Traffic Descriptors for VBR Video Teleconferencing Over ATM Networks

Amy R. Reibman and Arthur W. Berger

Abstract—**This paper examines the problem of video transport over ATM networks using knowledge of both video system design and broadband networks. The following issues are addressed: video system delay caused by internal buffering, traffic descriptors (TD) for video, and call admission.**

We find that while different video sequences require different TD parameters, the following trends hold for all sequences examined. First, increasing the delay in the video system decreases the necessary peak rate and significantly increases the number of calls that can be carried by the network. Second, as an operational traffic descriptor for video, the leaky-bucket algorithm appears to be superior to the sliding-window algorithm. And finally, with a delay in the video system, the statistical multiplexing gain from VBR over CBR video is upper bounded by roughly a factor of four, and to obtain a gain of about 2.0 can require the operational traffic descriptor to have a window or bucket size on the order of a thousand cells. We briefly discuss how increasing the complexity of the video system may enable the size of the bucket or window to be reduced.

I. Introduction

WE ENVISION the following scenario for carrying a video call in a Broadband Integrated Services Digital Network (B-ISDN) that is based on Asynchronous Transfer Mode (ATM). When the user decides to initiate a call, the video terminal contacts the network and uses a traffic descriptor to characterize the traffic it intends to submit to the network. The network accepts the call if it can provide the desired quality of service to a call conforming to that traffic descriptor. If the network does not have available resources, it may simply reject the call, or the network and terminal may negotiate alternative parameters for the traffic descriptor and/or the quality of service. If the call is established, the network monitors the submitted traffic to ensure that it does comply with the negotiated traffic descriptor.

In this paper, we describe the influence of the above scenario on the video system. We view the problem of video transport over ATM networks using knowledge of both video system design and broadband networks. While this paper contains material that may be familiar to either video researchers and broadband-network researchers, the unified perspective provides some important insights. An abbreviated version of this paper appears in [1].

Manuscript received September 18, 1992; revised September 15, 1993; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor D. Raychaudhuri.

The authors are with AT&T Bell Laboratories, Holmdel, NJ 07733 USA.

IEEE Log Number 9410804.

We examine the impact that the presence of a network traffic monitor (or Usage Parameter Control (UPC)) has on the video system. We argue that this monitor will force the video system to apply rate control to ensure its submitted traffic indeed meets the negotiated traffic descriptors.

We are interested in the traffic descriptor for a video call in a B-ISDN using ATM. Viewing a traffic descriptor as a means for specifying user traffic levels that will place demands on shared network resources, there are two fundamentally different approaches that can be taken 1) a *statistical* approach, and 2) an *operational* approach, see [2].

The statistical approach is the more conventional in traffic theory, and focuses on statistical traffic parameters, such as the long-term average rate and burst duration. Such an approach may be motivated by the apparent availability of methodologies for predicting performance of a network stressed by traffic with given statistical parameters. However, associated with a statistical framework for traffic is the difficult task of verifying a set of statistical traffic parameters. By the very nature of their definitions, statistical traffic parameters may require a lengthy observation interval for verification, making real-time traffic-compliance-testing nearly impossible. See [3] or [4] for examples of such difficulties.

With the operational approach, little heed is given to traffic at levels well within compliance with traffic contracts; rather focus is placed on traffic that is either just compliant or not compliant. This is achieved by implementing an operational traffic descriptor as a parameterized compliance-testing algorithm intended to discriminate "excessive" traffic from "nonexcessive" traffic. Thus, by focusing on traffic-compliance-testing itself, we immediately resolve the issue of real-time traffic-compliance-testing at the terminal.

Currently, the International Telecommunication Union (ITU, formerly CCITT) has only specified a traffic descriptor for the peak rate, though additional parameters are expected to be specified in the future [5]. The definition of the peak rate is operational, though the specifics of the definition are not of importance for the present paper.[1]

Herein, we begin the investigation of suitable traffic descriptors (TD's) for teleconference calls by comparing two potential TD's: One consisting of a peak rate and a sliding-window algorithm, and the other consisting of a peak rate and a leaky-

[1] In [5] the peak rate of a virtual channel connection and of a virtual path connection is defined to be the reciprocal of the minimum time between requests to send an ATM Protocol Data Unit (a cell) to the Physical Layer Service Access Point in an "equivalent" terminal, where the latter is a general model for end-user equipment.

bucket algorithm. (These algorithms are reviewed in Section V; for more details, see, e.g., [4] and references therein.) Note that herein the sliding-window and leaky-bucket algorithms are not attempting to approximate the peak rate but rather are an additional element of the TD. Also, the algorithms do not attempt to approximate the average rate of the call (number of bits transmitted divided by the duration of the call). The algorithms do determine, though, a rate at which the source could continuously submit compliant traffic; we call this rate the "negotiated average rate." Last, recall that these operational TDs are not intended to be a model that fully characterizes the source traffic; for such models see, e.g., [6]–[9].

For sample teleconference sequences, we determine the parameter values of the leaky bucket and sliding window so that the given video sequence is compliant with the traffic descriptor. Our results not only illustrate the magnitude of the parameter values that would be needed, but also provide a comparison of the two algorithms.

Note that herein we determine the parameters *after* the video sequence has taken place, which, of course, can not be done in a real video call. Since TD parameters must be chosen prior to video compression in a real video call, it is impossible to guarantee that the chosen parameters will allow the VBR stream to pass without constraint. This leads to the question of how can a video terminal emit traffic that is compliant with the traffic descriptor negotiated at call setup. This question is addressed in [10], where Reibman and Haskell presented a joint encoder/channel rate control algorithm. Voeten *et al.* [11] considered a preventive policing mechanism for a video terminal that also addresses this question.

Section II briefly describes the video used for the experimental results in this paper. It also describes some important characteristics of a generic packet video system. In Section III, we describe the notions of traffic descriptors and Usage Parameter Control used in this paper, and illustrate their importance for video systems. In Section IV, we discuss comparisons between CBR and VBR video. Section V presents traffic descriptor parameter values for video. We show that the delay has a significant impact on the peak rate of the video call and consequently on the number of calls that can be carried by the network. Section VI briefly discusses call admission and possible multiplexing gains given the actual video data. Section VII concludes the paper with some discussions about how this information can be applied to the design of real systems.

Throughout this paper, we describe the problem using the frame period as the discrete time unit; however, smaller sampling intervals are possible.

II. Video

A. Experimental video sequences

The video used in these examples was recorded at an actual meeting. The output of a CCD camera was digitized and recorded on a D1 tape machine, in order to create repeatable digital source material. The CCIR601 format video was converted into common intermediate format (CIF) (240 lines) using the MPEG-SM filter[14]. The video was coded using a one-layer codec that has syntax compatible with H.261 [12]. The codec uses exhaustive motion estimation with ± 15 search range, a constant quantizer step-size of 8, and intra/inter/motion-compensation decisions for each macroblock as in Reference Model 8 [13]. The first frame is coded intraframe, and all remaining frames are coded predictively. Within each frame, 3 macroblocks are transmitted intraframe.

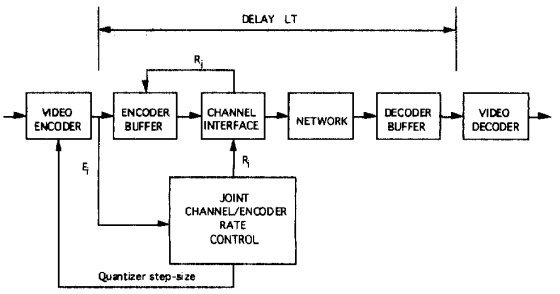

Fig. 1. A generic packet video system.

Eight sequences are used here. Sequence A is 10 min long and consists of a person listening and interspersing occasional comments and questions. Sequences B–F are both 5 min long, and each contains one active participant. In sequence B, the subject is constantly moving, while in sequences C–F, the subject moves only occasionally. Sequence G and H are 3 min 40 s. Sequence G contains one person listening, and sequence H contains two people. Detailed results are presented for only sequences A–C to conserve space.

In all the sequences, intervals with high activity do not necessarily imply that the subject is speaking. Often, a high activity region corresponds to a period in which the subject is silent but moving, and a low activity region corresponds to when they speak.

B. A Generic Packet Video System

The system we consider is shown in Fig. 1. A video signal is applied to the video encoder, which produces an encoded video bit-stream. The number of encoded bits produced by frame i is E_i. The encoded bit-stream is stored in the encoder buffer before being transmitted via the channel interface to the network. The rate control device selects R_i, the number of bits transmitted on the channel during frame period i, such that no video buffer constraints will be violated and no traffic that exceeds the negotiated TD parameters is submitted to the network [10]. In addition, the rate control device also selects the quantizer step-size used by the encoder. After being transmitted across the network, the video bit-stream is stored in the decoder buffer. It is then input to the video decoder, which outputs a video signal. For a CBR video system, everything is identical except the traffic descriptor specifies a constant bit rate.

A delay may be necessary in the video system to guarantee that the decoder will have access to all the bits corresponding to frame i by the time that frame needs to be displayed. The

delay is defined by the interval between the time the decoder receives the first bit at the start of the call until the time the decoder begins to decode. Once decoding commences, a frame must then be displayed every $T = 1/30$ seconds. Thus, the value of the delay, given by LT seconds where L is a non-negative real number, must be known *a priori*, at both the encoder and the decoder. Moreover, during the course of the call, since the video bit-stream may experience variable delay in the encoder buffer and in the network, there is a variable delay between the time the video is encoded and it arrives at the decoder buffer. Therefore, to ensure bits are available so that a frame can be decoded every T seconds, there must be a variable delay in the decoder buffer. However, the per-frame delay from the encoder buffer input to the decoder buffer output is constant and equal to LT plus the realized delay of the ATM cell that carries the first bits of the video transmission.

III. VBR Video and Network Policing

Variable bit-rate (VBR) video is expected to be advantageous both for the network and for the user. Through statistical multiplexing, the network should be able to carry more VBR video calls than constant bit-rate (CBR) video calls. Alternatively, the user is expected to obtain better quality with VBR video than CBR video, even when both systems have the same average rate.

However, if all streams have completely unconstrained bit-rate, the network either could not ensure a quality of service for established connections, or would be under-utilized due to a very conservative call admission policy. Hence, some form of service contract between the network and the user is necessary, along with traffic and congestion controls by end-system and network, see for example [15]. The user understands two things from the service contract. First, the network *will* transport (with agreed-to cell-loss rate (CLR)) any traffic the user submits that is compliant with the negotiated traffic descriptor. Second, if excess traffic is submitted, the network has the option of not transporting it. For video, some information is vital to the decoder and should not be dropped; therefore, the video terminal *must not* submit excess traffic as high priority.

The network enforces the negotiated traffic descriptor via the usage parameter control (UPC) function, informally known as the policing function. The UPC serves to protect the network and other users from malicious users or malfunctioning terminals. The UPC includes a monitoring algorithm for the incoming traffic and a control action that is applied to the excessive traffic. The control action could be either to immediately drop excessive traffic, or to mark excessive traffic as low-priority provided excessive traffic is within certain limits. For the sake of this paper, the control action to be performed is immaterial. We argue that neither the immediate loss of a high-priority cell nor its possible later loss can be allowed to happen. The video system must therefore ensure that all high-priority traffic submitted to the network is compliant with the negotiated traffic descriptor. Thus, the importance of the UPC for the present work is that its existence causes the video system to control the (high priority) traffic to be compliant with the negotiated traffic descriptor and that this control necessitates buffering in the video system or adjustment of quantizer step-size, or both. (Note that in this paper the leaky-bucket and sliding-window algorithms are part of the *traffic descriptor*—whether the UPC uses the same or different algorithms is not addressed.) We assume that the network will not shape the user's traffic to conform to the traffic descriptor.

IV. VBR Video Versus CBR Video

There are three primary advantages to VBR video compared to CBR video: shorter delay, better quality, and better statistical multiplexing. Any one of these advantages may be possible, but it is unlikely that all can be obtained simultaneously. In this paper, we examine the potential statistical multiplexing gain (SMG) of VBR video, and we equate the delay in both the VBR and CBR systems and attempt to make the video quality of both roughly equivalent.

The compressed video produced by most codecs is by nature VBR. It is converted into CBR for transmission across today's circuit switched channels by using an encoder-decoder buffer pair with feedback. The buffers induce a delay within the video system, and the rate-control feedback produces variable quality.

On the surface, VBR video should be simple to obtain; simply remove the encoder buffer and disconnect the feedback loop that controls the bit-rate by varying the quantizer step-size. However, as we saw in Section III, the high-priority bit-rate must never exceed the negotiated TD, so the VBR output of a video codec can not be completely unconstrained. A feedback loop will still be necessary, although it will not be exercised as frequently as for CBR.

To obtain comparable delay, we consider the delay induced by the buffers in the video system (see Fig. 1) and make the simplifying assumption that the network delay is comparable for both CBR and VBR video.

However, comparing the quality of VBR and CBR video systems is difficult since the relative quality between the two systems varies as a function of time. The VBR codec will have (nearly) constant quality, while the CBR codec will have variable quality since the quantizer step-size will vary with the image content. Therefore, while one may seem better momentarily, the reverse may be true a few seconds later.

A typical method to compare CBR and VBR video is to generate an unconstrained VBR video bit-stream and to set the rate of the CBR video to be equal to the peak rate of the VBR bit-stream. This is illustrated in Fig. 3 where the CBR bit-stream fills in the valleys between VBR peaks by reducing the quantizer step-size and thus increasing the instantaneous bit-rate. However, by doing so, the CBR video will undoubtedly have superior quality. Also, the delay will not be comparable, given the presence of delay in the CBR codec.

However, given that we are imposing comparable delays, we can use the buffer in the VBR codec to reduce the peak rate. In the appendix, we present an algorithm that generates a VBR bit-stream with minimum peak rate by making use of

TABLE I
STATISTICS FOR EIGHT SEQUENCES

Sequence	Mean rate (kb/s)	Video delay (frames)	Peak rate R_{max} (kb/s)	Peak to mean ratio	$\lfloor \frac{C}{R_{max}} \rfloor$
A	239.6	0	2561	10.69	49
		1	1400	5.84	91
		2	934	3.90	136
		3	847	3.54	150
		4	822	3.43	155
B	849.9	0	3376	3.97	37
		1	2113	2.49	60
		2	1617	1.90	78
		3	1561	1.84	81
		4	1526	1.80	83
C	351.1	0	2340	6.66	54
		1	1346	3.83	94
		2	924	2.63	138
		3	897	2.56	142
		4	880	2.51	144
D	351.1	0	2340	6.66	54
		3	924	2.63	137
E	320.1	0	2972	9.28	42
		3	823	2.57	154
F	322.1	0	2354	7.31	54
		3	824	2.56	154
G	216.4	0	2644	12.22	48
		3	924	4.26	138
H	214.2	0	4004	18.69	31
		3	1054	4.92	120

available buffering and delay. Throughout the following, we refer to the resulting constrained VBR bit-stream as the VBR bit-stream with delay. By using the algorithm in the appendix, the peak rate of the VBR bit-stream with delay is significantly smaller than the peak rate of the VBR stream without delay. Table I illustrates the reduction in the peak rate as the delay increases, for all eight sequences.

Ideally, to find the appropriate rate for the CBR video such that the quality is roughly equal to that of the VBR video, subjective tests for many choices of the CBR rate may be necessary [16]. Since this is quite difficult and tedious, in this paper we make the simple choice of setting the rate of the CBR to be equal to the peak rate of the VBR bit stream with delay. As our choice is somewhat arbitrary, we compare the resulting quality via plots of the peak signal-to-noise ratio (PSNR)[2] and via subjective tests.

In comparing the PSNR's of CBR and VBR encoded bit streams from multiple example sequences, we find that the PSNR for VBR is nearly constant, while the CBR PSNR fluctuates wildly. Typically, there are alternating intervals of varying length where the PSNR from the CBR video is greater than that from the VBR video and where the PSNRs are roughly equal. Fig. 2 shows a typical example for sequence B, where the rate for the CBR was chosen as discussed above. The two sequences have identical PSNRs for several 10 s intervals, which may be long enough to be the determining factor in a viewer's subjective evaluation.

For a more direct examination of the relative quality in the context of the present study, we conducted subjective tests for three example sequences using both expert and non-expert viewers. When the CBR video rate is the peak rate of the VBR bit-stream with delay, the quality of the CBR video is generally somewhat better than the VBR video in the subjective tests. However, typically, the preference for the CBR video was not strong. Undoubtedly, the quality is closer to being equivalent than when the CBR rate is the peak rate of unconstrained VBR. For the present work, we will consider the quality of the CBR and VBR video as roughly equivalent. Heuristically, this rough equivalence has the "same level of accuracy" as the approximate model we use to estimate the SMG, see Section VI, and thus is a reasonable assumption to make for the present study. In a more detailed study, to obtain quality that is more equivalent, the CBR rate should be chosen lower than was done herein, though how much lower requires further study.

[2]The PSNR is an objective measure for video quality. It provides a measure of the closeness of a coded frame to its original uncoded version. In general, larger values imply better quality. However, larger values of PSNR may not translate into improved subjective quality. Furthermore, the PSNR only measures quality for a given frame, and does not provide a measure of the temporal quality.

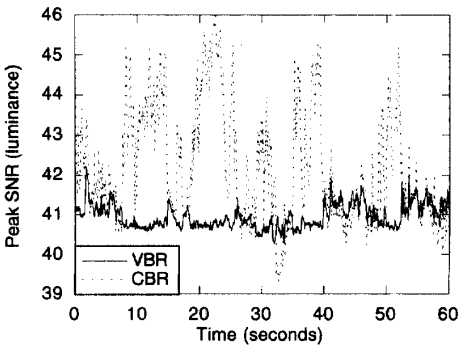

Fig. 2. Quality comparison between CBR and VBR.

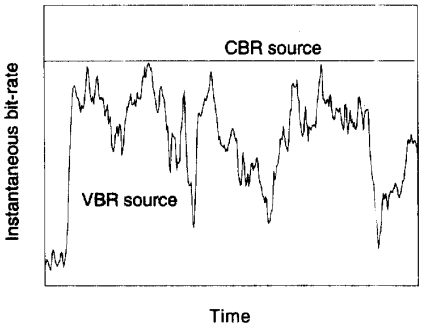

Fig. 3. Bit rate comparison between CBR and VBR.

As an interesting side note, Tan *et al.* [16] did a subjective evaluation of CBR and VBR and found that a CBR source with rate 1.84 times the VBR mean rate had comparable quality to an unbuffered VBR source with peak-to-mean ratio of 4. As shown in Table I, we found the same statistical parameters for sequence B with a delay of $L = 3$ frames.

V. TRAFFIC DESCRIPTOR PARAMETER SELECTION

As mentioned in the introduction, we compare two potential traffic descriptors for a video call: One consisting of a peak rate and a sliding-window algorithm, and the other consisting of a peak rate and a leaky-bucket algorithm. We define the peak rate to be $R_{max} = \max_i R_i$.

Furthermore, for the simplicity of the mathematical descriptions below, we assume the size units are measured in bits and

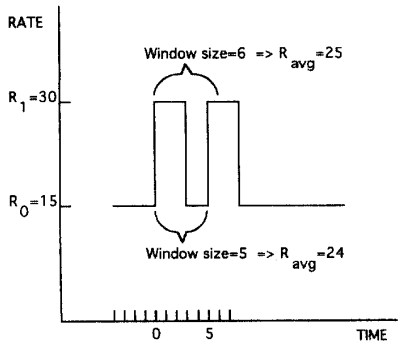

Fig. 4. Example of sliding window.

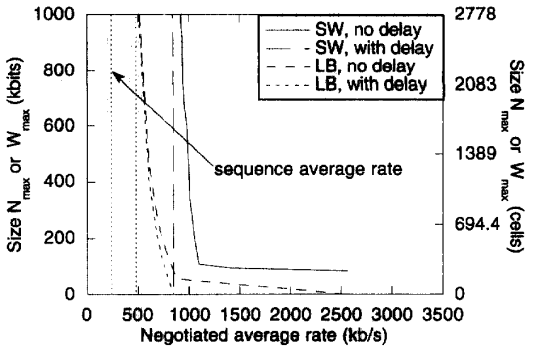

Fig. 5. TD parameters, sequence A.

the time-interval units are measured in frame periods. These can be easily converted into size units of bytes or cells, and time units of seconds.

A. Sliding Window

A sliding window specifies that no more than a given number of bits (or cells) can be emitted in a time interval of a specified length, where the time interval can begin at any epoch. The sliding window can be described by two parameters, the time duration S_{win}, and the maximum number of bits that can be transmitted in that time window, W_{max}. An alternate description could use W_{max} and the negotiated average bit-rate, $\bar{R} = W_{max}/S_{win}$. Mathematically, the constraint on the channel rate that is imposed by the sliding window is

$$\sum_{j=k+1}^{k+S_{win}} R_j \leq W_{max}, \tag{1}$$

for all k.

For a given sequence of R_i, it is a simple matter to determine the size of the sliding window parameters necessary to pass a given bit-stream without violation by computing the maximum number of bits in any window for each window length of interest.

The negotiated average rate does not necessarily decrease monotonically as the window size increases. Fig. 4 shows an example in which the negotiated average rate increases as the window size increases from 5 to 6. Therefore, using a larger window size may actually decrease the transmission efficiency.

B. Leaky Bucket

A leaky bucket is a counter that increments by one for each cell emitted, up to a maximum value, and decrements at a given rate to as low as zero—a cell can be emitted if the counter is less than the maximum value minus one.

The leaky bucket can be considered as an imaginary FIFO buffer of size N_{max} bits with constant drain rate $\bar{R}$ bits per frame period. Let N_i be the bucket fullness (in bits) at the end of frame period i. Since R_i bits arrive in frame period i,

$$N_i = \max\{0, N_{i-1} + R_i - \bar{R}\}, \tag{2}$$

if we ignore the finite capacity of the bucket. Thus, the traffic would be in compliance if we choose the bucket capacity N_{max} such that $N_i \leq N_{max} \forall i$.

C. Examples

We present examples both with and without delay in the video system. Without delay, the number of transmitted bits per frame period is equal to the number of encoded bits in that frame period, $R_i = E_i$. With delay, the R_i are obtained using the smoothing algorithm in the appendix.

Table I shows the peak rate R_{max} and mean rate for 8 teleconferencing sequences. In the last column, C denotes the capacity of the slowest link in the connection. In B-ISDN/ATM a physical layer of 155.52Mb/sec (e.g. SONET STS-3c) provides the transmission capacity of 149.760Mb/sec to the ATM layer. We assume that the video ATM Adaptation Layer (AAL) uses three bytes, and given the five bytes of ATM header in the 53 byte cell, the bit rate available for the video information, C, is 149.760 x 45/53 = 127.155Mb/s. The notation $\lfloor x \rfloor$ in the last column means the largest integer less than or equal to x. Thus, $\lfloor \frac{C}{R_{\max}} \rfloor$ equals the number of VBR video connections that can be carried on the link, assuming a peak rate allocation. Likewise, $\lfloor \frac{C}{\bar{R}_{\max}} \rfloor$ is the number of CBR video connections that can be carried, assuming the encoder controls the bit rate to be $R_{\max}$.

Table I shows how increasing the delay in the video codecs decreases the peak rate and increases the number of video connections that can be carried. For example, for sequence A, if the delay is increased from zero to three frames, then the number of CBR connections that can be carried on a link increases three fold from 49 to 150.

Figs. 5–7 show the parameter values of the sliding window and the leaky bucket that guarantee the sequences are compliant to the traffic descriptor for sequences A–C. The negotiated average rate $\bar{R}$ is plotted as a function of the "size of the TD." Herein, we use the phrase the "size of the TD" to refer to $W_{\max}$ for the sliding window algorithm and $N_{\max}$ for the leaky bucket algorithm. (To show the parameters of the leaky bucket and the sliding window on the same plot, we describe the sliding window in terms of the window size, $W_{\max}$, and the negotiated average rate $W_{\max}/S_{win}$. To obtain the window length, S_{win}, simply divide $W_{\max}$ by the negotiated

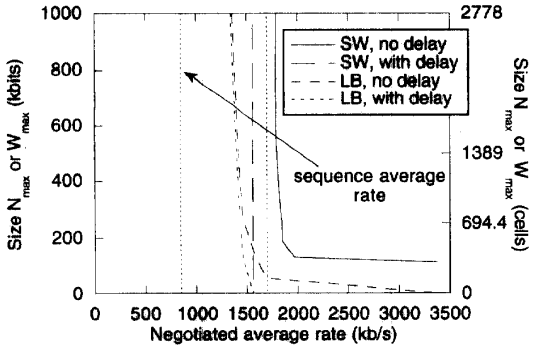

Fig. 6. TD parameters, sequence B.

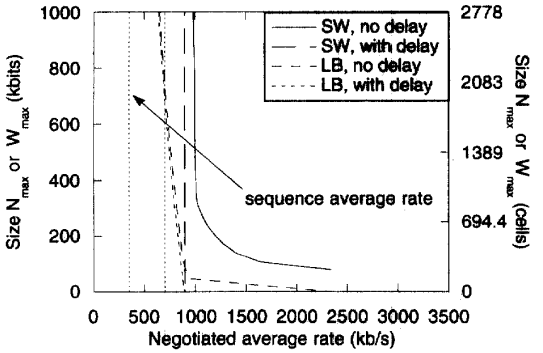

Fig. 7. TD parameters, sequence C.

average rate.) In these figures and throughout the paper unless otherwise specified, "with delay" refers to a video delay of $L = 3$ frames. In each figure, the left vertical dotted line indicates the actual average rate of the sequence. The right dotted line indicates twice the average rate. The vertical axis has been scaled to show clearly the knee of the curves, which occurs at approximately 500 cells for the sliding window and approximately 150 cells for the leaky bucket. For the reader's ease, we label the vertical axes in both kilobits and cells.

Four observations can be made from these figures. First, the general behavior is identical for each sequence, although there is some significant variation between the sequences.

Second, for both traffic descriptors, the size is generally quite large when the necessary negotiated average rate approaches twice the actual average rate, Because the scaling of the figures hides the upper size values, Tables II and III show the leaky bucket and sliding window sizes when the negotiated average rate, $\bar{R}$, is equal to and is twice the actual average rate of the sequence, respectively. When $\bar{R}$ equals the actual average rate, the TD size is huge—hundreds of thousands of cells for the sliding window. Even when $\bar{R}$ is as big as twice the actual average rate, the TD size can still be thousands of cells.

The third observation is that video delay can have a significant impact on the selection of the TD. As shown in Table I, the presence of video delay in the video system significantly reduces the peak rate. For each video sequence, the peak rate reduces by one half to one third when the codec delay

TABLE II
TD SIZE IN KILOBITS AND, IN PARENTHESES, IN NUMBER OF ATM CELLS ROUNDED TO THE NEAREST THOUSAND, WHEN THE NEGOTIATED AVERAGE RATE, $\bar{R}$, EQUALS THE ACTUAL AVERAGE RATE

Sequence	Leaky bucket without delay	Sliding window without delay	Leaky bucket with delay	Sliding window with delay
A	7,082 (20,000)	143,771 (400,000)	7,058 (20,000)	143,771 (400,000)
B	16,367 (45,000)	127,621 (355,000)	16,338 (45,000)	127,617 (355,000)
C	11,158 (31,000)	72,860 (202,000)	11,135 (31,000)	72,882 (202,000)

TABLE III
TD SIZE IN KILOBITS AND, IN PARENTHESES, IN NUMBER OF ATM CELLS, WHEN THE NEGOTIATED AVERAGE RATE, $\bar{R}$, IS TWICE THE ACTUAL AVERAGE RATE. (FOR SEQUENCE B WITH DELAY, THE ENTRIES ARE A DASH BECAUSE THE PEAK RATE IS LESS THAN TWICE THE ACTUAL MEAN RATE)

Sequence	Leaky bucket without delay	Sliding window without delay	Leaky bucket with delay	Sliding window with delay
A	1,217 (3,381)	9,568 (26,572)	1,169 (3,247)	9,520 (26,444)
B	59 (164)	1,924 (5,344)	-	-
C	751 (2,086)	6,784 (18,844)	693 (1,925)	6,719 (18,664)

goes from zero to three frames. For all these sequences when the delay is greater than one frame, the peak is determined not by the intraframe peaks, but by the bit-rate during high activity intervals. However, the negotiated average rate, $\bar{R}$, is not always as sensitive to the presence of delay as the peak rate is. For TD sizes that are below the "knee" of the curves in Figs. 5–7, the addition of delay has a major impact on $\bar{R}$, while for TD sizes above the "knee" the impact is minor, particularly for the leaky bucket.

The fourth observation is that for a given negotiated average rate, the leaky-bucket TD requires a bucket size that is significantly less than the window size required by the sliding-window TD. However, the implication of this numerical difference is not obvious, since the size parameters are not directly equivalent. Both TD's use an integrator, but the sliding window has a finite-memory integrator, while the leaky bucket could have an infinite-memory integrator if the bucket never empties. Therefore, to compare these TD's, we examine the "worst-case" ON-OFF source that would comply with these TD. We will use this ON-OFF source again in the next section for call admission.

For the leaky-bucket descriptor, the "worst-case" ON-OFF source is ON at the peak rate $R_{\max}$ until the bucket is full, and then OFF until it is empty. The ON and OFF periods are $N_{\max}/(R_{max} - \bar{R})$ and $N_{\max}/\bar{R}$, respectively (where we view the source as a continuous flow of bits), and the average rate is $\bar{R}$. For the sliding window, the "worst-case" ON-OFF source is ON at the peak rate $R_{\max}$ until the window is full, and then OFF until the average rate of $\bar{R} = W_{\max}/S_{win}$ is met. Thus, the ON and OFF periods are $W_{\max}/R_{\max}$ and $S_{win} - W_{\max}/R_{\max}$ respectively.

Table IV shows the ON period for a traffic burst that would be allowed by the TD when $\bar{R}$ is twice the true average rate and the TD size is chosen so that the video sequence is compliant (Table III). From Table IV we see that the ON period given the sliding-window TD is several times the ON period given the leaky-bucket TD. This observation holds for other values of $\bar{R}$ as well, where loosely speaking, we find that the ratio of the ON period for the sliding window to that for the leaky bucket is typically in the range of 2 to 10. Notice also in Table IV that the length of the ON period is measured in seconds.

TABLE IV
ON PERIOD OF TRAFFIC BURST ALLOWED BY TRAFFIC DESCRIPTOR, WHEN THE NEGOTIATED AVERAGE RATE, $\bar{R}$, IS TWICE THE ACTUAL AVERAGE RATE. (FOR SEQUENCE B WITH DELAY, THE ENTRIES ARE A DASH BECAUSE THE PEAK RATE IS LESS THAN TWICE THE ACTUAL MEAN RATE)

Sequence	Video delay (frames)	ON period for Sliding window (seconds)	ON period for Leaky bucket (seconds)	Ratio of ON periods of prior two columns
A	0	3.7	0.58	6.4
B	0	0.57	0.033	17.2
C	0	2.9	0.46	6.3
A	3	11.2	3.18	3.5
B	3	-	-	-
C	3	7.5	3.55	2.1

Given two calls with identical peak and average rates, a call admission policy that uses not only the peak and average rates, but also the longest possible burst duration [17], will be more likely to accept a call that has a shorter possible burst duration. Therefore, while these TD's describe the same video source, a network that uses this type of call admission policy will be more likely to admit the video call described by the leaky bucket than the call described by the sliding window. Therefore, the network may be utilized more efficiently if a leaky-bucket descriptor is used for a video call.

VI. CALL ADMISSION USING TD's

Given that the traffic descriptor for the video call consists of a peak rate and of a leaky-bucket or sliding-window algorithm, we compute a conservative estimate for the number of VBR video calls that can be admitted to a link. We compare this number to the number of CBR video calls that could be admitted and obtain an estimate for the statistical multiplexing gain from VBR video.

Given that the network operator is only told the peak rate and the parameters of a leaky bucket or sliding window algorithm, we suppose that for call admission the network operator makes the conservative assumption of periodic ON-OFF sources that stress the limits of the traffic descriptor. (The ON and OFF periods for both the leaky bucket and sliding window algorithms are given in the previous section.) Consider multiple calls that have identical traffic descriptors and assume each call has a random onset time which it maintains relative to the other calls. Let $p = \bar{R}/R_{\max}$ be the probability that a given source is ON at a particular time. Then the probability that m sources are ON when N calls have been admitted is the binomial probability $\binom{N}{m}p^m(1-p)^{N-m}$.

We assume the network buffers are small relative to the burst size. Therefore, if the instantaneous rate is greater than the network capacity, we assume cell loss will occur. We compute the cell loss ratio as

$$CLR = \frac{\text{Expected number of lost bits in an ON-OFF period}}{\text{Expected total number of bits in an ON-OFF period}} = \left[\sum_{m=n_0+1}^{N} [mR_{\max} - C]\binom{N}{m}p^m(1-p)^{N-m}\right] / (pNR_{\max})$$

where $n_0 = \lfloor C/R_{\max} \rfloor$ is the maximum number of calls that can be ON simultaneously without exceeding the link capacity C. This approximation for CLR is well known, see, e.g., [18].

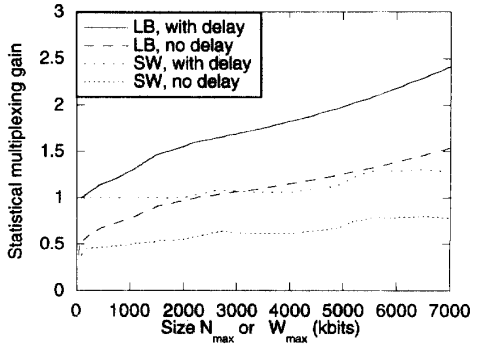

Fig. 8. Statistical multiplexing gain, sequence A.

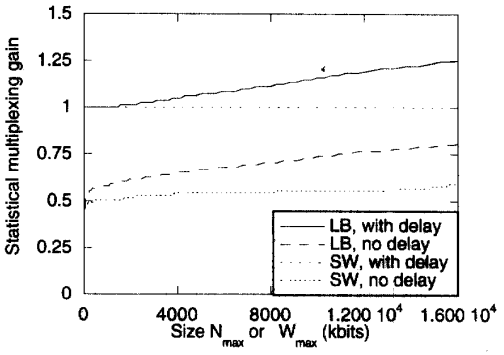

Fig. 9. Statistical multiplexing gain, sequence B.

A more accurate approximation that incorporates the size of the network buffer is presented by Rasmussen *et al.* [17]. We attempted to use their bounds, and although we could reproduce their results, the numerical computations became excessive for the parameter values of interest here. A still more detailed model that uses the distribution of the ON and OFF times is in [19]. For our present purposes, the simple approximation used herein is sufficient.

We compute the statistical multiplexing gain (SMG) to be the ratio of the number of VBR calls the link can accept with a $CLR \leq 10^{-9}, 10^{-6}$, or 10^{-3} to the number of CBR calls the link can accept without loss, n_0. It is important to realize that to compute the number of CBR calls the link can accept, we set the CBR rate to the VBR peak rate given a delay of three frames, as described in Section IV and Table I.

The statistical multiplexing gain when $CLR \leq 10^{-6}$ is shown in Figs. 8–10 for the three sequences A–C, as a function of the TD size. Our calculation for the number of calls that can be admitted does not directly use the TD size, but only the peak and negotiated average rates. However, to show the impact of the particular algorithms of leaky bucket and sliding window, we pick the independent variable to be the TD size ($N_{\max}$ or $W_{\max}$, respectively); the associated negotiated average rate that allows the traffic to be compliant for the particular descriptor can be found in Figs. 5–7.

The first observation about these figures is that, for the VBR sources without delay, the SMG can be less than 1, since the peak rate of the VBR without delay is significantly larger than

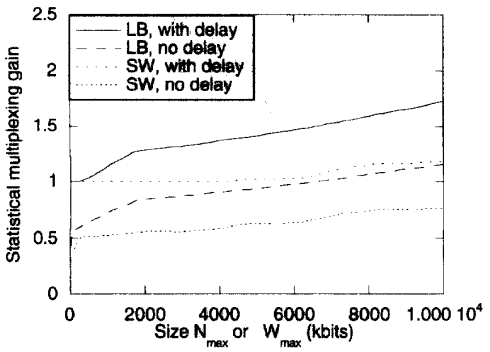

Fig. 10. Statistical multiplexing gain, sequence C.

the CBR rate, which is the peak rate of the VBR *with* delay. Without delay, the peak rate of the VBR source is large enough to offset any gains that could be obtained by its smaller mean. However, the SMG of the VBR sources with delay is always greater than or equal to 1. Therefore, using delays in the VBR video system is important for obtaining any SMG advantage over CBR.

Second, the statistical multiplexing gain increases as the TD size increases. The maximum statistical gain occurs when the negotiated average rate is equal to the actual average rate of the sequence. Therefore, for the network to carry the maximum number of calls, the TD size must be quite large. For example, for the leaky bucket algorithm, if the bucket size is 5 Mb (roughly 14 000 cells), then the SMG is respectively 2.19, 1.14, and 1.54 for the three sequences A–C. To obtain comparable SMGs with the sliding window algorithm requires a $W_{\max}$ that is over 46 Mbits.

Third, the statistical gains are fairly small. Table V shows the maximum statistical multiplexing gain for 8 sequences for different CLR, both without delay and delay $L = 3$. While some sequences have fairly large multiplexing gains, those scenes with more activity have only modest SMGs. However, the values shown in Figs. 8–10 and Table V are pessimistic because they use an ON-OFF source (not a video source) and a simple approximation for CLR. We expect the true SMG values to be higher, although they are upper bounded by the peak-to-mean ratios shown in Table I for $L = 3$. Furthermore, we expect the previous two observations to still be valid. That is, we expect the SMG to improve both as the video codec delay and the TD size increases.

Table VI illustrates how delay in the VBR codec affects the multiplexing gain for $CLR \leq 10^{-6}$. We compare the number of VBR calls with delay that would be admitted using the above call admission procedure to the number of VBR calls without delay. As the delay increases, the number of calls admitted increases. Increasing the delay to 2 frames significantly increases the SMG; small improvements can be expected by further increasing the delay.

VII. Conclusions

We have presented a comparison between VBR and CBR video. Delays are equated in both VBR and CBR systems, and the video qualities are made roughly equivalent. We presented the parameter values for a sliding-window and a leaky-bucket traffic descriptor that are necessary to ensure example video teleconferencing sequences are completely compliant, and we examined expected multiplexing gains.

We have four primary conclusions. First, the presence of delay in the video system can reduce the necessary peak rate, and can allow significantly more calls to be carried by the network, whether coded as VBR or CBR. A delay of $L = 2$ frames appears sufficient to obtain much of the gain; however, increasing the delay further will provide some additional improvement (though the subjective quality of interactive video may deteriorate if the delay is too large).

Second, for the sample teleconference sequences to be compliant with the leaky-bucket or sliding-window traffic descriptor, the size of the bucket or the size of the window may need to be large (on the order of thousands of cells), even when the negotiated average rate of the traffic descriptor is twice the true average rate. (Large bucket or window sizes have the disadvantage that a longer burst of cells at the peak rate could be submitted and still be compliant with the traffic descriptor.)

While we have chosen these parameters to ensure the sequences are compliant, in a real implementation, there is nothing to stop the user/network from agreeing to smaller

TABLE V
Maximum Statistical Multiplexing Gain

Seq.	Delay	10^{-3}	10^{-6}	10^{-9}
A	0	2.62	1.94	1.55
	3	3.11	2.69	2.43
B	0	1.35	1.00	0.81
	3	1.60	1.38	1.26
C	0	1.96	1.49	1.22
	3	2.28	1.99	1.81
D	0	1.86	1.35	1.06
	3	2.30	2.01	1.84
E	0	1.95	1.48	1.21
	3	2.29	2.01	1.83
F	0	3.30	2.33	1.79
	3	3.95	3.28	2.86
G	0	3.15	2.31	1.84
	3	3.27	3.18	2.84
H	0	3.28	2.20	1.62
	3	4.23	3.55	3.13

TABLE VI
Multiplexing Gains for CLR= 10^{-6} Using Nonzero Delay: Number of VBR Calls thatCan Be Carried Given Codec Delay, Divided by the Number of VBR Calls that Can Be Carried Given No Codec Delay

Sequence	$L = 1$	$L = 2$	$L = 3$	$L = 4$
A	1.22	1.35	1.38	1.39
B	1.23	1.37	1.38	1.40
C	1.20	1.32	1.33	1.34
D	1.24	1.39	1.49	1.52
E	1.20	1.33	1.36	1.37
F	1.27	1.36	1.41	1.45
G	1.23	1.36	1.37	1.38
H	1.32	1.50	1.61	1.70

parameters. However, this will imply that for the video system to obtain a compliant bit-stream, it will have to not only shape its source through the buffering, but also reduce its quality to decrease the overall bit-rate. In this case, the rate-control algorithm in the video system must choose the rate onto the network to conform to leaky bucket (or sliding window) constraints as well as encoder and decoder buffer constraints [10]. It is an open issue how much the bucket or window size could be reduced before the variation in quantizer step-size impairs the perceptible quality of the video. However, preliminary studies indicate that as the size of the leaky bucket decreases, it becomes less likely that the video system will be able to obtain any quality advantage compared to CBR video whose rate equals the leaky-bucket drain rate [10].

Third, comparing the two operational traffic descriptors, the leaky bucket is superior to the sliding window. For leaky-bucket and sliding-window parameters with a common negotiated average rate and chosen so that an example teleconference sequence is compliant, the worst case burst that could be admitted is several times smaller with the leaky bucket than with the sliding window.

Fourth, moderate multiplexing gains are only possible when a delay is present in the video system and when the size of TD is large. With a delay in the video system, the statistical multiplexing gain from VBR over CBR video is upper bounded by roughly a factor of four, and to obtain a gain of about 2.0 can require the operational traffic descriptor to have a window or bucket size of thousands of cells, given constant quantizer step-size.

While our results seem to indicate that VBR may not provide large SMG, we expect that potential for multiplexing still exists. However, further research will be necessary to extract the full potential.

In this paper, we selected the leaky bucket parameters given a complete video sequence apriori. In a real system, the leaky bucket parameters would have to be chosen before the complete sequence is available. If the parameters are chosen too small and there is no video buffering delay, then video quality may suffer when the bucket fills and the quantizer step-size oscillates as a result of the encoder rate-control algorithm. Two methods exist to allow less stringent adjustment of the quantizer step-size, and thus produce better video quality. In the first method, no delay is introduced into the video system, and the leaky bucket parameters would be chosen much larger, perhaps twice as large as given herein for the case of zero delay. Then, the video system could use the leaky bucket in the same way a CBR system currently uses an encoder buffer. The encoder rate-control algorithm would adjust the quantizer step-size to keep the leaky bucket from overflowing; however, to ensure the quantizer step-size does not oscillate, the leaky bucket parameters must be sufficiently large. In the second method, some delay is incorporated into the video system. This method also can ensure the quantizer step-size does not oscillate. While the first method may be needed for delay critical applications, the second method yields greater statistical multiplexing gains within the network, which in turn can lead to lower cost to the end-users for the video connection.

APPENDIX
VBR BUFFERING

We consider the system shown in Fig. 1. To model the buffer dynamics, we assume time is discretized at the frame level, although smaller intervals could be used if desired. Both the encoder and decoder buffer content must be constrained to prevent overflow and underflow. The encoder and decoder buffer sizes are $B^e_{\max}$ and $B^d_{\max}$.

In [10], Reibman and Haskell present bounds on the transmitted bit-rate such that neither the encoder nor decoder buffers overflow or underflow. We rewrite these here as bounds on the cumulative rate transmitted across the channel.

$$\sum_{j=1}^{i} E_j - B^e_{\max} \le \sum_{j=1}^{i} R_j \le \sum_{j=1}^{i} E_j \tag{3}$$

$$\sum_{j=1}^{i-L} E_j \le \sum_{j=1}^{i} R_j \le \sum_{j=1}^{i-L} E_j + B^d_{\max}, \tag{4}$$

when $i > L$, and L is the delay as defined in Section II–B. The left side of relation (3) prevents encoder buffer overflow, while the right side prevents encoder buffer underflow. Similarly, the left side of relation (4) prevents decoder buffer underflow, and the right side prevents decoder buffer overflow. Upper and lower bounds on the cumulative transmitted rate can be found by taking the maximum of the left sides, and the minimum of the right sides.

$$D_i \le \sum_{j=1}^{i} R_j \le U_i, \tag{5}$$

where

$$D_i = \max\{\sum_{j=1}^{i} E_j - B^e_{\max}, \sum_{j=1}^{i-L} E_j\} \tag{6}$$

and

$$U_i = \min\{\sum_{j=1}^{i} E_j, \sum_{j=1}^{i-L} E_j + B^d_{\max}\}. \tag{7}$$

Any sequence of rates $\{R_i\}$ that satisfies relation (5) is valid from a buffering standpoint. Choosing one R_i may affect the range of possible choices for other R_j.

For a given video sequence, delay, and buffer sizes, the easiest way to see the effect of these bounds visually is to plot $U_i - i\tilde{R}$, and $D_i - i\tilde{R}$ as a function of the time index i, where $\tilde{R}$ is some convenient choice of rate. An example is shown in Fig. 11 for the first 200 frames of sequence A, where $\tilde{R}$ is chosen to be the average rate of the 200 frame subsequence. The bounds are shown as dashed lines. The delay is $L = 3$ frames, and the buffer sizes are large enough that the constraints on the rate are imposed solely by the delay. The actual rate is indicated by the slope of the path: a positive slope corresponds to a rate greater than $\tilde{R}$, while a negative slope corresponds to a rate less than $\tilde{R}$. A zero rate corresponds to a slope of $-\tilde{R}$.

We note that, for this sequence, it is impossible to transmit at a constant rate without overflowing or underflowing one

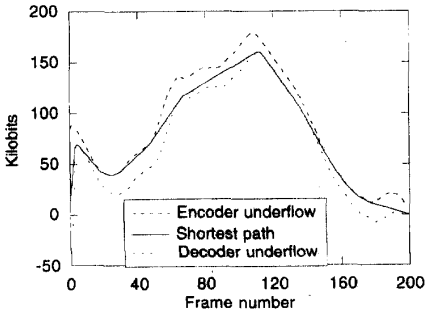

Fig. 11. Bounds on cumulative rate imposed by buffering.

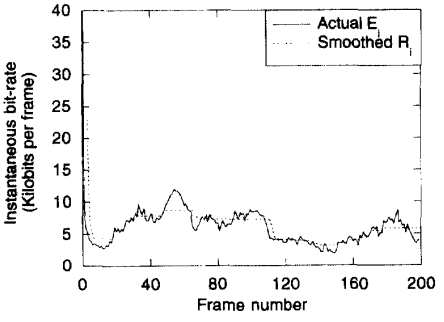

Fig. 12. Actual and smoothed rate for sequence A.

of the buffers, since no straight line passes through these bounds. However, paths through these bounds that correspond to variable rate transmission do exist. An example is shown by the solid line.

To generate a plausible constrained VBR bit-stream, we make use of the full extent of the allowable delay, by choosing the path through the bounds that has the shortest length. This provides a solution that is not causal, so it can not be implemented. However, it does use the delay to its maximal extent. Furthermore, this path will minimize the maximum rate since it will have the smallest possible slope.

The instantaneous rate associated with the shortest path is independent of the choice of $\tilde{R}$. The shortest path can be found by applying the algorithm described in [20].

The shortest path does not necessarily minimize the average rate for a given sliding window size, especially for large windows. However, for sequences A–C, the shortest path does minimize the average rate for windows up to 30 frames long. In addition, this one path minimizes the leaky bucket size for all drain rates. Visually, the bit-rate resulting from the shortest path is quite smooth, as shown in Fig. 12. The actual coded bit-rate is quite noisy, while the smoothed rate is constant for large time intervals.

As we said, the shortest path determines noncausally a path through these bounds. However, once a leaky bucket (or sliding window) size has been found that allows this non-causal choice of R_i to pass without constraint, a causal rate-control algorithm can be used. The causal rate-control algorithm can select its rate using the fullnesses of the leaky bucket (or sliding window), the encoder buffer, and decoder buffer [10]. Therefore, the non-causal shortest path rate is useful in determining traffic descriptor parameters, since a causal algorithm can then be used to determine the actual rate in a real system.

ACKNOWLEDGMENT

The authors would like to thank D. Lubinsky for his help with the shortest path algorithm, and A. Eckberg and B. Haskell for their thoughtful reviews of earlier drafts. The authors would also like to thank the anonymous reviewers for their conscientious and thorough comments which greatly improved the paper.

REFERENCES

[1] A. R. Reibman and A. W. Berger, "On VBR video teleconferencing over ATM networks," in *Proc. IEEE GLOBECOM'92*, Dec. 1992, pp. 314–319.
[2] A. W. Berger and A. E. Eckberg, "A B-ISDN/ATM traffic descriptor, and its use in traffic and congestion controls," in *Proc. GLOBECOM'91*, Dec. 1991, pp. 266–270.
[3] A. W. Berger, A. E. Eckberg, T. C. Hou, and D. M. Lucantoni, "Performance characterizations of traffic monitoring, and associated control, mechanisms for broadband 'packet' networks," in *Proc. GLOBECOM'90*, Dec. 1990, pp. 350–353.
[4] E. P. Rathgeb, "Modeling and performance comparison of policing mechanisms for ATM networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 325–334, Apr. 1991.
[5] ITU-T Recommendation I.371, "Traffic control and congestion control in B-ISDN," Mar. 1993.
[6] G. Ramamurthy and B. Sengupta, "Modeling and analysis of a variable bit rate video multiplexer," in *7th Int. Teletraffic Congr. Sem.*, Oct. 1990, p. 8.4.
[7] R. Grunenfelder, J. P. Cosmas, S. Manthorpe, and A. Odinma-Okafor, "Characterization of video codecs as autoregressive moving average processes and related queueing system performance," *IEEE J. Selected Areas Commun.*, vol. 9, pp. 284–293, Apr. 1991.
[8] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. D. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Trans. Commun.*, vol. 36, pp. 834–843, July 1988.
[9] D. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 2, pp. 49–59, Mar. 1992.
[10] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 2, pp. 361–372, Dec. 1992.
[11] B. Voeten, F. Van der Putten, and M. Lamote, "Preventive policing in video codecs for ATM networks," in *Fourth Int. Workshop on Packet Video*, 1991, pp. G1.1–G1.6.
[12] Recommendations of the H-series, Tech. Rep., CCITT, Aug. 1990.
[13] Description of reference model 8 (RM8), Document 525, CCITT SGXV Working Party XV/4, 1989.
[14] International Standards Organization, International Standard 11172-2, "Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbits/s," May 1993.
[15] A. E. Eckberg, B. T. Doshi, and R. Zoccolillo, "Controlling congestion in B-ISDN/ATM: Issues and strategies," *IEEE Commun. Mag.*, vol. 29, pp. 64–70, Sept. 1991.
[16] W. S. Tan, N. Duong, and J. Princen, "A comparison study of variable bit rate versus fixed bit rate video transmission," in *Australian Broadband Switching and Services Symp.*, 1991, pp. 134–141.
[17] C. Rasmussen, J. H. Sorensen, K. S. Kvols, and S. B. Jacobsen, "Source-independent call acceptance procedures in ATM networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 351–358, Apr. 1991.
[18] J. W. Roberts, Ed., Performance evaluation and design of multiservice networks: Final Rep. of COST 224. Tech. Rep., Commission of the European Communities, 1992.
[19] B. Bensaou, J. Guibert, and J. W. Roberts, "Fluid queueing models for a superposition of on/off sources," in *7th Int. Teletraffic Congr. Seminar*, Oct. 1990, pp. 9.3.
[20] D. T. Lee and F. P. Preparata, "Euclidean shortest paths in the presence of rectilinear barriers," *Networks*, vol. 14, pp. 393–410, 1984.

Amy R. Reibman (S'83-M'87) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Duke University in 1983, 1984, and 1987, respectively.

From 1988 to 1991, she was an assistant professor in the Department of Electrical Engineering at Princeton University. She is currently a Member of the Technical Staff in the Visual Communcations Research Department at AT&T Bell Laboratories. Her research interests include video compression and packet video.

Dr. Reibman is a member of Sigma Xi, Eta Kappa Nu, Tau Beta Pi. She was the Technical Program Chair for the Sixth International Workshop on Packet Video, Portland, OR, September 1994, and she is currently an Associate Editor for IEEE TRANSACTIONS ON IMAGE PROCESSING. Her email address is amy@research.att.com.

Arthur W. Berger (M'83) received the Ph.D. degree in applied mathematics from Harvard University in 1983.

Since 1983 he has been a member of technical staff at AT&T Bell Laboratories, Holmdel, NJ, where he has worked on network planning, congestion controls in telecommunication switching systems, and most recently on traffic controls and traffic engineering for Broadband ISDN/ATM.