In-Service Monitoring for Cell Loss Quality of Service Violations in ATM Networks

Hongbo Zhu, *Student Member, IEEE,* and Victor S. Frost, *Senior Member, IEEE*

Abstract—**A new method for in-service cell loss ratio (CLR) QoS estimation for asynchronous trasfer mode (ATM) networks has been developed. For a typical CLR, a large number of cells must be observed before statistically meaningful results can be achieved. These results may be obsolete resulting in ineffective network management reactions. For a variety of arrival processes, many analyses have shown there exists a relationship between the logarithm of the cell loss probability and buffer size. For models that do not possess long-range dependence, this relationship is often linear. On the other hand, for the fractional Brownian motion model that captures long-range dependent traffic behavior, this relationship has a polynomial form. The proposed method uses these relationships and observations of cell loss for several small pseudo-buffers to characterize the behavior of the actual system. Specifically, a real-time technique to dynamically detect the failure of meeting a cell loss quality of service (QoS) objective has been developed. The method requires a short observation period and is suitable for in-service monitoring of CLR QoS. Simulation studies show the effectiveness of this method for both modeled traffic and measured network trace data.**

I. Introduction

In ASYNCHRONOUS Transfer Mode (ATM)-based networks, quality of service (QoS) requirements are very stringent. A prime concern is to ensure that there are adequate resources to meet the traffic demand or to prioritize the use of resources when short falls are unavoidable. Network monitoring and estimation have to be performed in order to keep abreast of demand and are essential in network traffic control.

To provide dynamic network control and management, in-service monitoring and estimation (ISME) have been employed as opposed to the conventional out-of-service testing (OOST) techniques [3]. One potential problem for ISME is that some QoS indicators are specified in terms of the probability of occurrence of certain rare events, e.g., in ATM-based networks, cell loss probability is often specified to be less than 10^{-9}. Monitoring using direct statistical methods is impracticable for estimating such small probabilities. In the above example, at least 10 billion cells have to be monitored before any statistically meaningful information can be collected. Assuming a link rate of 155 Mb/s and a cell arrival probability of 0.6 in each time slot, monitoring 10 billion cells would take 12 h. The statistical information obtained after such a long monitoring period may be obsolete and the network management system reaction may be too late.

Here, an ISME method is developed to quickly detect CLR QoS violation for a single buffer in an ATM system. Once a CLR QoS violation is detected, an alarm can be sent to the network management system. The purposes of using ISME for CLR QoS assurance are to:

- Monitor the CLR performance under in-service conditions and verify that the performance meets the QoS requirements.
- Identify the location and causes for the CLR performance degradation without affecting customers.
- Conduct reactive and preventive maintenance by continuously investigating performance trends.

Some potential techniques for accelerating the monitoring and estimation speed have been described in the literature [6], [9]. These techniques involve complex analysis and are applicable only to particular cases. The new method proposed in this paper employs simple linear regression and hypothesis testing techniques, requires a short monitoring period, and is shown to be effective for a variety of traffic types, including fractional Brownian motion, used to characterize long-range dependence of traffic [8].

This paper is organized as follows: Section II will review the basis for the proposed technique, that is, the relationship between buffer size and log(CLR). Section III will present the ISME procedure for the detection of CLR QoS violation. Section IV will conduct a performance evaluation of the detection scheme. Section V presents our conclusions.

II. Relationship Between CLR and Buffer Size in ATM Systems

This section is dedicated to establishing the validity of using a generalized relationship between buffer size and log (CLR) for ISME. Analytical results from previous works will be reviewed and summarized to demonstrate this relationship for both Markovian-type arrival processes and a recently proposed long-range dependent traffic model, namely, fractional Brownian motion [8].

A. Markovian Arrival Process

For Markovian-type queueing systems, the log(CLR) is known to decrease proportionally with increasing buffer size. This linear relationship is also a natural outgrowth of the fluid flow analysis of ATM systems [12]. Furthermore, in

Manuscript received February 1995; revised September 1995; approved by IEEE/ACM Transactions on Networking Editor H. Miyahara. This work was supported in part by Bell Northern Research.

The authors are with the Department of Electrical Engineering and Computer Science, University of Kansas, Telecommunications and Information Sciences Laboratory, Lawrence, KS 66045 USA.

Publisher Item Identifier S 1063-6692(96)02716-1.

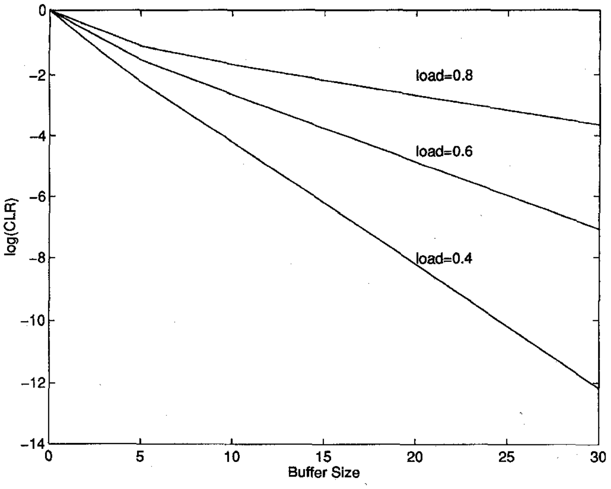


Fig. 1. log(CLR) versus buffer size, Geom/Geom/1/N model, $s = 1.0/11.0$.

[14], the equivalent capacity concept developed for bandwidth allocation also assumes a linear relationship between buffer size and log(CLR).

Here, we present some previous analytical results showing the relationship between buffer size and log(CLR). Both the Geom/Geom/1/K queueing model and the N-state Markovian arrival process are considered.

For a single server queue with a buffer size of K cells, assume both the cell arrival and the cell departure processes are identically and independently distributed (i.i.d.) Bernoulli processes. Also assume p is the arrival probability of a cell during a time slot and s is the departure probability of a cell during a time slot. The cell interarrival time and interdeparture time are both geometrically distributed (Geom/Geom/1/K model). The Geom/Geom/1/K model can be used to characterize the queueing behavior in an ATM switch using input queueing. In this case, a customer arriving to an empty queue must wait at least until the next slot for service (noncut-through). It can be shown the CLR has the following expression [4]:

$$\text{CLR } = P_b = \frac{(1-\rho)\rho^K(1-s)}{1-\rho^{K+1}-(1-\rho)s} \tag{1}$$

where P_b denotes the cell blocking probability, K is the buffer size, and $\rho = [p(1-s)]/[s(1-p)]$. Fig. 1 shows the log(CLR) versus buffer size for three different values of ρ. It is obvious that these curves are all asymptotically straight lines.

In general, when the input bit rate is characterized by an N-state Markov chain, the queue fill distribution is of the form [14], [12]

$$F(x) = \sum_{i=1}^{N} a_i\, \alpha_i \phi_i\, e^{z_i x}. \tag{2}$$

z_i and ϕ_i are, respectively, the generalized eigenvalues and eigenvectors associated with the solution of the differential equation satisfied by the stationary probabilities of the system, and a_i are coefficients determined from boundary conditions. The CLR is estimated by truncating the queue length distribution and computing the probability that the queue length exceeds the real buffer size. Specifically, an asymptotic CLR estimation is obtained as follows:

$$G(x) \sim \rho^N \left\{ \prod_{i=1}^{N-\lfloor c \rfloor -1} \frac{z_i}{z_i+r} \right\} e^{-rx} \tag{3}$$

where $G(x)$ represents the probability that queue fill exceeds x. Also, c and ρ are normalized link capacity and normalized offered load, respectively. The value r is the dominant eigenvalue z_0, and can be calculated as

$$r = z_0 = \frac{(1-\rho)(1+\lambda)}{1-c/N}. \tag{4}$$

Other eigenvalues z_i are the negative roots of the following quadratics by setting $k = i$:

$$A(k)z^2 + B(k)z + C(k) = 0, \qquad k = 0, 1, \cdots, N \tag{5}$$

where

$$\begin{aligned} A(k) &= \left(\frac{N}{2-k}\right)^2 - \left(\frac{N}{2-c}\right)^2 \\ B(k) &= 2(1-\lambda)\left(\frac{N}{2-k}\right)^2 - N(1+\lambda)\left(\frac{N}{2-c}\right) \\ C(k) &= -(1+\lambda)^2\left\{\left(\frac{N}{2}\right)^2 - \left(\frac{N}{2-k}\right)^2\right\}. \end{aligned}$$

Note, (3) indicates an asymptotically linear relationship between buffer size and log(CLR). Recently, this linear relationship has been proven to hold within a much more general context [18], [22]. In fact, based on the derived fundamental bounds and experience with numerical experiments, the authors of [22] proposed the following model for systems with general Markovian sources:

$$\log\,(\text{CLR}) \sim -\alpha - \delta B \tag{6}$$

where B is buffer size, and δ and α are both positive constants. This generalized result has been used in many situations to develop various algorithms for control and routing of ATM networks [14], [17], [22].

B. Long-Range Dependent Traffic Model

Recently, by investigating Ethernet traffic and variable bit rate (VBR) video trace data, the authors of [7] and [8] propose that network traffic arrival processes may possess self-similarity and long-range dependence that cannot be characterized by conventional Markovian-type traffic models. A wide-sense stationary stochastic process exhibits long-range dependence if its auto-correlation function decays hyperbolically as the lag increases. In turn, a new traffic model, namely, *fractional Brownian motion* is proposed in [7] and [8] to model self-similarity and long-range dependence.

For long-range dependent traffic processes, the linear relationship between buffer size and log (CLR) no longer exists. Fig. 2 shows the result of an extensive simulation we conducted to observe the buffer size versus log (CLR) using

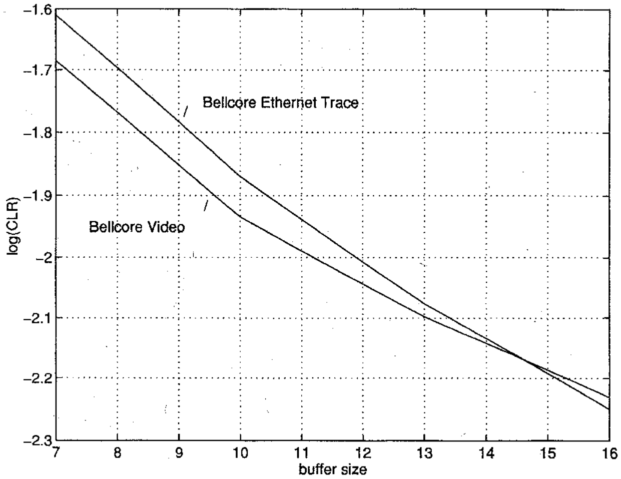


Fig. 2. log (CLR) versus buffer size for Bellcore trace data.

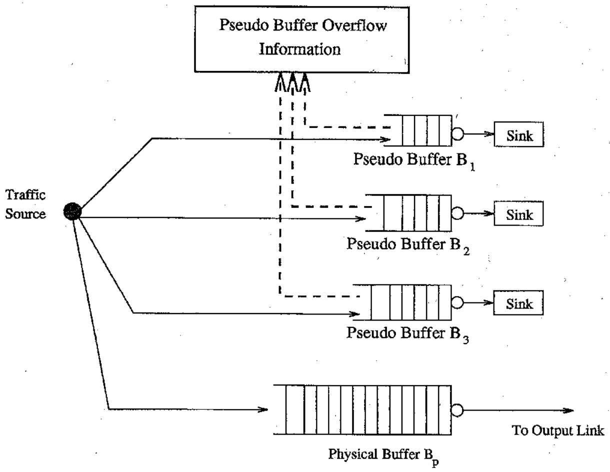


Fig. 3. Use of pseudo-buffers.

Bellcore trace data [7]. This simulation result illustrates a nonlinear relationship between buffer size and log (CLR)(CLR).

In contrast to the Markovian model (6), buffer size and log (CLR) for the fractional Brownian motion model results in the following generalized relationship:

$$\log\,(\text{CLR}) \sim -\delta B^{\beta} \qquad (7)$$

where δ and β are constants determined by traffic processes under consideration (see [19]–[21]). In this paper, we treat δ and β as unknown parameters and an on-line regression method was developed to estimate these two parameters. Taking the logarithm in both sides of (7), we obtain

$$\log\,[-\log\,(\text{CLR})] \sim \log\,(\delta) + \beta\,\log\,(B). \qquad (8)$$

Note the relationship in (7) is *intrinsically linear*, if $\log\,[-\log\,(\text{CLR})]$ and $\log\,(B)$ are both viewed as variables.

It is important to note that the existence of long-range dependence in actual traffic is a controversial issue. For example, the authors of [22] show that buffer size versus log(CLR) exhibits a linear relationship through extensive simulation using video teleconferencing trace data. They further claim that their simulation study does not conform to the long-range dependence observed in [7] and [8].

Despite the existing controversy, the log(CLR) versus buffer size relationship for both Markovian and fractional Brownian motion source models are taken into account in the development of the on-line ISME algorithm in the next section.

III. A CLR QoS Violation Detection Algorithm

A. General Description

Using both the Markovian and the long-range dependent models, a fast CLR QoS violation scheme has been developed using ordinary least square and hypothesis testing techniques. The basic idea is to employ several pseudo-buffers whose sizes are much smaller than the physical buffer size [15], [17]. Fig. 3 shows a system model using pseudo-buffers. Notice the pseudo-buffers can be implemented as simple counters, incrementing on each cell arrival when not at its limit, and decreasing on each departure.

Although the desired value of CLR QoS is very small (10^{-9}) at the physical buffer size, the corresponding CLR is much higher (e.g., 10^{-2}) at these pseudo-buffer sizes. The cell overflow events associated with these small pseudo-buffers can be observed and counted in a background mode on-line, and a linear regression algorithm is applied to obtain an estimation of CLR at the physical buffer size. To obtain statistically meaningful CLR information associated with these pseudo-buffers, fewer arriving cells need to be observed compared to those needed for a large physical buffer size. As will be shown, the monitoring period can thus be significantly reduced. For example, assume a certain QoS requires the CLR at the physical buffer size to be less than 10^{-9}. To achieve this performance, the physical buffer size is designed to be 200 cells. To monitor CLR associated with this physical buffer, one would have to observe at least 10^{10} incoming cells to get a statistically meaningful estimate for CLR. With the proposed method, three pseudo-buffers whose sizes are seven, 10, and 13 can be used. One can expect that under the same traffic that causes the physical buffer to have a CLR of 10^{-9}, those small pseudo-buffers will have much higher CLR's, say, the order of 10^{-2}. Therefore, the monitoring period can be reduced to an order of (10^3–10^4) cells. The idea of tail extrapolation introduced here is not new; it has been successfully applied to estimate very low data link level bit error probabilities [16].

B. CLR Estimation

For further evaluation and analysis, the following definitions and notation will be used:

n Number of cells to be observed to get one sample log(CLR).

N Number of log(CLR) samples.

$N \times n$ Total number of cells observed during one monitoring period.

C_i Number of overflows counted during one n-cell period.

$l_i = \log(C_i/n)$ This is the observed $\log(\text{CLR})$ at ith pseudo-buffer for one n-cell period.

L_n Estimated $\log(\text{CLR})$ at physical buffer during one n-cell period.

$\overline{L}$ Sample mean of estimated $\log(\text{CLR})$ at physical buffer. This is the decision variable used in hypothesis testing.

Q Desired QoS $\log(\text{CLR})$ value at physical buffer.

Note, (6) and (8) do not represent the same regression model. In order to take both models into account when performing linear regression, we use the well-known goodness-of-fit R^2 [2] test to test their validity. The procedure that selects the appropriate regression model based on the observed traffic is described as follows:

1) Whenever n observed samples arrive, perform linear regressions separately based on both regression models [(6) and (8)]. Both regression results are stored.
2) Perform R^2 tests on both regressions, and compare the R^2 results. Keep two running counters, C_M and C_L. If the R^2 result associated with the Markovian model is greater than that associated with the long-range dependent model, then C_M is incremented by one; otherwise, counter C_L is incremented by one. C_M and C_L represent the number of times when the R^2 test chooses the Markovian model and long-range dependent model, respectively.
3) Perform steps 1) and 2) until N n-cell periods have been observed. Then compare the counter results. If C_M is greater than C_L, then choose the Markovian model as the valid model and abandon regression results from the long-range dependent model; otherwise, choose the long-range dependent model as the valid model and abandon the regression results from the Markovian model.
4) Extrapolate to get N sample estimation at the physical buffer size based on the chosen regression model. Note, if the long-range dependent model is the chosen model, then the extrapolation results would be $\log[-\log(\text{CLR})]$; in this case the results will be immediately converted to $\log(\text{CLR})$ for further detection purposes.

Both regression models have the form of $Y = b_0 + b_1 \times X$, where Y and X are regression variables. Let b_0^* and b_1^* denote the best estimates of b_0 and b_1, respectively. The b_0^* and b_1^* are obtained as

$$b_1^* = \frac{\sum (X_i \times Y_i - \overline{X} \times \overline{Y})}{\sum (X_i^2 - \overline{X}^2)} \tag{9}$$

$$b_0^* = \overline{Y} - b_1^* \overline{X}. \tag{10}$$

The R^2 is defined as

$$R^2 = \frac{\sum (b_0^* + b_1^* \times Y_i - \overline{Y})^2}{\sum (Y_i - \overline{Y})^2}. \tag{11}$$

In the current application, l_i or $[\log(l_i)]$ and B_i (or $\log(B_i)$) are the regression variables under the Markovian model (or the long-range dependent model). Note, since all pseudo-buffers are fed by the same traffic source, the errors between the observed and the regressed values may be correlated. Furthermore, simulation results indicate that the variance of l_i's are different for different pseudo-buffer sizes as expected. Using the ordinary least square method in these situations may cause a regression error. We claim the regression error is small. To further reduce the regression error, we choose to monitor pseudo-buffers for N n-cell periods and to conduct N regressions by using the ordinary least square method. Then the N independent samples obtained at the physical buffer are averaged to obtain the decision variable $\overline{L} = 1/N \sum_{j=1}^{N} L_j$. This averaging reduces the variance at each pseudo-buffer by a factor of N. Later simulations with both traced and modeled traffic will verify the effectiveness of this method.

Note, $\overline{L}$ represents an estimate of the expected value of L, denoted by μ_L. The variance of L denoted by σ_L^2 can also be estimated from samples. An estimation for σ_L^2 is S_L^2, expressed as

$$S_L^2 = \frac{1}{N-1} \sum_{j=1}^{N} (L_j - \overline{L})^2. \tag{12}$$

C. The QoS Violation Detection Algorithm

Assume the cell arrival process during the short monitoring period is stationary. According to the central limit theorem, when N is large (>30), $\overline{L}$ has a Gaussian distribution with mean μ_L and variance σ_L^2/N (again assuming independent samples). Using the known characteristics of the distribution of $\overline{L}$, a hypothesis testing can be formed to determine if a violation of CLR QoS has occurred during the current monitoring period. Define two hypotheses:

1) H_0: CLR QoS is satisfied,
2) H_1: CLR QoS is violated.

If H_1 is selected, an alarm is sent out immediately. On the other hand, if, based on the current monitoring period, H_0 is selected, no alarm is sent and the network simply keeps on monitoring. A Neyman–Pearson detection algorithm [2] is used to make selections between the two hypotheses.

Let D_i denote the decision in favor of H_i. Also, let P_M denote the *miss probability* $P(D_0|H_1)$, and P_F denote the *false alarm probability* $P(D_1|H_0)$. Note, the probability of detect P_D is $1 - P_M$. To evaluate P_M and P_F, the conditional probability density functions $f_{\overline{L}|H_0}$ and $f_{\overline{L}|H_1}$ are required. Once $f_{\overline{L}|H_0}$ and $f_{\overline{L}|H_1}$ are specified, a threshold T can be set to satisfy a predetermined value of P_F and, consequently, P_M can be computed from T. The detection scheme should be designed such that both P_F and P_M are small. The decision rule is as follows:

- If $\overline{L} > T$, accept H_0, or decide CLR QoS is satisfied.
- If $\overline{L} < T$, decide CLR QoS violated, and send an alarm.

Since $\overline{L}$ has a Gaussian distribution, the conditional mean and variance are the only information needed to complete the knowledge of $f_{\overline{L}|H_0}$ and $f_{\overline{L}|H_1}$. We shall denote the variance

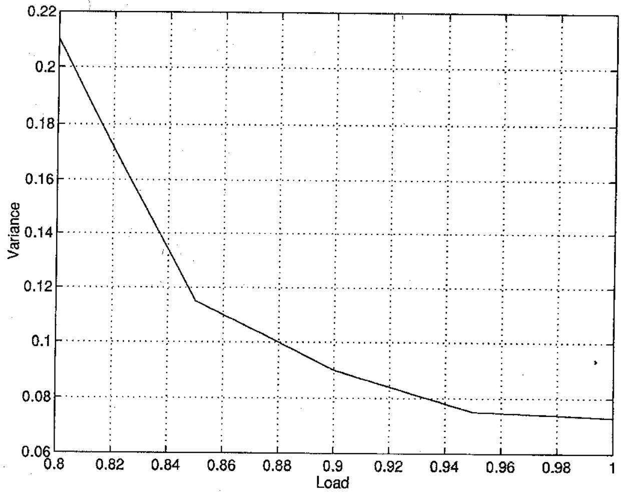


Fig. 4. $S^2_{\overline{L}}$ versus load.

for $f_{\overline{L}|H_0}$ and $f_{\overline{L}|H_1}$ as $\sigma^2_{\overline{L}|H_0}$ and $\sigma^2_{\overline{L}|H_1}$, respectively. In this scheme, $\overline{L}$ represents an estimation of the mean under $f_{\overline{L}|H_1}$ and the mean under $f_{\overline{L}|H_0}$ is simply Q. From extensive simulation studies, it was found that $\sigma^2_{\overline{L}}$ decreases when traffic load increases. For example, Fig. 4 shows the $S^2_{\overline{L}}$ for a 32 homogeneous on/off source as a function of load. This behavior is expected because as load increases, the system observes more cell losses. Therefore, it is expected that $\sigma^2_{\overline{L}|H_1}$ is smaller than $\sigma^2_{\overline{L}|H_0}$, and they should be estimated separately. Equation (12) can be used as an empirical estimate of $\sigma^2_{\overline{L}|H_1}$. The value of $\sigma^2_{\overline{L}|H_0}$ can be found from network management experience, because for most of time, CLR QoS is satisfied, i.e., during normal network operations, the load is no larger than the load that causes Q. Also, because $\sigma^2_{\overline{L}}$ decreases when load increases, the $\sigma^2_{\overline{L}}$ obtained from normal network operation should be no smaller than $\sigma^2_{\overline{L}|H_0}$. Thus, the $\sigma^2_{\overline{L}}$ obtained during normal network operation represents a worst-case value in the sense of assigning a desired P_F and calculating T.

Once $f_{\overline{L}|H_0}$ and $f_{\overline{L}|H_1}$ are identified, the threshold T can be found by using $f_{\overline{L}|H_0}$ and the desired value of P_F. In fact, T should be chosen such that

$$\text{Prob}\,(\overline{L} > T|H_0) = P_F. \tag{13}$$

Since $f_{\overline{L}|H_0}$ is normally distributed with a mean of Q and variance of $\sigma^2_{\overline{L}|H_0}$, the probability in (13) can be expressed as

$$\text{Prob}\left(Z > \frac{T-Q}{\sigma_{\overline{L}|H_0}}\right) = Q\left(\frac{T-Q}{\sigma_{\overline{L}|H_0}}\right) = P_F \tag{14}$$

where $Z = (\overline{L} - Q)/\sigma_{\overline{L}|H_0}$, and $Q(x)$ is the Q-function. The value of Z corresponding to P_F can be found from the Q-function table, and T can be solved accordingly. After T is obtained, P_M, P_D, and P_F can also be easily computed.

The ISME for CLR QoS is summarized in Fig. 5. Note in Fig. 5, the values of $\sigma^2_{\overline{L}|H_0}$, T, and P_F are determined off-line.

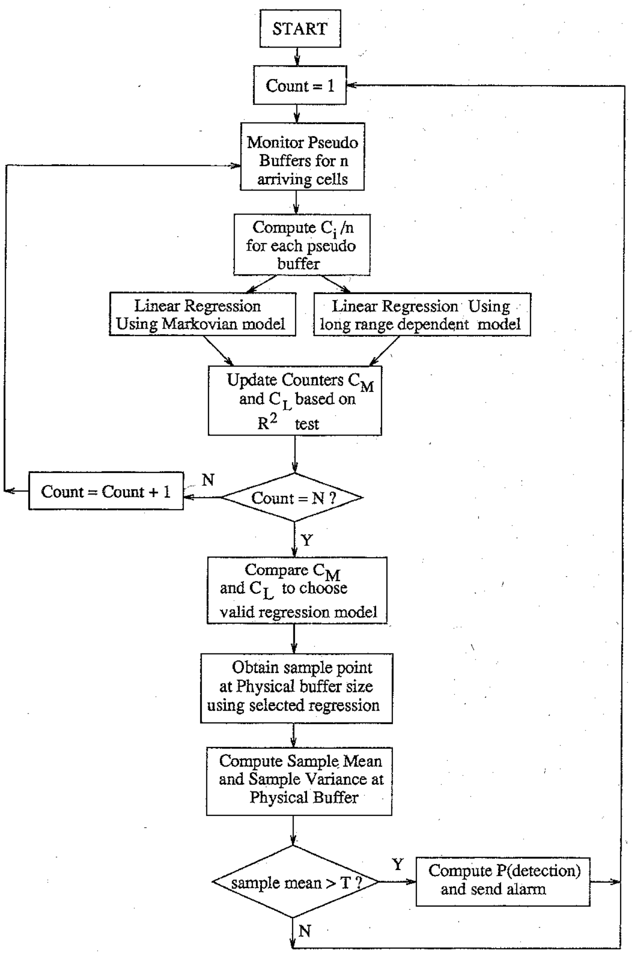

START
Count = 1
Monitor Pseudo Buffers for n arriving cells
Compute C$_i$ /n for each pseudo buffer
Linear Regression Using Markovian model
Linear Regression Using long range dependent model
Update Counters C$_M$ and C$_L$ based on R^2 test
Count = Count + 1
N
Count = N ?
Y
Compare C$_M$ and C$_L$ to choose valid regression model
Obtain sample point at Physical buffer size using selected regression
Compute Sample Mean and Sample Variance at Physical Buffer
sample mean > T ?
Y
Compute P(detection) and send alarm
N

Fig. 5. ISME Procedure for CLR QoS.

IV. Performance Evaluation of the QoS Violation Detection Algorithm

A. Description of Simulation Environment

In this section, the performance of the CLR QoS violation detection scheme is evaluated using simulations with modeled and traced traffic data. The goals of simulation study are to:

- validate the effectiveness of the detection scheme for a variety of traffic types, and
- evaluate total number of cells needed for detection for adequate performance.

Two standard queueing models and two traces of real traffic data are used here. The standard queueing models are the Geom/Geom/1/K model and 32 aggregated homogeneous on/off sources with single server queue and fixed service time for each cell. The real traffic sources are Bellcore video source and Bellcore LAN trace data [7], [8].

Simulations are designed using three different traffic loads ρ_1, ρ_2, and ρ_3, where ρ_1 is assumed to be the load under which the CLR QoS is just satisfied, i.e., ρ_1 creates the targeted QoS log(CLR). Notice the physical buffer is assumed to be of a size that causes the QoS CLR to be in the range of (10^{-9},

TABLE I
TRAFFIC TYPES USED IN SIMULATIONS

Traffic Type	ρ_1	ρ_2	ρ_3	CLR QoS	Phy. Buff. size
Geom/Geom/1/K	0.6	0.65	0.7	$10^{-9.5245}$	40
32 On/Off Sources (Burstiness Index 6)	0.55	0.6	0.65	$10^{-8.5600}$	200
Bellcore Video Trace	3 sources	$1.1\rho_1$	$1.2\rho_1$	$10^{-10.0614}$	300
Bellcore LAN Trace	— —	$1.1\rho_1$	$1.2\rho_1$	$10^{-12.6046}$	280

TABLE II
PERCENTAGE OF TIMES MARKOVIAN MODEL WAS CHOSEN FOR REGRESSION

	% of times Markovian model is chosen		
Traffic Type	ρ_1	ρ_2	ρ_3
Bellcore Video	17%	15%	19%
Bellcore LAN	25%	20%	23%
32 On/Off Sources	85%	83%	90%
Geom/Geom/1/K	80%	84%	82%

10^{-11}), and ρ_2 and ρ_3 are loads that cause the CLR QoS to be violated. The relationship between these three loads is $\rho_1 > \rho_2 > \rho_3$. Pseudo-buffer sizes of seven, 10, and 13 are used for simulations for Geom/Geom/1/K and on-off models, while pseudo-buffer sizes of seven, 10, 13, and 16 are used for simulations for Bellcore Ethernet and video trace data. Note, using more pseudo-buffers would have the advantage of reducing the regression error and would assist in characterizing the nonlinear relationship of $\log(\text{CLR})$ versus buffer size for the fractional Brownian motion model. However, using more pseudo-buffers may result in a longer observation period because CLR decreases at larger pseudo-buffer sizes.

First, for each traffic type, a simulation under ρ_1 is performed using $n = 10\,000$ and $N = 100$ which provides 100 samples at each pseudo-buffer. Then, linear regressions (based on the chosen model) are performed to obtain 100 samples of $\log(\text{CLR})$ at the physical buffer size. The Q value is set to be the sample mean of these 100 samples, and $\sigma^2_{\overline{L}|H_0}$ can also be obtained from these 100 samples. The extrapolated value of Q is close to the simulated value. For example, using the results presented in Fig. 2, for Bellcore video trace data, an extrapolated CLR at buffer size of 60 is 4.98×10^{-5}, while the simulated CLR at the same buffer size is 5.2×10^{-5}; for Bellcore Ethernet trace, an extrapolated CLR at buffer size of 60 is 2.3566×10^{-5}, while the simulated CLR at the same buffer size is 1.7×10^{-5}. The alarm threshold T is found according to the detection algorithm. Table I summarizes the traffic types used in simulations and the corresponding parameters associated with these traffic types.

Then, for each traffic type, simulations under ρ_2 and ρ_3 are performed. The T value is used to decide if ρ_2 and ρ_3 cause CLR QoS violation based on the decision rule described in the last section.

A question may arise about which of the two regression models was chosen by the R^2 test for each simulation. Table II presents the percentage of times in N regressions the Markovian model was chosen to be the valid regression model.

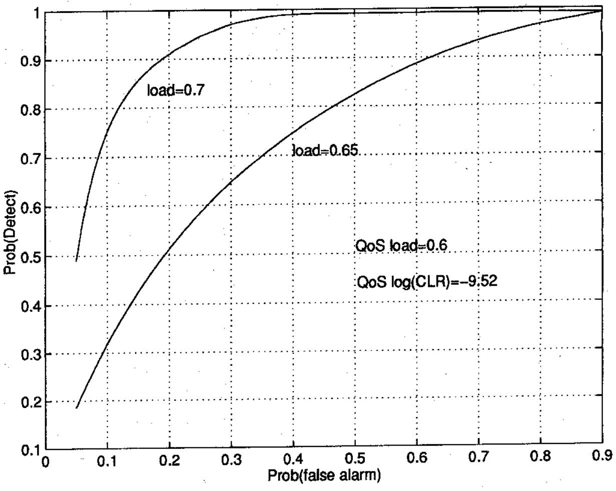


Fig. 6. ROC for Geom/Geom/1/K queueing model.

From these results, it is obvious that the long-range dependent model was chosen to be the valid model for all simulations associated with Bellcore video and Ethernet trace data while the Markovian model was chosen to be the valid model for all simulations associated with on/off and Geom/Geom/1/K sources.

B. Validation of Effectiveness of the Detection Algorithm

We evaluate the effectiveness of the detection scheme by presenting the receiver operating characteristic (ROC) for each traffic type. The ROC is a plot of P_D versus P_F. As described above, all simulations are conducted using fixed $n = 10\,000$, and $\sigma^2_{\overline{L}}$'s are estimated using simulated data. For all ROC's shown here, $N = 1$ is used. Notice $N = 1$ would present a worst-case performance for the detection algorithm. In the real situation, N must be large (>30) in order to make a Gaussian assumption for $\overline{L}$ as well as to obtain a good estimation of $\sigma^2_{\overline{L}}$. More realistic values of N and the corresponding P_F and P_D will be considered later.

Figures 6–9 show the ROC curves for four traffic types. All ROC curves are plotted using $N = 1$. Large N can be expected to significantly improve the detector performance as will be shown later. During simulation, the Bellcore video source was segmented to form several homogeneous video sources. Each video source was then segmented into ATM cells. In Fig. 8, ρ_1 is the traffic load of three video sources which causes the QoS CLR to be $10^{-9.5437}$. Bellcore LAN trace data is also segmented into ATM cells, and ρ_1 is the load that causes the QoS CLR to be $10^{-10.0044}$. In both Bellcore video and LAN trace data simulations, ρ_2 and ρ_3 are achieved by scaling the service time used in ρ_1. ρ_2 is $1.1 \times \rho_1$, and ρ_3 is $1.2 \times \rho_1$.

C. Evaluation of Monitoring Period Requirement

The total number of cells needed for the detection algorithm is $n \times N$. Generally speaking, increasing N will increase P_D. Since P_D is upperbounded by one, it can be expected that beyond a certain N, P_D will not increase significantly. Because of this fact, one would be willing to specify a desired

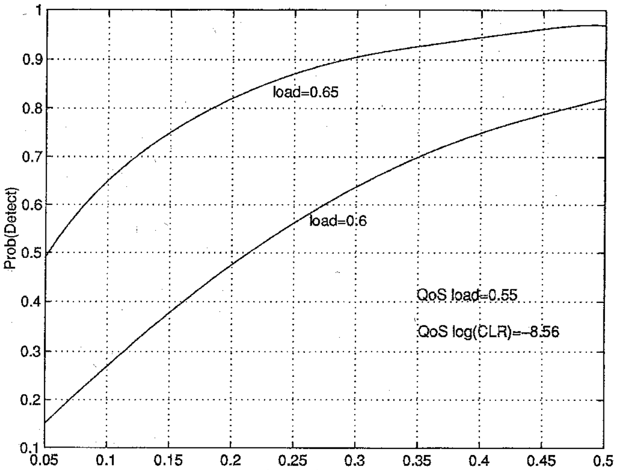


Fig. 7. ROC for 32 Homogeneous on/off sources. Burstiness index is six.

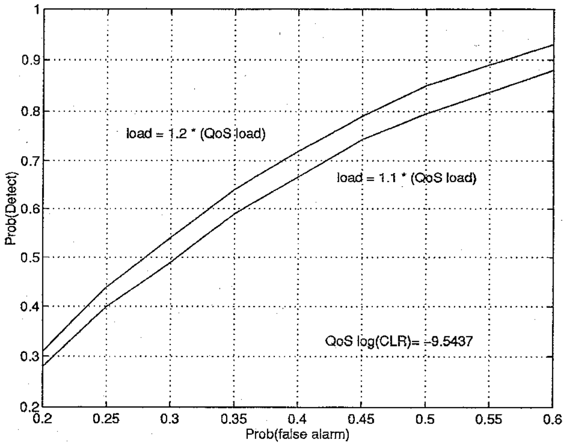


Fig. 8. ROC using Bellcore video source.

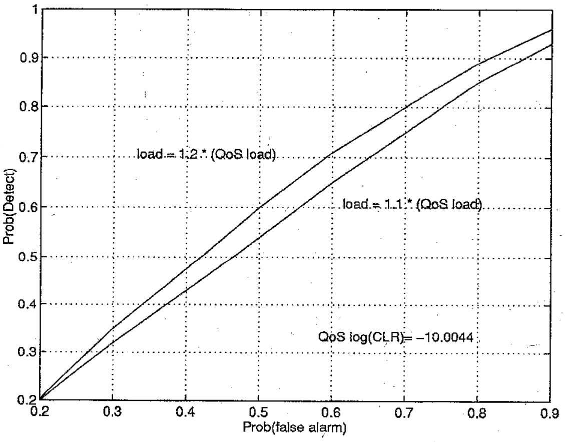


Fig. 9. ROC using Bellcore LAN trace source.

P_D and find the N needed to achieve this P_D. The general approach for studying the required value of N is to plot P_D versus total cells observed with several fixed values of P_F.

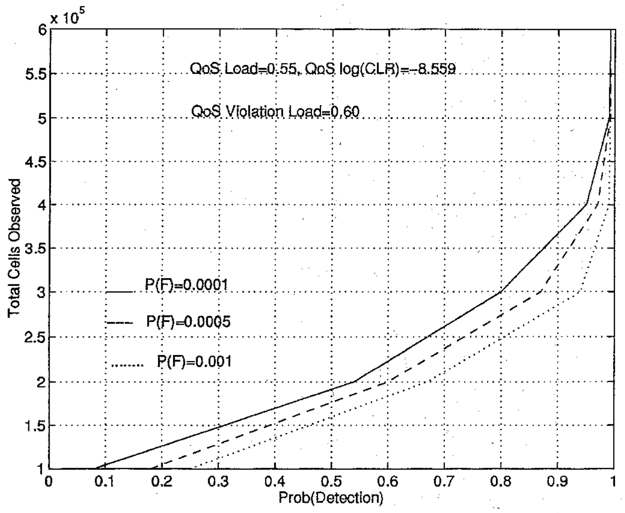


Fig. 10. P_D versus total cells observed, homogeneous on/off sources.

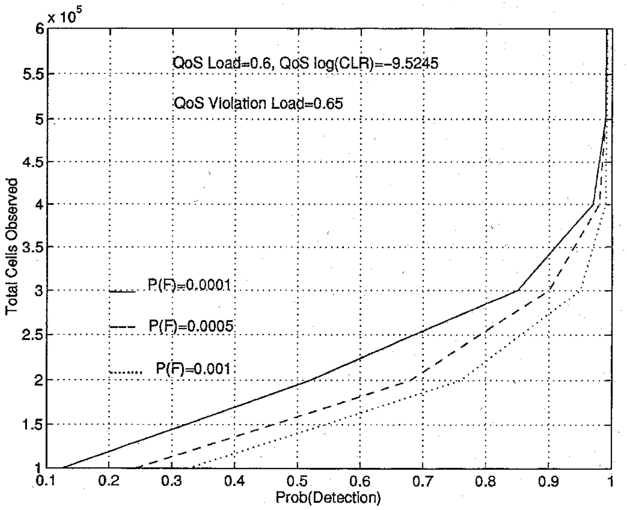


Fig. 11. P_D versus total cells observed, Geom/Geom/1/K model.

Figures 10–13 show the relationship between P_D and total cells observed for the traffic types considered here. These figures clearly show the effectiveness of the proposed algorithm, i.e., a small increment in load can be quickly detected (small $N \times n$) with a high P_D and low P_F. These figures also show the trade-off between observation period and P_D. In Fig. 10, for example, for $P_F = 0.0001$, when $(n \times N)$ increases from 30×10^4 to 40×10^4, the P_D increases from 0.8 to 0.95. However, increasing $(n \times N)$ from 50×10^4 to 60×10^4 increases P_D from 0.99 to 0.991 which is only 0.1% of an increment. In this case, one would not hesitate to sacrifice a very small improvement in P_D in order to reduce the monitoring period. Therefore, one would determine desired P_D to be 0.99 and use a monitoring period of 50×10^4 cells.

Figures 10–13 all contain three curves for $P_F = 0.0001$, $P_F = 0.0005$, and $P_F = 0.001$, respectively. For Geom/Geom/1/K and on/off traffic sources, 5×10^5 cell arrivals must be observed to detect a load increment of 0.05

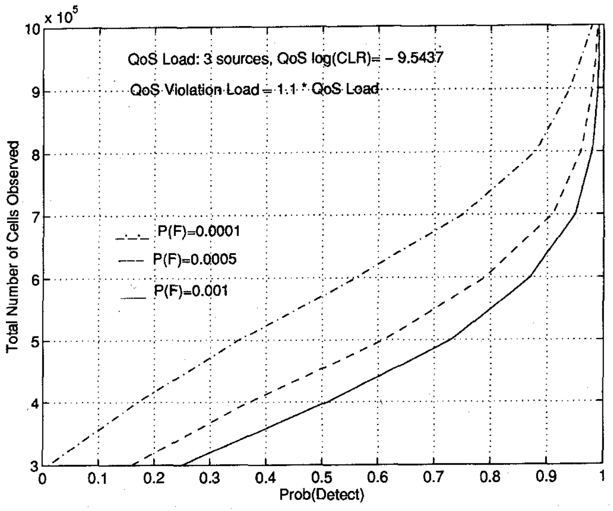


Fig. 12. P_D versus total cells observed, Bellcore video sources.

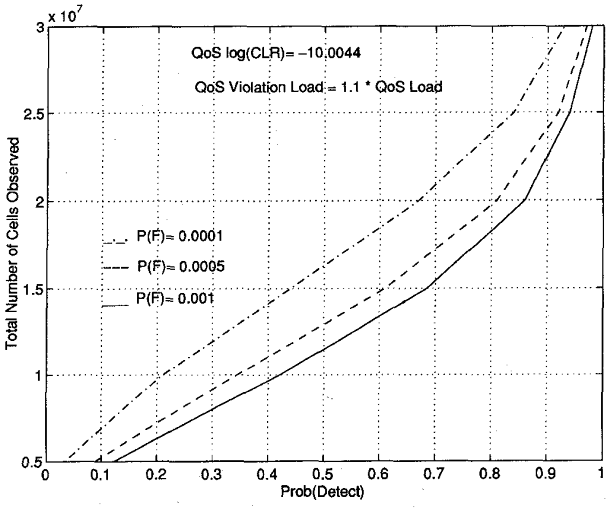


Fig. 13. P_D versus total cells observed, Bellcore LAN trace source.

with $P_D > 0.99$. Also, as can be seen from Fig. 12, to detect a 10% load increment for the Bellcore VBR video trace data with $P_D > 0.98$ and $P_F < 0.0001$, a total number of 10×10^5 cells need to be observed. In Fig. 13, to detect a 10% load increment for the Bellcore LAN trace source, with $P_D > 0.95$ and $P_F < 0.0001$, a total number of 30×10^6 cells are needed. Table III illustrates the savings in total number of cells needed by using our ISME scheme. In Table III, the total number of cells needed when not using the scheme is approximated by using the value of Q. For example, if Q is 10^{-9}, then the total number of cells needed without the scheme is assumed to be 10^{10}, i.e., at least 10 loss events are required. Notice that 10 independent loss events will only produce an estimate valid within a factor of two, thus, the monitoring period approximated in this way may not guarantee a P_F of 0.0001 and a P_D of 0.95. These numbers represent the minimum feasible observation interval when not using the ISME scheme.

TABLE III
COMPARISON STUDY, $P_F = 0.0001$, $P_D > 0.95$

		Total Cells Observed	
	Load Increase	Without Scheme	With Scheme
Geom/Geom/1/K	$\Delta\rho = 0.05$	10^{10}	5×10^5
Geom/D/1/K	$\Delta\rho = 0.05$	10^{11}	4×10^5
On/Off Source	$\Delta\rho = 0.05$	10^9	5×10^5
Video Source	×1.1	10^{11}	10×10^5
LAN Trace Source	×1.1	10^{13}	30×10^6

TABLE IV
32 HOMOGENEOUS ON/OFF SOURCES

P_F		P_{D_1}		P_{D_2}	
Predicted	Simulation	Predicted	Simulation	Predicted	Simulation
0.1	0.09	0.27	0.27	0.65	0.66
0.2	0.2	0.47	0.48	0.82	0.83
0.3	0.31	0.65	0.65	0.91	0.92
0.4	0.43	0.75	0.77	0.95	0.94
0.5	0.51	0.82	0.83	0.98	0.96

TABLE V
Geom/Geom/1/K MODEL

P_F		P_{D_1}		P_{D_2}	
Predicted	Simulation	Predicted	Simulation	Predicted	Simulation
0.1	0.08	0.33	0.33	0.75	0.77
0.3	0.38	0.65	0.67	0.97	0.96
0.5	0.48	0.82	0.80	0.99	1.0
0.7	0.68	0.93	0.90	0.993	1.0
0.9	0.9	0.99	0.98	0.995	1.0

The saving factors of total number of cells needed are in the order of 10^5–10^7. For example, a $Q = 10^{-9}$ and a link rate of 155 Mb/s with an arrival probability of 0.6 in each time slot requires about 12 h while the ISME technique needs only 2–35 (2–3 s). However, without the scheme, the monitoring period takes more than 12 h. The saving on monitoring period is tremendous.

D. Validation of the Gaussian Assumptions

In the detection algorithm, a key assumption is that $\overline{L}$ is normally distributed. It has been claimed, based on the central limit theorem, that when $N > 30$, this assumption should be quite good. It is of interest to validate this assumption by comparing the simulated P_D and P_F with the P_D and P_F predicted from Gaussian distribution. To simulate the P_D and P_F, a large number of monitoring periods should be observed. A reliable simulation would require an observation of a very large number of cells. Unfortunately, when $N > 30$, the time required to simulate is too long, and therefore, $N = 1$ was used. Notice when $N = 1$, $\overline{L}$ reduces to L and the distribution of L may not be normal. However, if the simulated P_D and P_F using $N = 1$ are close to those predicted from Gaussian assumption, then the Gaussian assumption for $\overline{L}$ is expected to be valid for large N.

Tables IV–VII show the comparison of predicted results to the simulated results for Geom/Geom/1/K, Geom/D/1/K, on/off models, and Bellcore video sources. From the results presented here, we are confident that when N is large (> 30), the Gaussian approximations will be valid.

TABLE VI
BELLCORE VIDEO SOURCE

P_F		P_{D_1}		P_{D_2}	
Predicted	Simulation	Predicted	Simulation	Predicted	Simulation
0.1	0.08	0.21	0.18	0.78	0.88
0.2	0.23	0.42	0.38	0.92	1.0
0.3	0.32	0.64	0.69	0.98	1.0
0.4	0.45	0.78	0.82	0.99	1.0
0.5	0.6	0.90	0.92	0.992	1.0

TABLE VII
BELLCORE LAN TRACE DATA

P_F		P_{D_1}		P_{D_2}	
Predicted	Simulation	Predicted	Simulation	Predicted	Simulation
0.1	0.08	0.09	0.10	0.08	0.10
0.3	0.25	0.33	0.36	0.36	0.40
0.5	0.47	0.55	0.58	0.61	0.65
0.7	0.65	0.76	0.79	0.82	0.82
0.9	0.85	0.93	0.96	0.96	1.0

V. CONCLUSION

In this paper, a CLR QoS violation detection scheme has been developed and shown to be effective for both measured trace and modeled traffic data. As long as a curve can be fit to the buffer size versus $\log(\mathrm{CLR})$ characteristic, the approach proposed here can be applied. In this paper, the relationship between buffer size and CLR for both the conventional Markovian traffic models and the recently proposed fractional Brownian motion model were taken into account. The performance of the scheme and its underlying approximations have been validated. It has been shown that the scheme is sensitive to small increments in load.

The scheme aims to be implemented in a real-time environment where telecommunication services will not be interrupted by monitoring procedures. Real-time monitoring can be achieved using this scheme, because the implementation is simple, sensitive, and the monitoring period is short; and enabling network management systems to be promptly informed of impending CLR QoS violations.

REFERENCES

[1] H. Bruneel, E. Desmet, B. Steyaert, and G. H. Petit, "Tail distribution of queue length and delay in discrete-time multiserver queueing models, applicable in ATM networks," ITC-13, 1991.

[2] A. M. Breipohl and K. S. Shanmugan, *Random Signals Detection, Estimation and Data Analysis.* New York: Wiley, 1992.

[3] H. Murakami, R. E. Mallon, S. R. Hughes, N. Sato, K. Asatani, and T. L. Graff, "In-service monitoring methods-better ways to assure quality of digital transmission," *IEEE J. Select. Areas Commun.*, Feb. 1994.

[4] T. G. Robertazzi, *Computer Networks and Systems,* 2nd ed. New York: Springer-Verlag, 1994.

[5] K. Sam Shanmugan, *BONeS Designer, Introductory Overview,* Comdisco Systems Inc., 1992.

[6] Q. Wang, "New solution techniques for performance analysis of ATM networks," Ph.D. dissertation, Univ. Kansas, 1992.

[7] J. Beran, R. Sherman, and W. Willinger, "Long range dependence in variable-bit-rate video traffic," *IEEE Trans. Commun.*, vol. 43, no. 2/3/4, 1995.

[8] W. Willinger, W. E. Leland, M. S. Taqqu, and D. V. Wilson, "On the self-similar nature of ethernet traffic (extended version)," *IEEE/ACM Trans. Networking*, 1994.

[9] H. Yamada and S. Sunita, "A traffic measurement method and its applications for cell loss probability estimation in ATM networks," *IEEE J. Select. Areas Commun.*, vol. 9, no. 3, pp. 315–326, 1991.

[10] H. Ahmadi and W. Denzel, "A survey of modern high performance switch techniques," *IEEE J. Select. Areas Commun.*, vol. 7, no. 7, 1989.

[11] M. Hluchyj and M. J. Karol, "Queueing in high performance packet switching," *IEEE J. Select. Areas Commun.*, vol. 6, no. 9, 1988.

[12] D. Anick, D. Mitra, and M. M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *Bell Syst. Tech. J.*, vol. 61, no. 8, 1992.

[13] A. Bhargava and M. Hluchyi, "Frame losses due to buffer overflows in fast packet networks," in *Proc. IEEE INFOCOM* June 1990.

[14] R. Guérin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, Sept. 1991.

[15] H. Zhu and V. S. Frost, "A new method for in-service estimation of cell loss QoS in ATM networks," presented at *IEEE Symp. Planning Design Broadband Networks*, Montebello, Quebec, Canada, Oct. 21–23, 1994.

[16] M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, *Simulation of Communication Systems.* New York: Plenum, 1992.

[17] C. Courcoubetis, G. Kesidis, A. Ridder, J. Walrand, and R. Weber "Admission control and routing in ATM networks using inferences from measured buffer occupancy," *IEEE Trans. Commun.*, vol. 43, p. 1778, 1995.

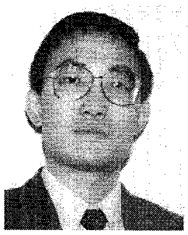
[18] N. G. Duffield, J. T. Lewis, N. O'Connell, R. Russell, and F. Toomey, "Statistical issues raised by the Bellcore data," presented in *11th UK Teletraffic Symp.*, 1994.

[19] N. G. Duffield and N. O'Connell, "Large deviations and overflow probabilities for the general single-server queue, with applications," in *Proc. Cambridge Phil. Soc.*, 1995.

[20] R. Dahlhaus, "Efficient parameter estimation for self-similar processes," *Ann. Statist.*, vol. 17, pp. 1747–1766, 1989.

[21] M. Nomura, T. Fujii, and N. Ohta, "Basic characteristics of variable bit rate video coding in ATM environment," *IEEE J. Select. Areas Commun.*, vol. 7, no. 5, pp. 752–760, 1989.

[22] A. Elwalid, D. Heyman, T. V. Lakshman, D. Mitra, and A. Weiss, "Fundermental bounds and approximation for ATM multiplexers with applications to video teleconferencing," *IEEE J. Select. Areas Commun.*, Aug. 1995.



Hongbo Zhu (S'94) received the B.S.E.E. and M.S.E.E. degrees from the University of Kansas, Lawrence, in December 1993 and May 1995, respectively.

Since May 1994, he has been a Research Assistant in the Telecommunications and Information Sciences Laboratory, the Electrical Engineering and Computer Science Department, University of Kansas, working on Sprint, BNR, and ARPA projects related to ATM technology. His current research interests are in the general areas of communications networks with an emphasis on congestion control and resource allocation in B-ISDN/ATM.

Victor S. Frost (S'75–M'82–SM'90) was born in Kansas City, MO, on March 6, 1954. He received the B.S., M.S., and Ph.D. degrees from the University of Kansas, Lawrence, in 1977, 1978, and 1982, respectively.

He joined the faculty of the University of Kansas in 1982, where he is currently a Professor of Electrical Engineering and Computer Science. He has been the Director of the Telecommunications and Information Sciences Laboratory at the University of Kansas since 1987. His current research interest is in the areas of integrated communication networks, high speed networks, communications system analysis, and simulation. He is currently involved in research on the MAGIC, ACTS ATM Internetwork, and SPARTAN ATM WAN testbeds.

Dr. Frost received the Presidential Young Investigator Award from the National Science Foundation in 1984, the Air Force Summer Faculty Fellowship, the Ralph R. Teetor Educational Award from the Society of Automotive Engineers, and the Miller Professional Development Awards for Engineering Research and Service in 1986 and 1991 respectively. He is a member of Eta Kappa Nu and Tau Beta Pi. He served as Chairman of the Kansas City Section of the IEEE Communications Society from June 1991–Dec. 1992.