speed-up the simulation without the use of IS. In this paper, we examined traffic sources described by heterogeneous connection traffic descriptors in order to analyze CBR sources versus VBR sources, as well as different classes of VBR sources. We used Monte Carlo simulation to estimate the cell loss probability of the combined traffic stream, as well as the individual classes using both random buffer filling and priority buffer filling schemes, and presented how an adaptation of the method of interval reduction can be used to apply IS. For the experimental examples considered here, the improvement in simulation efficiency compared to conventional Monte Carlo simulation is inversely proportional to the estimate. Utilizing the traffic source derived from the standardized connection traffic descriptors along with the multinomial formulation and an adaptation of the method of interval reduction results in efficient simulation suitable to the design and testing of the switches and CAC algorithms planned for use in ATM networks.

References

- [1] ATM User-Network Interface Specification, Version 3.0. The ATM Forum, September 10, 1993. The ATM Forum.
- [2] S. Parekh and J. Walrand. A Quick Simulation Method for Excessive Backlogs in Networks of Queues. *IEEE Trans. Automat. Contr.*, AC-34(1):54–66, Jan. 1989.
- [3] Q. Wang and V. S. Frost. Efficient Estimation of Cell Blocking Probability for ATM Systems. *IEEE/ACM Trans. on Networking*, 1(2):230–235, 1993.
- [4] P. Heidelberger. Fast Simulation of Rare Events in Queueing and Reliability Models. In *Proc. of Performance '93*, Rome, Italy, October 1993.
- [5] H. Perros. A Literature Review of Call Admission Algorithms. Technical Report TR-94/8, Center for Comm. and Signal Processing, North Carolina State University, April 1994.
- [6] A. W. Berger and A. E. Eckberg. A B-ISDN/ATM Traffic Descriptor, and its Use In Traffic and Congestion Control. In *Proc. IEEE Global Telecom. Conf., GLOBECOM '91*, Phoenix, AZ, May 1991.
- [7] T. Murase, H. Suzuki, S. Sato, and T. Takeuchi. A Call Admission Control Scheme for ATM Networks Using A Simple Quality Estimate. *IEEE J. Select. Areas in Commun.*, 9(9):1461–1470, Dec. 1991.
- [8] Y. Miyao. A Call Admission Control Scheme in ATM Networks. In *Proc. IEEE Int. Conf. Commun., ICC '91*, Denver, CO, Dec. 1991.
- [9] J. W. Roberts and J. T. Virtamo. The Superposition of Periodic Cell Arrival Streams in an ATM Multiplexer. *IEEE Trans. on Commun.*, 39(2):298–303, Feb. 1991.
- [10] C. Rasmussen, J. H. Sorensen, K. S. Kvols, and S. B. Jacobsen. Source-Independent Call Acceptance Procedures in ATM Networks. *IEEE J. Select. Areas in Commun.*, 9(3):351–358, April 1991.

- [11] B. T. Doshi. Deterministic Rule Based Traffic Descriptors for Broadband ISDN: Worst Case Behavior and Connection Acceptance Control. In *14th International Teletraffic Congress*, Paris, June 1994.
- [12] H. Saito. Call Admission Control in an ATM Network Using Upper Bound of Cell Loss Probability. *IEEE Trans. on Commun.*, 40(9):1512–1521, Sept. 1992.
- [13] Y. Sato and K. Sato. Evaluation of Statistical Cell Multiplexing Effects and Path Capacity Design in ATM Networks. *IEICE Trans. on Commun.*, E75-B(7):642–648, July 1992.
- [14] B. Jabbari and F. Yegenoglu. An Upper Bound for Cell Loss Probability of Bursty Sources in Broadband Packet Networks. In *Proc. IEEE Int. Conf. Commun., ICC '91*, Denver, CO, Dec. 1991.
- [15] J. A. Freebersyser and J. K. Townsend. A Multinomial Formulation for Efficient Simulation Simulation of Periodic On/Off Sources. To appear in the conference on *Modeling, Analysis, and Simulation of Computer and Telecomm. Systems (MASCOTS '96)*.
- [16] R. Z. Gold. Tests Auxiliary to χ^2 Tests in a Markov Chain. *Ann. Math. Statist.*, Vol. 34:56–74, 1963.
- [17] H. Kahn. Use of Different Monte Carlo Sampling Techniques. *Symposium on Monte Carlo Methods*, pages 146–190, Mar. 1954.
- [18] J. A. Freebersyser and J. K. Townsend. Efficient Simulation of Cell Loss Probability Using Standardized ATM Connection Traffic Descriptors. In *Proc. IEEE Int. Conf. Commun., ICC '95*, pages 298–303, Seattle, June 1995.