VBR video: Trade-offs and potentials

T. V. Lakshman
Bell Labs Lucent Technologies
Holmdel, NJ
lakshman@research.bell-labs.com

Antonio Ortega
Integrated Media Systems Center
Dept. of Electrical Eng.-Systems
University of Southern California
Los Angeles, CA 90089-2564
ortega@sipi.usc.edu

Amy R. Reibman
AT&T Laboratories
Red Bank, NJ 07701
amy@research.att.com

Abstract

In this paper, we examine the transport and storage of video compressed with a variable bit rate (VBR). We focus primarily on networked video, although we also briefly consider other applications of VBR video, including satellite transmission (channel sharing), playback of stored video, and wireless transport. Packet video research requires careful integration between the network and the video systems; however, a major stumbling block has resulted because commonly used terms are often interpreted differently by the video and networking communities.

This paper then has two main goals: (i) to clarify the definitions of terms that are often used with different meaning by networking and video coding researchers, and (ii) to explore the trade-offs entailed by each of the various modalities of VBR transmission (unconstrained, shaped, constrained, and feedback). In particular, we evaluate the trade-off among the advantages (better video quality, less delay, and more calls) which were identified by early proponents of VBR video transmission. An underlying theme of the paper is that increased interaction between the video and network design has potential for improving overall decoded video quality without changing the network capacity.

Contents

1 Introduction

Variable Bit Rate (VBR) transmission of video and other media encoded at variable rate has been for some time a desirable objective, as it seems natural to match the variable rate output of a coder to a variable transmission rate over the channel. For example, video transmission over packet networks has been a topic of high interest for at least the last ten years. The importance and potential of variable rate video transmission started a series of focused research efforts in the mid to late eighties as attested by events such as the First International Packet Video Workshop, which took place at Columbia University in May 1987, and the appearance of papers that have since been widely referenced in the packet video literature [1, 2, 3]. From the very beginning it was recognized that successful deployment of packet video would require advances in both compression and networking technologies, along with a great deal of interaction and collaboration between the two communities. Thus, the Packet Video Workshop aimed at attracting researchers from both communities to foster the interaction. Nowadays, while VBR video services in various forms are already available (e.g. Video over the Internet or channel sharing for satellite transmission of video), no systems have been defined that can claim to provide all of the advantages (better video quality, less delay, more calls) which were identified by the early proponents of variable bit rate transmission of video over packet networks. We will concentrate our discussion on packet video because historically this has been the area of major interest for VBR transmission. We discuss briefly other applications for VBR video in our conclusions.

The purpose of this paper is not to cover exhaustively the main accomplishments of the last ten or so years but to provide a tutorial overview and draw some conclusions about what these results imply for the establishment of future VBR video transmission systems. Our emphasis will be in describing all the alternative forms of VBR video coding and studying how they interact with the underlying VBR networking transmission infrastructure. In particular we will concentrate our discussion in two main areas, namely, (i) to clarify definitions of terms that are often used with different meaning by networking and video coding researchers, and (ii) to explain the trade-offs that each of the various modalities of VBR transmission entail.

Widely used terms in the packet video research are often seen from different perspectives in the video and networking communities. One example of this difference in perspective is the notion of

Quality of Service (QoS). In the networking community, QoS is measured in the network-centric terms of delay jitter, packet losses and bounds on delay. As a result, the design goal from the network's perspective has been to meet negotiated QoS guarantees while maximizing the number of calls and the revenue. In contrast, the designer of a video system is concerned with maximizing decoded video quality (as measured either by subjective or objective measures such as Peak SNR), which is clearly affected by the negotiated network transmission parameters, and also by appropriate choices of encoder parameters like image resolution and frame rate. In this paper, we use QoS to describe the effect on video quality as a result of both network and video impairments.

The definition of Variable Bit Rate (VBR) video itself is often a cause of confusion. Any bit rate trace obtained from a video encoder is normally considered to be VBR. Studies of network performance have usually relied on traces generated by video encoders operating "open loop", that is, without any kind of feedback. In this paper, we argue that feedback to the video encoder is in fact one of the key characteristics of video (as opposed to data) transmission over packet networks. In data transmission feedback can only be used to adapt the way the information is sent (e.g. reducing the transmission rate if congestion occurs as in TCP/IP) but the information itself cannot be changed. Instead, video encoders can modulate the data they produce by adjusting a number of parameters, including quality, frame rate and resolution. Our discussion will thus be organized by classifying the various types of VBR video transmission in terms of the type of feedback present and the degree of knowledge of the system available at the video encoder.

We consider a generic system, shown in Fig. 1, where the video encoder emits encoded data to a buffer. This encoded data, or video traffic, is then drained at a variable rate which is monitored at the User Network Interface (UNI) [4]. The UNI monitors the transmitted rate and compares the connection parameters to those negotiated between user and network at the time of connection set-up. The network policing functions, if used, can be considered to be implemented at the UNI. Finally, traffic transmitted through the network experiences some QoS. Information about the currently available QoS, as determined for example by the existence of congestion, might be also transmitted back to the UNI and thence to the source (as for example in an Available Bit Rate, ABR, scheme [5]).

With this generic system diagram we define the following modes of operation:

1. Unconstrained VBR (U-VBR), where the video encoder operates independently of the UNI. For example the encoder operates with a constant quantization scale throughout transmission. Most video rate modeling efforts have been based on U-VBR traces.

2. Shaped VBR (S-VBR), where the buffer is linked to the UNI, but is not connected to the encoder. In this case, the encoder produces a bitstream that is identical to that in U-VBR. However, now a shaping algorithm can determine the actual cell transmission patterns. While the content of the bitstream is unaffected, the traffic patterns may be smoothed out at the cost of some additional delay.

3. Constrained VBR (C-VBR)*, where the encoder has knowledge of not only the buffer state but also the networking constraints at the UNI. Thus the video encoder can modulate its output so as to maximize the video quality given all the applicable constraints, including those related to delay and transmitted rate. Here, the bitstream content *is* affected, but the changes are made by the video encoder, which can change the rate in a manner that has the least impact on perceptual quality.

4. Feedback VBR (F-VBR), which adds information about the network state to what is made available to the encoder. This allows the same trade-offs as in C-VBR to be considered with the additional advantage that the encoder can adjust to changes in the state of the network (for example congestion periods).

Each section will provide a clear definition of each class of VBR transmission, will state advantages and disadvantages of each, and discuss the state of the knowledge on each (e.g., guaranteeing QoS for U-VBR, smoothing techniques for S-VBR, or rate control techniques for C-VBR). Looking at these classes separately will help us identify the trade-offs between the various resources.

The promised advantages of VBR video are (i) *better video quality* for the same average bit-rate, by avoiding the need to adjust the quantization as in CBR, (ii) *shorter delay*, since the encoder buffer size can be reduced without encountering an equivalent delay in the network, (iii) *increased call-carrying capacity* because the bandwidth per call for VBR video may be lower than for CBR

*This was referred to as Shaped Bit-Rate, or SBR by Hamdi *et al.* [6, 7], however, we find that this terminology can sometimes cause confusion among the networking community. We have therefore decided upon the current naming convention.

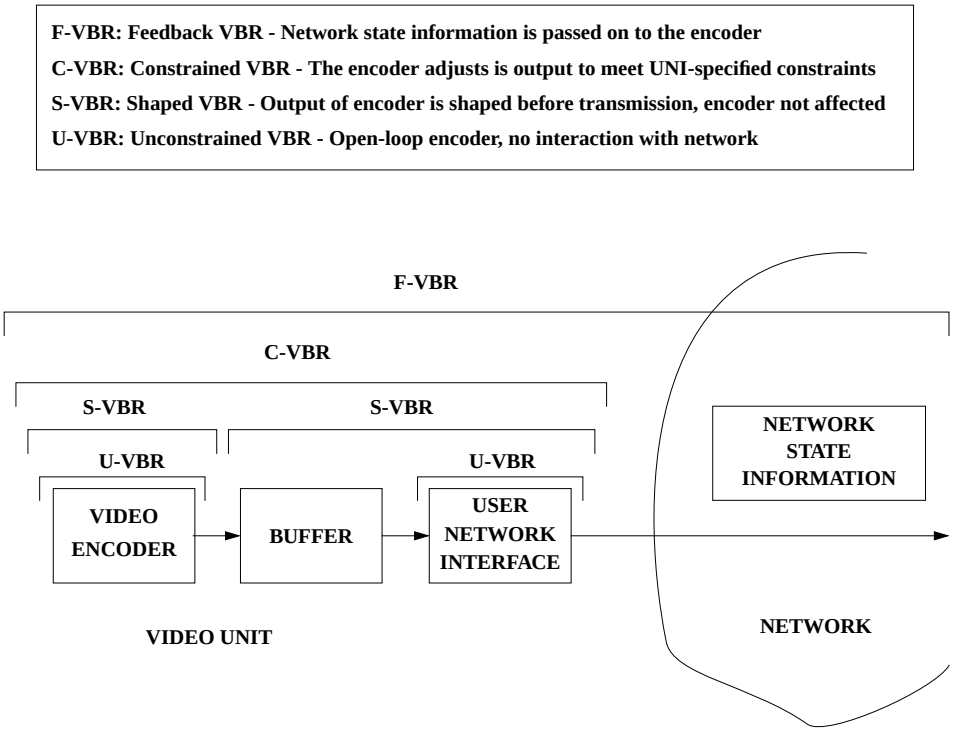


Figure 1: Different types of VBR transmission considered in this paper. U-VBR has independently operating encoder and UNI. S-VBR links the buffer with the UNI so that shaping can be performed but it does not affect the encoder. Under C-VBR, the encoder takes into account both buffer state and constraints imposed by the UNI to encode the input data. Finally, under F-VBR, the encoder has all the information in the C-VBR mode and in addition has access to feedback on the current state of the network.

for equivalent quality. While these potential advantages were heavily emphasized in early packet video papers, further research has shown that no design can maximize them simultaneously. We refer to the advantages here as metrics, because they can be used to measure the degree to which a specific design achieves its goals. A central theme of this paper is to explore the trade-offs among these advantages for the different VBR types. Therefore, in Section 3 we give a detailed explanation of these metrics.

Clearly, a Constant Bit Rate (CBR) transmission mode, because of its predictable traffic patterns, makes the task of network management easier, although it also precludes gain via statistical multiplexing. Further, the CBR mode is not well matched to the inherently VBR characteristics of coded video. However, it is difficult to support good quality video using VBR transport without the network providing feedback to the source or providing end-to-end delay and loss guarantees[†].

[†]Transmission of video over channels without such feedback is possible, as demonstrated by live video multicasts

Therefore, we will consider U-, S-, and C-VBR with transport that provides bounds on delay and loss (e.g., RT-VBR for ATM [8]) and F-VBR with transport that provides explicit-rate feedback (e.g, ABR for ATM).

This paper is outlined as follows. In Section 2, we describe the notion of quality of service, both from the network and the video viewpoint. Section 3 defines the three potential advantages of variable bit-rate transmission of video. Section 4 considers the models available for characterizing video sources. Sections 5, 6, and 7, consider U-VBR, S-VBR, and C-VBR, respectively. Section 8 presents the trade-offs between these three modes through an example based on our coding experiments. Sections 9 discusses F-VBR and Section 10 concludes with some observations about future research and describes some other applications that can benefit from variable bit-rate video.

2 Quality of Service

Quality of Service (QoS) is a popular term in the networking literature that is commonly used to characterize many types of "services", including for example non-real-time transmission applications such as file transfer. Typical metrics used for QoS may include the available bandwidth, the absolute delay, the cell delay variation (CDV) and the cell loss rates. Clearly, it would be desirable to have ways of translating these lower level networking parameters into the quality measures that can be more readily appreciated by the end users.

For file transfers or WWW applications this translation is in fact relatively straightforward, since the end user will mostly be interested in the delay required to complete the transfer. For example, given a maximum bandwidth, round-trip times, and the cell loss rates of a certain link, one can determine typical overall transmission times for an ftp transfer. It would then be easy to predict the user's "assessment" of the quality of the end-to-end service.

Quality assessment is far more complicated when it comes to real-time transmission of video. Given the same objective of measuring the quality observed by the user, it becomes clear that the end quality is measured not just by delay (for example the round-trip delay in a videoconferencing application) but also by the decoded video quality. The encoded video quality is completely determined by the video encoder and its parameter selection (quantizer, frame resolution, and frame

over the Internet. However, the resulting video quality cannot be as high as in a guaranteed transmission environment, since the video encoding algorithm has to be made robust to almost certain channel losses (and thus both video quality and compression factors suffer).

rate), while the decoded video quality depends not only on the same encoder parameter selection, but also on the networking QoS parameters (network delay, cell loss, and cell delay variation (CDV)).

2.1 Networking QoS parameters and Video Quality

In this paper, we assume that the available bandwidth, along with network delay, cell loss and cell delay variation, can be negotiated between user and network at the start of connection. We also assume that with appropriate admission control and policing these parameters can be guaranteed. For a given bandwidth we now discuss how the networking QoS parameters affect the video quality.

Network-based delay is treated very briefly in Section 3.2. The perceived quality of interactive video applications can degrade dramatically when the round-trip delays increase, although delay does not noticeably affect the quality of applications with one-way transmission.

The impact of cell losses on VBR video quality is not considered here, primarily because techniques for cell loss resilience are identical for both VBR and CBR video. The interested reader can find details on error concealment algorithms for video in [9, 10], for example. Robustness to transmission losses can also be enhanced by using error correction [11] and by matching the degree of protection to the perceptual importance of the transmitted information [12].

The remaining network-based QoS parameter, jitter or cell delay variation (CDV) typically causes more problems for VBR video than CBR video. Therefore, we discuss it briefly here, although space constraints prevent us from giving a detailed treatment. Because different cells of a stream "see" different queue lengths in the network buffers, each cell experiences a different delay. Hence, even if cells are submitted equally spaced as in a CBR stream, they will arrive at the receiver with variable spacing. In multi-hop networks with best-effort service (e.g., the current Internet), these variable delays can sometimes be on the order of seconds. In networks with guaranteed QoS, when there are many hops, the jitter has to be controlled by scheduling and traffic engineering to constrain it to application requirements.

For both CBR and VBR video streams, jitter can cause problems for two reasons. First, if the CDV is large, there might be instances when the data arrives at the video decoder too late to be used. While these cells are not lost, they might as well be. Therefore, the system must be designed with sufficient additional end-to-end delay (and hence sufficient additional buffering) to ensure that

cells incurring the maximum delay will still arrive before they need to be decoded. Because the maximum jitter delay is typically much smaller than the overall end-to-end delay, this does not usually impose a significant problem. In this paper, we assume there is sufficient buffering in the video decoder to accommodate this additional requirement.

However, the second problem, the inaccuracies that jitter introduces into the timing recovery algorithms, can be more significant particularly for higher quality video. Timing recovery in general is a difficult problem, and we do not discuss it in detail here. The designers of many of today's Internet-based packet video systems avoid the problem of accurate timing recovery by running the decoder clock at a marginally faster rate than the encoder clock, thus ensuring decoder buffer overflow will never occur. The decoder buffer may underflow, but underflows can be handled by frame-repeats, an impairment that is not considered too obtrusive given the fact that quality requirements are typically not very stringent in these applications. In general, however, much more sophisticated timing recovery algorithms are required to accommodate jitter in systems requiring high quality video.

2.2 Selection of Networking QoS parameters for VBR video

The distinction we draw between file transfers and video transmission is significant and illustrates the fact that the overall end-to-end quality of service depends on the choices made when selecting *both* video encoding parameters and networking QoS parameters. This is also true of data transmission in general but the selected parameters might be different for real-time video transmission.

This can be seen by considering an example where the video application has to choose between (i) a high throughput, high loss transmission or (ii) a low throughput, low loss transmission. For file transfers, unless round-trip times are very large or a loss-sensitive congestion control like TCP's is used, a higher throughput high loss connection may result in shorter transfer time than a low throughput low loss connection. (Even if some cells need to be retransmitted, the transfer is completed whenever all the cells have been received and reordered.) However, for video the choice may be less clear since the second choice (i.e. lowering the quality at the encoder but suffering few channel losses) usually results in better *decoded quality* than allowing losses to occur during transmission, even if the encoded quality was superior in case (i). In addition, for a real-time service it may not be possible to retransmit information, so any loss in the network results in lost

information at the decoder, rather than increased delay as in the case of file transfer.

As another example, consider video transmission over the Internet. While the service provided, as exemplified by multicast over the Mbone, can hardly be described as being "high quality" video, it nonetheless can be built into very usable applications. This is in contrast with typical assumptions about video that indicate it can only be successfully transmitted over broadly reliable channels. The success of Internet video applications shows how it is necessary to make an appropriate choice of a video compression algorithm that is well matched to the particular channel characteristics. As this example illustrates, incorporating knowledge of the channel characteristics into the video encoder allows the encoder to make appropriate selections regarding frame rate, resolution, and target PSNR. Clearly, incorporating as much knowledge about the network as possible into the overall system design is advantageous, and equally clearly, it is the video encoder that can most flexibly adapt to this knowledge. This premise motivates our discussion throughout this paper.

3 Quality, Delay, and Multiplexing Gain

In this section, we define each of the potential advantages available by transporting video using a variable rate rather than a constant rate: better video quality, shorter delay, and lower bandwidth allocation. A precise definition of these advantages is vital to understand the rest of the paper and, in particular, the experiments of Section 8. Moreover, we discuss a framework, based on all three parameters, to set up a fair comparison between CBR and VBR transport of video.

3.1 Video quality

In general, measuring video quality is a complex undertaking, considering all the relevant factors including the human spatiotemporal contrast sensitivity, spatial and temporal masking, and the impact of time-varying quality. Because this is such a difficult problem, we will, as many other papers do, resort to using the Peak Signal-to-Noise Ratio (PSNR) as an admittedly flawed measure of video quality. To obtain more qualitative measures of quality we use the following rough rules of thumb. Finer quantizers provide better quality than coarser quantizers. Quantizing using varying coarseness in different frames typically provides worse quality than quantizing with the same quantizer in all frames, and in general, less variability between frames tends to be better[‡]. It is also

[‡]We are concerned here more with inter-frame quantization differences than with intra-frame quantization differences. We recognize the important role that varying the quantizer coarseness based on the local spatiotemporal

worth noting that while preventing variability between consecutive frames is important, the same is not necessarily true between different scenes within a long sequence, because maintaining a "consistent" level of perceptual quality in an efficient manner may require using different quantization levels for different scenes.

Maintaining the decoded quality to be as consistent as possible throughout the connection should be a primary goal of the system designer. Note that in the presence of such factors as channel losses, the best choice may not necessarily be that which maximizes the throughput or maintains a constant quantizer throughout the transmission.

3.2 Delay

Delay is introduced in a number of components in the overall video transmission system, including (a) video encoder preprocessing and decoder post-processing, (b) actual encoding and decoding time, (c) frame reordering at both ends for B-frames in MPEG video, (d) encoder and decoder buffering (not including buffers for network adaptation), (e) interface delays for network adaptation between network and codec (including packetization delays), and (f) network transport delays.

In this paper, we use the total end-to-end delay as a performance metric. Obviously, lower delay is preferable since it is required for interactive applications and allows reduced latency times in one-way communication systems. We are primarily concerned with the delays (d) introduced by the encoder and decoder buffering, excluding the network adaptation buffering. These delays are of particular interest because they can be much larger than the other delays, and because the amount of buffering in the video system can strongly influence both the video quality and the multiplexing gains through the associated traffic characteristics. In particular, the buffering delays introduced by the network while multiplexing VBR traffic can be designed to be on the order of 10-20 msec so that the maximum jitter is constrained to be of this magnitude. This delay is typically much smaller than the equivalent buffering delays that would be required in the video system to smooth VBR traffic into CBR traffic. These latter are usually of the order of several frame times (i.e. multiples of 33 msec).

While more detailed analyses of delay issues can be found in [13, 14, 15], we summarize here some of the basic conclusions. A common misconception is to assume that delay in a VBR video

activity plays in improving video quality for a given channel rate constraint.

system can be variable. In fact, unless the system supports a variable frame rate[§], the delay in the system is *constant* for the duration of the connection. The delay between the time when frame i is captured and the time when frame i is displayed must remain constant because both capture and display devices operate on a constant frame rate. Even if the video source has been compressed and stored, the decoder is constrained in that once decoding commences, a frame must then be displayed every $T = 1/30$ seconds. At the start of transmission the delay is introduced in the system by having the decoder wait a certain time after it receives the first bit of the video stream before starting to decode it. Because the number of bits per frame is variable and the channel rate need not be exactly matched to this rate, this delay may be necessary 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.

Thus, with constant frame/rate T frames/second at encoder and decoder, the time elapsing from frame capture at the source to frame display at the destination will be constant for every frame, and equal to $\Delta N \cdot T$ seconds. ΔN represents the number of frames in the system at any given time, and it must be known *a priori* at both the encoder and the decoder. Note that the ΔN frames in the system may need to be stored in the encoder or decoder buffers, although the network might be transmitting some fraction of those bits at any given time. Thus, larger values of ΔN require larger buffers at the encoder. Larger buffers can accommodate larger rate fluctuations from frame to frame, which in turn enables higher quality. as will be seen in our examples.

It is important to note that buffering and delay constraints cannot be ignored for stored (i.e., not real-time) video any more than for interactive video, since these constraints are relevant for so-called workahead smoothing [14, 16]. For this application, the actual amount of buffering delay may not be determined so much by the real-time delay constraints but instead by the amount of physical buffering available in the set-top decoder, which is oftentimes restrictive. However, delay itself plays a vital role because of the necessity of the video decoder to emit a continuous stream of frames, one every 33 milliseconds. Once display has started with a given delay, that delay must be maintained.

[§]Variable frame rate can be caused by either the encoder or the decoder dropping frames. If the encoder drops frames the effect on perceptual quality can be relatively small. However if the decoder drops frames (e.g. because they are received too late to be decoded) significant quality degradation occurs, especially when motion-compensated coding methods are used.

3.3 Multiplexing gain

We now discuss how one might measure the multiplexing gain of VBR video, i.e., the ability for the network to carry more VBR video streams than CBR video streams. Ideally, one would like to measure it in terms of the multiplexing gain compared to CBR video[¶]. However, this is problematic, because to accurately characterize this value, it is essential to remove the influence of the other two parameters, namely the delay and the quality. It is straightforward to equate the delay between a VBR stream and a CBR stream, but equating the quality between the two is difficult indeed.

To determine the actual multiplexing gain compared to CBR video, one must first find the CBR rate that would produce equivalent quality to the VBR stream. However, systematic methods to determine the appropriate CBR rate have not yet been found. Clearly, the CBR rate must be between the peak rate of the unconstrained VBR stream and the mean rate of the VBR stream. Both have been used in the literature, but the first wildly overestimates the CBR rate and hence the multiplexing gains, while the latter significantly underestimates them. A somewhat more accurate estimation of the equivalent CBR rate can be found by assuming it is equal to the peak rate of the *smoothed* or shaped source, where smoothing is done using the same amount of buffering delay as the CBR source would have [14]. In this manner, delay between the VBR and CBR sources are equivalent. However, subjective tests indicated that this technique may still overestimate the required CBR rate [14].

Extensive subjective tests were used in [17] to determine the CBR rate that gave perceptually equivalent quality to a C-VBR stream. Subjective testing using 10 subjects was performed on a 12-second teleconferencing sequence using AB comparison. The results indicated that a CBR 256 kbps sequence had equivalent quality to a U-VBR sequence with mean rate 139 kbps and a peak to mean ratio of 4. Using the network model, this indicated a SMG for the VBR video of 1.49 at a CLR of 10^{-6} [17]. This method does determine the correct equivalent CBR rate. However, subjective tests are generally exhaustive, and repeated tests with different viewers may produce a fairly large standard deviation in the CBR rate. Moreover, the results can only be valid for the specific video sources used in the test, since it is not clear these results can be extended even

[¶]In this paper, we discuss network utilization assuming that only video is being multiplexed. However, in general, data can be assumed to be carried as best-effort traffic in the unused capacity. Naturally, the revenue the networking company receives depends on the relative prices between the different services. However, exploring this relationship is beyond the scope of this paper.

to similar sequences. Therefore, until an accurate perceptual quality metric is created that can accurately and objectively characterize the overall quality of a sequence whose visual quality is time varying, the actual CBR rate that provides equivalent video quality to that of a VBR stream with nearly constant quality will be unknown.

Because there is yet no known way to determine computationally the appropriate CBR rate, we will merely use instead a metric of the number of calls that can be accepted without exceeding some specified cell loss rate. Methods to estimate the number of calls that can be carried typically rely on statistical models of the video sources, which we consider next.

4 Statistical Modeling of VBR Video

We now discuss some of the issues arising in modeling of VBR video sources. Note that most of the studies published so far, including those described in this section, are related to modeling of U-VBR video. However, we present statistical modeling separately from any particular VBR mode because it is used for performance analysis and admission control for each VBR mode. Any system purporting to provide QoS guarantees for transported data does so based on some *a priori* model of the traffic to be supported. Clearly, traffic which exceeds the parameters of the model will receive reduced QoS, because the network either explicitly polices based on predetermined traffic descriptors, or implicitly gives service only when the source complies with some parameters, e.g., when per-connection buffering and fair queueing is used.

4.1 Source modeling

The typical approach to source modeling has been empirical in the sense that many data sets of traffic, such as traces of the number of bits per frame, are analyzed and used to formulate source models that are valid for a class of sources. There has been little success in formulating source models that use the coding algorithm itself as the basis.

Since no model of a data set is perfect, the quality of source models is judged by the performance of the model in some "real" networking application. This is usually taken to be the model's accuracy in predicting mean cell loss rates in a single bottleneck link with many multiplexed video sources and is tested by simulation using the traffic traces. Data analysis and formal goodness-of-fit tests are intermediate quality assessments that can be used to improve the chances that the model will

indeed work for the required purpose.

For source modeling purposes, video sources can be roughly classified into those which are moderately active (video teleconferences are presumed to be representative of this category), very active (movies), and those with a deterministic structure (MPEG-coded sources with a fixed group-of-pictures). Models differ widely amongst these categories, and it is not even clear that models that describe a large class can be found, although as will be discussed later, this might be easier for S-VBR or C-VBR than for U-VBR. Below we briefly describe, for illustrative purposes, video source models that are representative of those developed for different types of video sequences. The different models that have been proposed are too numerous for us to attempt an exhaustive survey. A few examples are [68, 69, 54, 70, 71, 72].

4.1.1 Models for Video Teleconferences

In [58, 59], traffic models for video teleconferences were formulated by examining data recorded during several 30 minute video teleconferences. A key point is that traffic models look similar despite the sequences differing in the details of the coding scheme [58]. The important features of the video teleconference models can be summarized as follows. The number of cells per frame can be modeled by a stationary process. The marginal distribution of the number of cells per frame follows a gamma distribution (negative binomial if a discrete distribution is used) and so the number of cells per frame is given by

$$X(t) = \frac{\lambda(\lambda t)^{s-1}}{\Gamma(s)} e^{-\lambda t} \tag{1}$$

where $\Gamma(s)$ is the gamma function. The parameters s and λ are called the *shape* and *scale* parameters respectively; they can be obtained from the mean and variance of the source. Let ρ be the lag-1 correlation. These correlations are typically very high for teleconference sources with $\rho = .98$ for the source studied in [59].

An accurate model for the number of bits (or cells) per frame is the DAR(1) model [59] which is a Markov chain determined by three parameters: the mean, variance, and ρ. The transition matrix is computed as:

$$P = \rho I + (1 - \rho)Q \tag{2}$$

where ρ is the autocorrelation coefficient, I is the identity matrix, and each row of Q is identical and

consists of the negative binomial (or gamma) probabilities $(f_0, ..., f_K, F_K)$ where $F_K = \sum_{k<K} f_k$, K is the peak rate, and the number of columns correspond to the number of (possibly aggregated) source rates. Each k, for $k < K$, corresponds to possible source rates less than the peak rate of K. The autocorrelation of the data decays exponentially up to a lag of 100 frames or so and then decays less slowly (giving rise to long-range dependence [18]). The DAR(1) model also has an exponentially matching autocorrelation and so matches the autocorrelation of the data over approximately hundred frame lags. This match is more than enough for traffic engineering since delay requirements do not permit network buffers to introduce delays more than the equivalent of three to four frame lags. When using the DAR(1) model, it is sufficient for admission control and traffic engineering, to know the mean, variance and lag-1 correlation of the source. The DAR(1) model can be used with any marginal distribution. It was used in [56, 57] to model entertainment and MPEG-2 coded video sequences with marginal distributions other than the gamma distribution.

4.1.2 Source Model for Active Sequences

For active sequences, the use of a single model based on a few physically meaningful parameters and applicable to a large number of sequences does not appear to be possible. One modeling strategy for these sequences is to develop a method for identifying scene changes and then to construct three models: one for scene lengths, one for the number of bits in scene change frames, and one for the number of bits per frame for frames within scenes. Below we describe in more detail the method used in [56].

A scene change is presumed to occur when a frame contains an abnormally large number of bits in comparison to its predecessors and successors. Therefore, at a scene change the second difference will be negative with large magnitude. The frames whose second difference, normalized by the average frame size in the recent past (say 25 frames), exceeds a threshold are taken to be scene change frames.

Because scene lengths appear to be uncorrelated, a model for scene lengths should characterize the distribution of the number of frames per scene. For the sequences examined in [56], the scene lengths are usually, but not always, characterized by a unimodal distribution such as the Gamma distribution, the Weibull distribution, or the Generalized Pareto distribution.

Next, a model is needed for the size of the first scene in a frame. As with scene lengths, these

scene-change frames are uncorrelated so it is sufficient to characterize the number of bits per frame. Some sequences have good Weibull or Gamma fits while others were not adequately characterized by any of the common distributions. Finding a model with a few parameters may thus not be possible.

Intra-scene frames can be modeled using a Markov chain or a DAR(1) model as in the case of video teleconferences. The marginal distributions are usually Gamma, Weibull or Pareto. Only the first frame after a scene change frame needs special consideration; it is adequately modeled by

$$Y_n = a + bX_n + \epsilon_n \tag{3}$$

where Y_n is the second frame in the scene, X_n is the first frame in the scene, and $\{\epsilon_n\}$ are independent and identically distributed zero-mean normal random variables.

4.1.3 MPEG-2 Source Model

The basics of MPEG coding are described in [19, 65]. Here we describe a few important aspects necessary to understand the source models. There are three picture (frame) types, called I, B, and P frames, which often, but not necessarily, appear repetitively and in patterns. For example, a common pattern is the following 15 frame pattern: I B B P B B P B B P B B P B B. This is a called a group of pictures (GOP). The GOP length can vary; neither the length of 15 or the I,B,P structures used here are universal. The P frames are predicted using previous I and P anchor frames. The B frames are predicted in both backward and forward directions by using I or P frames. The I frames are coded using intraframe techniques with no temporal prediction, so they have less compression than the B and P frames.

The histograms and autocorrelation functions of the various frame-types are examined in [57]. The different frame types all have similar characteristics, particularly high correlations. The histograms have a long right tail, which indicates that the bit-rate is quite bursty. Also, the short-term correlations are very high, which can be exploited for forecasting as illustrated using the source model described below [57].

The I frames have a log-normal distribution, and the autocorrelation of I frames decays geometrically and has the form $.823^k$ where k is the lag. Consequently, as for video teleconferences, I frames can be modeled by a Markov chain with a DAR(1) transition matrix similar to that used for

video teleconference. The matrix Q in this case has rows which are discretizations of the log-normal distribution instead of negative binomial. Excluding some outliers, the B and P frames also have log-normal distributions (although the distributions are not identical since their mean and variances are different). The correlation between successive B frames is very high (0.90). The sequence of frames consisting only of the first B frame in a GOP can also be modeled as a DAR(1) process with $\rho = .8$ and log-normal marginals. The succeeding B and P frames (after the first B frame) in a GOP are highly correlated. Hence, given a model for the first B frame in a GOP, subsequent frames can be obtained by generating correlated random pairs using a scheme such as that of [60].

4.2 Source Modeling Insights and Limitations

Source models have only been useful for traffic engineering for the class of video teleconferencing sources. This is because models are useful for these analyses only if they rely on a small set of physically meaningful parameters and if the same model can describe many sequences from a class. The mean, variance, and one-lag correlation are sufficient to characterize video teleconference sequences [58]. However, as stated above, adequate models for active sequences need many parameters because there are three marginal distributions (scene lengths, scene change frames, and intra-scene frames) to be specified and also at least one correlation. While there might be alternate models with fewer parameters, a single model applicable to a large number of sequences is unlikely to be found, given that the sequences to be modeled are statistically so different. The same appears to be true for MPEG coded sequences; i.e., models for specific sequences can be developed but models that describe many sequences adequately and with few parameters appear difficult to develop.

Despite these difficulties, a number of insights have been developed with the aid of source models that are useful in the engineering of networks to carry video. One insight is that video sources have very high short-term correlation which cannot be ignored for traffic engineering purposes. The one-frame-lag correlation is as high as .98 in some sources. This high short-term correlation must be accounted for during buffer dimensioning and admission control. Although it reduces multiplexing gains by requiring larger buckets for shaped-VBR (as we will see later), it also makes possible fairly accurate short-term traffic forecasts that can be used both for source-smoothing and for making rate-requests for carrying video over channels with explicit rate allocation such as ATM ABR. In [62], a very simple but successful forecasting rule is used: $X_{n+k} = \mu + \rho^k(X_n - \mu)$ where ρ is the

correlation coefficient, and μ, the mean number of cells per frame, is computed on-line. Another simple forecasting rule comes from the GBAR model [54]: given X_{n-1}, multiply it by B_n (a sample from an independent beta distributed random variable) and then add A_n (drawn from a gamma distribution). Both distributions have parameters which can be computed once from the mean, variance and 1-lag correlation of the teleconferencing sequence of interest.

Another insight is that video traffic exhibits fluctuations over different timescales. This empirical observation is explained by attributing short-timescale fluctuations to activity within a scene and long-timescale fluctuations to scene changes. This observation about fluctuation on multiple timescales is one of the motivations for transmitting video using renegotiated CBR (R-CBR) service [46]. Here buffering at the UNI is used to absorb short term fluctuations so that the network sees a quasi-CBR source whose rates change at the timescale of the long-timescale fluctuations. These rate changes result in a renegotiation of the CBR rate but since they happen on a long timescale the network support needed for R-CBR is much less than for VBR. However, to carry real-time video the scheme depends on distinguishing short-term fluctuations from long-term trend changes, which is difficult to accomplish accurately without so much delay that the trend change degrades performance.

The difficulties in developing useful models for anything other than video conferencing sequences were encountered even with simple coding algorithms (e.g. U-VBR using a fixed quantizer for all frames.) Obtaining realistic models for U-VBR obtained with more complex coding will be even more difficult, e.g., a coder which attempts to optimize its quality by adjusting the various coding such as GOP size, rate per block, choice of frame mode, etc.

However, S-VBR or C-VBR sequences are generated with an explicit rate constraint, imposed to accommodate the applicable leaky bucket parameters, and thus the observed transmitted rates will be easier to model, although the models are likely to be different from those that can be matched to U-VBR streams. There is a risk, however, that traffic engineering calculations based on U-VBR models would fail once the traffic as been shaped at the UNI (S-VBR), or at the video encoder (C-VBR).

In summary, while models should play a role in evaluating the system performance, they are unlikely to be useful until the interface between video application and network has been defined. We now consider in detail the various VBR system modes shown in Figure 1 which represent different

degrees of interaction between the video application and the network.

5 Unconstrained VBR

It is easy to obtain unconstrained VBR video. The simplest and most common method, used in most of the modeling experiments described above, is to fix the quantization step size in a video encoder. It is often assumed that this provides nearly constant quality video, but this is not strictly true. More sophisticated methods are necessary to achieve truly constant quality video (see for example [20, 21, 22, 23]).

There are a few straightforward techniques to reduce the bit-rate without sacrificing quality. As an example, consider the case when a long sequence, consisting of many different scenes, is encoded using a constant quantization step size. It is well known that scene changes result in masking so that the first few frames of the new scene can be coded at a relatively lower quality without affecting the perceived resolution. Thus a constant quantizer would not necessarily be efficient in this scenario. Along similar lines, the human visual system is more sensitive to errors in certain types of regions (e.g. textured vs. flat) so it is sensible to use bits where they are more useful perceptually, which again may not correspond to a constant quantization allocation.

Regardless of the method (a simple constant-quantizer algorithm, or a more sophisticated approach) selected to achieve approximately constant quality, the key feature of U-VBR is that the video output rate is selected solely for the purpose of video quality, and is not constrained by any networking requirements.

It may be argued that the lack of an explicit policing mechanism at the UNI makes the provision of QoS guarantees impossible for U-VBR, even when the source characteristics are known, because one misbehaving source can endanger these guarantees for all sources. However, there are three reasons to study U-VBR anyway. First, if network switches have fair queueing and per-flow buffer allocation, fair queueing provides implicit policing that enables QoS guarantees for U-VBR by isolating sources that send more traffic than their traffic descriptors allow. Second, a study of U-VBR indicates the performance that is achievable in the most optimistic scenario. Third, the research into traffic models necessary for U-VBR admission control provides insights regarding source behavior that are useful for S-VBR, C-VBR and F-VBR.

The two key challenges in carrying unconstrained VBR in networks with service guarantees

(such as guarantees on mean cell losses and delays) are i) finding appropriate source models that remain tractable for computing admission control while capturing relevant statistical aspects of the source-rate fluctuations, and ii) designing admission control algorithms that take into account various network features (such as different scheduling disciplines) without being overly conservative. A discussion of source modeling techniques for various types of sources was given in Section 4. We now describe the corresponding techniques for admission control.

5.1 Guaranteeing QoS for unconstrained VBR

Guaranteeing QoS for unconstrained VBR essentially implies solving the following traffic engineering problem: How many video calls can be simultaneously carried by a system with link bandwidth C bits/second, with a specified delay at most d and loss rate for source i at most clr_i? Designing admission control for ensured QoS depends on a solution to this problem.

In general, the number of calls that can be carried under QoS constraints needs to be determined for the whole network while taking into account the buffer management and scheduling disciplines used in network switches. For instance, if switches use per-connection buffering with fair queueing, then this has to be taken into account in the admission control computations. This means that the number of different calls that can be carried without violating any connection's QoS constraints must be determined assuming three things: a specific source model, the available buffer space that can be allocated to different connections (based on delay requirements), and the presence of fair queueing scheduling. Even for a single switch, determining this number without being very conservative is difficult when per-connection buffering and fair queueing are used. Furthermore, in the network case, the source statistics can significantly change after the first switch, making it difficult to compute capacity of downstream switches without being conservative.

When switches use shared buffers and first-in-first-out scheduling, the number of calls that can be carried can be quite accurately determined (provided that source models are accurate and have the properties necessary for analytical tractability). One method, called the Chernoff-dominant Eigenvalue method [49] applies to sources modeled by time-reversible Markov chains. Many other schemes (most of which are based on different methods for calculating an "effective bandwidth" associated with the source and its QoS requirements) can also be used to determine the number of calls that can be carried [73, 74, 75, 76, 77, 78, 79], for a specified loss rate and delay.

Note that the loss probability calculated is the aggregate loss probability. In some circumstances [59], phase effects caused by the periodic nature of video frames may result in individual sources seeing losses that are an order of magnitude or more different from those of other sources. Since the admission control algorithm computes total loss, quality of service guarantees may not be met for some admitted sources unless phase effects are eliminated. Also, the impact on QoS during transient periods caused by call arrivals and departures have to be accounted for, possibly by making the admission control more conservative. Another factor to consider is that the admission control calculations typically only consider the mean stationary loss probability. It may be necessary to consider the distribution of losses as well.

In [49], the multiplexing gain is found to be in the range of 3 to 4 whereas the peak-to-mean ratio for this traffic is 5. Here the multiplexing gain is calculated by comparing the number of calls carried to the number that would be carried if peak rate allocation were used (and not by comparing it to the number of CBR calls of "appropriate" rate that could be carried). The maximum possible multiplexing gain is therefore 5. Purely from a network multiplexing point of view, adequate multiplexing gain is extracted in this case using U-VBR, provided accurate source models are available. However, as we have said, such models are not available for most types of sources and also there is no mechanism to communicate complicated source models to the network. Instead, the simpler leaky bucket source descriptions are used, which can reduce the multiplexing gains for video considerably.

6 Shaped VBR

Given the difficulties in accurately modeling unconstrained sources, one solution has been for sources to declare a peak rate, average rate, and a maximum burst size to the network. The source shapes the incoming traffic to conform to the traffic contract either by adapting the traffic source's rate (such as by quantizer control for video sources) or by buffering in the shaping device. We assume the latter in this section, while the former will be discussed in Section 7. The most simplistic shaping scheme is to use a leaky bucket shaper where the shaper delays cells not eligible for transmission by holding them in a shaping buffer. More sophisticated schemes first reduce burstiness by smoothing the traffic before it enters the shaper. This smoothing can be done by quantizer control, or by introducing delays judiciously in combination with arrival forecasts. We defer discussions pertaining

to quantizer control to the section on Constrained VBR. Examples of S-VBR smoothing algorithms can be found in [62, 14]. If we were to shape unconstrained VBR by using leaky buckets, either the leaky bucket sizes would be large or the token rates would be close to the peak rate. By smoothing the unconstrained VBR source, the shaper parameters can be made more network "friendly". The network in turn polices the incoming shaped traffic to ensure that it conforms to the declared traffic contract. Non-conforming traffic is either discarded or marked for service as "best effort".

Arrival constraints can be specified by some non-decreasing function of time. The most popular are called the single and the dual leaky bucket constraints [24, 25]. A shaper or policer imposes a burstiness constraint on the source. This burstiness constraint $\hat{A}_i(\tau)$, is the maximum amount of traffic that source i can send in time τ. So $\forall\, s < t,\ A_i(s,t) \leq \hat{A}_i(t-s)$. $A_i(s,t)$ is the amount of traffic sent by connection i during time interval $[s,t]$. $A_i(s,t)$ depends on the traffic characteristics of the source and on the parameters of the shaper. The constraint imposed by a single leaky bucket shaper is completely characterized by the burstiness constraint $\hat{A}_i(\tau) = \sigma_i + \rho_i \cdot \tau$, where σ_i is the bucket size and ρ_i is the token rate. For a dual leaky bucket, the burstiness constraint is

$$\hat{A}_i(\tau) = min({\sigma_i}^1 + {\rho_i}^1 \cdot \tau, {\sigma_i}^2 + {\rho_i}^2 \cdot \tau) \tag{4}$$

where ${\sigma_i}^1, {\sigma_i}^2$ are the individual bucket sizes and ${\rho_i}^1, {\rho_i}^2$ are the token rates (usually picked to be the peak and long term average rates).

Given the burstiness constraints $\hat{A}_i(\tau)$ for all sources, a possible approach to network resource allocation is the worst-case approach, where traffic sources are assumed to be adversarial to the extent permitted by the shapers[‖]. Performance measures of interest are

- *worst-case delay:* $D_i^* = \max_{A_1,\cdots,A_N} \max_t D_i(t)$;
- *worst-case queue length:* $Q_i^* = \max_{A_1,\cdots,A_N} \max_t Q_i(t)$.

$D_i(t)$ and $Q_i(t)$ denote the delay and queue length for source i at time t. The maximization is over all possible arrival patterns and for all time intervals. This permits the possibility of collusion amongst sources. Both these parameters depend on the scheduling policy and the shaper parameters. The worst-case queue length determines the buffer size needed to avoid loss.

[‖]Note that by adversarial we mean that the sources not only utilize the maximum allowable bandwidth, but also produce the traffic patterns that are most difficult for the network to handle. For example, sources could be sending on/off data traffic instead of the agreed-to video traffic.

A deterministic worst-case service guarantee ensures that no packets are dropped or delayed beyond the guaranteed values. As in the stochastic case, this requires admission control tests to determine whether the network has enough resources to admit a new connection without risking service quality guarantees. An example of the use of deterministic delay bounds for shaped VBR is [47], which proposes traffic constraint functions that can be based on leaky buckets, multiple leaky buckets, or rate-interval pairs (called D-BIND). Exact and "sufficient" worst-case admission control tests are given for first-in-first-out, static priority, and earliest-deadline-first scheduling. The sufficient, as opposed to exact, tests are suggested because of the high computational complexity of the exact tests for some scheduling policies. For the generalized processor sharing (GPS) scheduling policy, Parekh and Gallager [66, 67] derive end-to-end delay bounds when the source traffic conforms to leaky bucket constraints.

Since in these worst-case analyses no independence of sources is assumed, the admission control tests assume that sources can simultaneously produce worst-case traffic. Hence, network utilization can be low while still considerably better than peak rate allocation. With 50 ms delay bounds, 56% utilizations for an MPEG sequence are obtained in [47], compared to 23% for peak rate allocation. However, the assumption that the sources can simultaneously produce worst-case traffic yields a strict admission control policy.

By taking sources to be statistically independent and allowing the possibility of a few losses, Elwalid, Mitra, and Wentworth [50] show that further multiplexing gains can be extracted while retaining the adversarial worst-case behavior of individual sources. For independent shaped sources, Mitra and Morrison [61] show that traffic processes which maximize the loss probability estimate (using Chernoff bounds) are periodic, on-off processes with random phases. Admission control schemes are proposed in [50] for use in this scenario of worst-case source behavior, statistically independent sources, and an allowed small loss in the network.

7 Constrained VBR

Only so much can be accomplished by smoothing the output of an unconstrained video encoder with S-VBR. In some cases, the output still may not satisfy the existing network constraints, or viewed another way, the required network constraints may be too expensive or lead to very low utilization. If so, to obtain quality video over packet networks it is necessary to incorporate the

encoder "in the loop" as shown in Figure 1. An encoder which can adapt the rate it submits to the UNI to satisfy both the network traffic constraints and the video system buffering constraints will be in a position to achieve the best possible *decoded* video quality, since the compliance to networking constraints will minimize network losses.

We focus on applications that require real-time encoding of video at the time of transport. However, the constraints and rate optimization methods we describe are also valid for video that is compressed and stored prior to transmission, if the bitstream is modified using techniques like Dynamic Rate Shaping (DRS) [26]. Using DRS to shape the compressed stream results in worse quality than if the same rate-shaping had been performed during compression. On the other hand, if the video is already compressed, various traffic prediction problems are much simpler because good estimates are available for the encoded video rate. For example, choosing renegotiation times for R-CBR, choosing traffic descriptor parameters, and performing work-ahead smoothing are all much simpler, because the end-to-end delay requirement for each bit is already known. If the requirements of an application do not mandate a stored bitstream, however, the advantages cited above in general will not offset the quality degradations resulting from using DRS instead of real-time adjustment of the quantizer during encoding.

We start by presenting the constraints on the compressed video traffic to meet simultaneously the buffering requirements in the video system and the traffic descriptor restrictions. We then show that optimization of video quality for a leaky bucket traffic descriptor can be achieved by separately optimizing the encoding rates and the channel rates. We close the section by discussing the statistical characterization of C-VBR video.

7.1 Optimizing video quality under delay and network channel constraints

7.1.1 Buffering constraints

As indicated in Section 3.2, in a real-time video communications system both encoder and decoder are attached to synchronous devices, and thus the end-to-end delay of a frame traversing the system should be constant. This is also true in a system where the information has been already encoded but continuous display at the decoder is required (i.e., without frame repeats or frame skipping). If some of the information corresponding to a video frame arrives at the decoder after the scheduled decoding time, the information will be useless and the frame will thus be considered lost. Since

the information received at the decoder is stored in a buffer before actually being decoded, this situation is called decoder buffer underflow.

Clearly, information is also lost if either the encoder buffer or the decoder buffer overflows. Encoder buffer underflow is not an issue in a VBR transmission environment by definition, since one can lower the transmission rate to zero if there is no data to be transmitted. In this section we describe the constraints such that neither the encoder or decoder buffer overflows or underflows.

Let $B^e(i)$ and $B^d(i)$ be the encoder and decoder buffer occupancies, respectively. Let $C(i)$ and $R(i)$ be, respectively, the channel rate during the i-th frame interval and the encoded source rate used for the i-th frame. As introduced in Section 3.2, ΔN is the number of compressed video frames stored in either the encoder or decoder buffer at any given time. Then, the constraints on the encoded rate $R(i)$ imposed by the requirements that neither the encoder or decoder buffer overflow or underflow are [13, 15]

$$C(i) - B^e(i-1) \leq R(i) \leq C(i) + B^e_{max} - B^e(i-1) \tag{5}$$

for the encoder, and

$$C(i+\Delta N) + B^d(i-1) - B^d_{max} \leq R(i) \leq C(i+\Delta N) + B^d(i-1) \tag{6}$$

for the decoder. These constraints depend on the previous encoded rates and the previous channel rates through the buffer occupancies $B^e(i-1)$ and $B^d(i-1)$. Detailed discussions of how these constraints affect the selection of the encoded rates $C(i)$ can be found in [13, 15, 27] so we refer the interested reader to these references and concentrate on the main results.

It can be shown [15] that to avoid decoder buffer underflow it is sufficient to guarantee that the encoder buffer state is such that

$$B^e(i) \leq B_{eff}(i) = \sum_{j=i+1}^{i+\Delta N} C(j), \tag{7}$$

where we define $B_{eff}(i)$ to be the effective buffer size. Equation (7) has the immediate intuitive interpretation that there must be sufficient channel rate to transport the data currently stored in the encoder buffer before it is needed by the decoder ΔN frames in the future. Thus, the constraints on the rate used to encode the current frame, $R(i)$, depend not only on the channel rate at the current time, but also on the channel rate in the future, ΔN time slots, $C(i+j)$ for $j = 1, \ldots, \Delta N$. Hence,

for generic channel constraints, selecting the encoding rates typically requires a joint approach that either simultaneously or iteratively selects the channel rates as well.

There are two major differences in the constraints for VBR encoded rates compared to the similar constraints for CBR. First, the encoder and decoder occupancies are not mirror images of each other as they are in CBR; therefore both must be considered when choosing $C(i)$ and $R(i)$ [13]. Second, there is an additional constraint implicit in CBR video that must be considered explicitly for VBR video: the sum of the bits used to encode any consecutive ΔN frames must never exceed the combined physical buffering capacity available at the encoder and decoder. This is because at any given time the bits for the ΔN frames that have been encoded but not yet decoded must be stored someplace. This can be expressed as

$$0 \leq \sum_{j=i}^{i+\Delta N-1} R(j) \leq B^d_{max} + B^e_{max}. \tag{8}$$

7.1.2 Network channel constraints

It is possible to manipulate the constraints in (5) and (6) into constraints on the channel rate at a given time depending on the encoded rates [13]. This can indicate, for example, a lower bound on the current channel rate in order to prevent starvation (i.e., underflow) at the decoder buffer. However, these constraints are not very useful for indicating an upper bound on the channel rate, since they just prevent the channel from sending more bits than have actually been encoded.

It is more useful to bound the channel rates using a traffic descriptor like a leaky bucket or sliding window. The general constraint imposed by a generic traffic descriptor whose state depends only on the previous state and the current channel rate is given in [15]. Here we consider only the simple case where the channel traffic is constrained by the leaky bucket traffic descriptor.

The leaky bucket constrains the input channel rates to keep within an imaginary buffer that has constant output rate ρ and size σ. The state variable, or counter, is the leaky bucket occupancy at time i which is given by $LB(i) = \max(LB(i-1)+C(i)-\rho, 0)$. The channel constraint is $LB(i) \leq \sigma$ at all times. This can be written in terms of the cumulative channel rates as

$$0 \leq \sum_{j=0}^{i} C(j) - i \cdot \rho \leq \sigma, \tag{9}$$

where the max(.) term in the definition of $LB(i)$ can be ignored if $C(i)$ is sufficiently large.

7.1.3 Optimizing video quality given a leaky bucket channel constraint

The constraints presented above have been used in one form or another by a variety of authors [28], [27], [15], since they were first presented in [13]. Reibman and Haskell [13] presented a rudimentary rate control based on the RM8 rate control algorithm for H.261 video. Subsequently, Hamdi *et al.* [6, 7] and Ding [28] have examined better ad-hoc approaches to quantizer selection given simultaneous buffer and rate constraints. More recently, Chen *et al.* [27] and Hsu *et al.* [15] have presented different algorithms for quantizer selection that optimize video quality given the simultaneous constraints. An iterative process is used in [27] to choose alternately the encoded rates $R(i)$ and the channel rates $C(i)$. Hsu *et al.* [15] extend the results of [29] for a CBR channel by jointly (i.e., simultaneously) selecting both the encoded rates and the channel rates.

However, these researchers have treated the selection of the encoding rate and the transmitting rate as a coupled problem with interacting constraints. A simpler optimization procedure is possible if we combine the constraint in equation (9) with a similar constraint for both the encoder and decoder buffers. As a result, the cumulative encoded rate is constrained by

$$\max\{0, \rho \cdot \Delta N - B^d_{max}\} \leq \sum_{j=0}^{i} R(j) - i\rho \leq \sigma + \min\{\rho \cdot \Delta N, B^e_{max}\}. \tag{10}$$

If the physical buffer sizes are chosen sufficiently large such that $B^e_{max} \geq \rho \cdot \Delta N$ and $B^d_{max} \geq \rho \cdot \Delta N$, this constraint reduces to the well-known constraint on the encoded rate imposed by a constant-rate channel,

$$0 \leq \sum_{j=0}^{i} R(j) - i\rho \leq B^E_{max}, \tag{11}$$

with virtual buffer size $B^E_{max} = \sigma + B^e_{max}$. Note that this is exactly equivalent to the constraint on the encoded bit-rates $R(i)$ that would result if we had a physical buffer of size $B^E_{max} = \sigma + B^e_{max}$ with drain rate ρ, since the quantity between the inequalities of equation (11) is the fullness of a virtual buffer with constant drain rate ρ.

Thus, the problem of encoder rate control in a leaky-bucket channel reduces to the well-known encoder rate control problem in a CBR channel, with the exception of the constraint imposed by equation (8). As a result, instead of a joint optimization to select the source and channel rates, a two step procedure is possible. That is, the encoding rates can be optimized without explicitly considering the actual channel rates, solely for the purpose of achieving optimal video quality.

Once the encoding rates are chosen, there is usually a range of choices for the actual channel rates. Thus, the channel rates can then be selected, for the purpose of increasing network utilization by submitting good (i.e., smooth) traffic characteristics to the network. Optimal smoothing algorithms [62, 14, 16] produce maximally smooth traffic using the available source buffers and tolerable end-to-end delay. Salehi *et al.* [16] have shown that this "work-ahead smoothing" can double statistical multiplexing gains for S-VBR traffic. It is unlikely that the same magnitude of improvement will be achieved for C-VBR because the stream is usually less bursty; however, smoothing will undoubtedly still be worthwhile.

7.2 Characterizing C-VBR sources

7.2.1 Constraint selection

There are two aspects of characterizing C-VBR sources. First, the designer must understand how to choose the appropriate constraint parameters (buffer sizes, traffic descriptors, and end-to-end delay) to obtain the required design goals (short delay, good received quality, and many calls or high utilization). In particular, it is known how to process an already compressed bitstream to determine the necessary constraint parameters, but it is not well known how to choose appropriate constraint parameters prior to encoding, which is vital for applications requiring real-time encoding and transport.

It is a straightforward task to process an already compressed bitstream to determine the leaky bucket parameter values that would be necessary for the stream to pass through the UNI without violating the associated policing constraints. In general, a continuum of possible choices is available (for example, all the way from largest leaky bucket and mean rate to no leaky bucket at the peak rate), all of which will provide a different allowable burstiness, although the actual burstiness of the video source does not change. Given a known call admission algorithm and a knowledge of the possible network buffer allocation, it is straightforward to determine which set of values will lead to the greatest utilization. An issue that must be addressed before either S-VBR or C-VBR will be carried on packet networks efficiently is the choice of appropriate traffic descriptor parameters before the compressed bitstream is available.

7.2.2 Modeling of C-VBR sources

Second, an understanding of the statistical nature of the resulting constrained VBR sources is useful for designing call admission procedures. Modeling of C-VBR sources is an open area for study. The goal of such a study would be to provide a more accurate characterization of the video traffic beyond the adversarial worst-case on-off traffic model. This could be useful for call admission and other networking tasks, because it is clear that typical video traffic does not behave adversarially.

Little work has been done in this area. The "straightforward" approach would be to gather video data generated by a C-VBR encoder that used a particular rate control algorithm to meet a particular channel constraint and then to model the resulting trace using techniques similar to those used for U-VBR. The difficulty with this approach is that the resulting model could not be used to understand the behavior of a C-VBR source operating with a different rate control algorithm or a different channel constraint.

An alternate approach would be to start with a source coded at a given "consistent" or constant quantization scale. Then using the methods of section 5 a statistical model could be designed to capture the "intrinsic"(i.e., unconstrained) behavior of the source. The model for C-VBR, then, would include a statistical component equivalent to a U-VBR model, followed by a simple multiplicative scaling function which would allow computation of the corresponding rate if a different quantizer had been used [30]. This model of C-VBR traffic is not easily analyzed. However, it does allow straightforward simulation to explore the effect of a constraint, or the choice of a rate control algorithm.

Several authors have proposed the idea of selecting constraints that are based on the imposition of a statistical model onto the source [31, 32, 33]. In this approach the policing function imposes specific statistical properties on the output traffic. One specific approach would be to determine "desirable" or useful statistical properties for the bitstream, map those into a policing function, and then force the encoded video to meet the constraints. Because it is possible for video (as opposed to data) to give comparable quality of service with different bit-rate patterns, it may be possible to determine desirable bit-rate characteristics from the network point of view and design the video to fit.

This approach has the advantage that the negotiation between the source and the network

about the connection *is done directly on the basis of the statistical model.* Consequently, the network can more readily perform the analyses required for call admission control. As an example of this approach, Heeke [33] proposes as the model a Markov chain with several states, each state corresponding to an output rate. This approach constrains not only the marginal distribution of the output rate, but also its correlations. If the associated parameters are appropriately chosen, subjective picture quality can be maintained [33]. Skelly *et al.* [32] and Voeten *et al.* [31] propose a histogram model and a "gabarit" (i.e., envelope of a Gaussian distribution), respectively, to constrain only the marginal distributions.

8 Trade-offs between quality, delay and bandwidth

We now present a quantitative and qualitative comparison of the three VBR modes discussed so far in terms of the three metrics introduced in Section 3, namely, video quality, end-to-end delay, and number of admissible calls. In the C-VBR mode, the flexibility introduced by allowing the encoder to optimize its parameter selection to meet the network traffic constraints creates a host of issues for the system designer to consider when choosing the constraints. All the parameters to be selected (encoder and decoder buffer sizes, leaky bucket size and drain rate, and end-to-end delay) impact the three performance metrics. In general, it may only be possible for the parameters to be selected to optimize one of the three metrics.

We use the optimization approach outlined in section 7.1.3 for our C-VBR experiments, but other quantizer selection algorithms [13, 28, 6, 7] could be used instead. We use two sets of sequences for our results. Figure 2 and Tables 1 and 2 use a 15,000-frame segment of the movie *Mission Impossible* compressed using JPEG to explore the trade-offs between U-VBR, S-VBR and C-VBR. JPEG enables a computationally simpler optimization of the rate selections for C-VBR than is possible using either MPEG or H.261. For C-VBR, we examine the leaky bucket constraint, since it has been studied most. Table 3 uses three 9000-frame, 5-minute teleconferencing sequences compressed using H.261 for examining the impact of smoothing U-VBR to obtain S-VBR **.

The following subsections each discuss one of the metrics of Section 3. We use the average PSNR as a measure of quality. For delay, we show only the impact of buffering, and assume all other delay terms are fixed. Thus, our estimate of delay will be proportional to the size of the

** More detail regarding these sources can be found in [14].

smoothing buffer (S-VBR) or the encoder buffer (C-VBR). For U-VBR we assume no buffering delay at encoder or decoder. Finally, the multiplexing gain is a function of the Leaky Bucket (LB) parameters of a particular source. A large LB rate and bucket size allows fewer calls to be carried for the same overall bandwidth and QoS.

8.1 Video quality gains

First, let us consider the achievable video quality. As indicated earlier, for a given average coding rate, the more the rate is allowed to vary between frames, the greater potential for better video quality. Because in C-VBR the virtual buffer size is larger than the physical buffer size, additional rate variations are allowed beyond those available solely with the physical buffers, and therefore better quality can be obtained. However, the corresponding potential quality improvement is limited, as it is bounded above by the quality achievable by a CBR system with a physical buffer the size of the virtual buffer, B^E_{max}. Furthermore, because of the physical buffering constraints of equation (8), the full flexibility provided by a larger virtual buffer may not be realizable and we may not be able to achieve this upper bound. In our simulations, including those in [15], the constraint in equation (8) does not affect the outcome very often. It primarily has the influence of preventing very large individual frames, because, for example, it does not allow an individual frame to be larger than the two physical buffers.

The notion of virtual buffer is illustrated in Figure 2(a), where we plot contours of constant PSNR for combinations of delay and LB size. As can be seen, the quality is in large part determined by the size of the combined physical buffer and LB, i.e., the size of the virtual buffer introduced above. Thus C-VBR can achieve the same performance as a CBR system, but with lower end-to-end delay. Alternatively, quality can be improved by increasing the leaky bucket size for a fixed physical buffer and average rate. However, the improvements in quality are relatively modest. Further, the improved quality comes at the cost of reduced multiplexing gain, because a larger LB allows burstier traffic for the same average rate.

The observation that the C-VBR quality is upper-bounded by that of a CBR channel with larger physical buffer is a significant one. As a result, the quality is clearly bounded by the quality of the video sequence coded with the same average rate but without any buffering or delay constraint. Thus, the constraint imposed by the average rate is more significant than that imposed by buffering,

LB Size (bits)	LB Rate (bits/frame)	U-VBR PSNR (dB)	C-VBR PSNR (dB)
2567928	173088	44.98	46.14
1569296	177528	44.98	46.19
810352	183080	44.98	46.28

Table 1: Comparison between U-VBR and C-VBR

particularly for relatively short sequences. This is also evidenced in Figure 2(b) where it can be seen that small increases in average rate (which corresponds to the LB drain rate) result in more significant increases in quality than comparable increases in LB size. This is intuitively obvious since increasing the LB rate increases the overall rate allocation, while increasing the LB size only affects the burstiness of the transmitted rates. Conversely, increases in LB rate also degrade multiplexing gains more significantly than increases in LB size do (for fixed total aggregate rate).

It is interesting to note that better quality can also be obtained for the same leaky bucket constraints by using C-VBR instead of U-VBR or S-VBR. Tables 1 and 2 show the pairs of leaky bucket parameters (size and drain rate) required such that the *Mission Impossible* U-VBR or S-VBR stream, respectively, do no violate the traffic descriptor. These leaky bucket parameters are selected to be just large enough to accommodate the worst-case burst in each stream. In our examples we select LB parameter pairs to represent both high rate/small bucket and low rate/large bucket situations, corresponding to policing near the peak or near the long term average, respectively. The tables show that the overall quality is improved if we then encode using the C-VBR mode, for the same leaky bucket parameters. This is because the quality in the hardest scene is identical (since this scene produces the burst size that forced those leaky bucket parameters), but the overall quality is improved because the easier scenes can have better quality. Thus, for the same LB constraints the C-VBR sequence produces a higher average rate. From one perspective, the comparison is not "fair" because the average rates are not equal; however, from another perspective, it is fair because since each source has the same LB parameters, each will be allocated the same network resources and will experience the same QoS. We note that, as discussed above, U-VBR has the best quality when the average source rate (rather than the LB drain rate) is fixed.

To conclude, while a leaky bucket rate constraint allows some quality improvements over a constant rate channel, the quality improvements are bounded and may not offset the increased

LB Size (bits)	LB Rate (bits/frame)	S-VBR PSNR (dB)	C-VBR PSNR (dB)
2554768	170872	44.98	46.08
783888	186400	44.98	46.17

Table 2: Comparison between S-VBR and C-VBR

cost to support the resulting traffic, particularly when physical buffer sizes in the video system already are large enough to accommodate several frames of compressed data. Also, for a given LB constraint, better quality can be achieved with C-VBR mode, and thus traffic engineering assumptions may have to be based on expecting this type of "greedy" traffic, rather than the traffic with lower average rate ("non-greedy") generated by either U-VBR or S-VBR.

8.2 End-to-end delay reduction

Next, we examine how a leaky bucket channel can reduce buffering delays. As noted above, the leaky bucket "looks" just like a physical buffer, so the leaky bucket constraint can be folded into both the encoder and decoder buffer constraints. Therefore, if the rate variability is absorbed not in a physical video buffer but instead in the network in the form of a leaky bucket constraint (i.e., in an "imaginary" buffer), then the physical delay is reduced without sacrificing the rate variability. Theoretically, we can obtain the same variability in the encoded rate that is available with a constant rate channel and decoder delay ΔN when we use a leaky-bucket channel with zero video buffering delay, provided $\Delta N = \sigma/\rho$, and $\sigma = B^e_{max} = B^d_{max}$. This is illustrated in Figure 2(a) where quality depends only on the virtual buffer size. Hence, larger LB sizes allow a lower overall physical delay.

Similarly, Table 3 illustrates the trade-off between end-to-end delay and multiplexing gain. It demonstrates the advantage of increasing the end-to-end delay for the purpose of smoothing the video traffic to improve multiplexing performance. Because the encoded bitstream is identical for both U-VBR and S-VBR, the quality is unaffected. Table 3 shows the multiplexing gains as a function of smoothing delay for three teleconferencing sequences. A delay of zero corresponds to U-VBR, while delays of one through four correspond to S-VBR with optimal smoothing within the delay constraint [62, 14, 16]. Statistical multiplexing gains (SMG) are computed as the ratio of the number of VBR sources that can be admitted without exceeding a cell loss rate of 10^{-6}

(computed using a simple networking model [14]), divided by the number of CBR sources that can be carried. The rate of the CBR source having equivalent quality is estimated to be the peak rate of the optimally smoothed S-VBR source using a delay constraint of 3 frames, corresponding to the 100 ms of delay typical in CBR video teleconferencing [14].

As can be seen, as the end-to-end delay increases to a few frames, the multiplexing gains increase significantly. Further increases to the delay beyond 3 or 4 frames yield marginal improvements. Hence, to send the same video quality with a shorter end-to-end delay requires burstier traffic for the same average rate, and therefore fewer calls can be multiplexed [14]. Similar statements can be made for C-VBR where again decreasing the delay (by for example exchanging physical buffer size for increased LB size) for fixed quality produces burstier traffic and reduced potential for multiplexing gain.

Sequence	Delay (frames)	Maximum VBR SMG	Maximum CBR SMG	Upper bound SMG (PMR)
A	0	1.94	0.33	3.53
	1	2.37	0.60	
	2	2.63	0.91	
	3	2.69	1.00	
	4	2.70	1.03	
B	0	1.00	0.46	1.84
	1	1.23	0.74	
	2	1.37	0.96	
	3	1.38	1.00	
	4	1.40	1.02	
C	0	1.49	0.38	2.55
	1	1.79	0.67	
	2	1.97	0.97	
	3	1.99	1.00	
	4	1.99	1.02	

Table 3: Maximum Statistical Multiplexing Gain for CBR vs. VBR. Compared to equivalent CBR with delay $\Delta N = 3$

Despite this fact, it may be advantageous for some applications like interactive communications via video teleconferencing to be designed with low delay. For example, a network provider may offer a service to those customers willing to pay a premium for low-delay transmission of highly interactive video conferencing. While such a service could be offered using CBR transport of video,

much higher bandwidth would be required, as illustrated in Table 3. Thus, highly interactive video conferencing will be cheaper using VBR video than CBR video.

To summarize, for a given quality, the designer can choose to optimize either the end-to-end delay or the multiplexing gain. Improvements in one will degrade the performance of the other.

8.3 Multiplexing gains

We discussed above the trade-off between multiplexing gain and delay, for a fixed quality. The remaining issue is to understand how the imposition of stricter channel constraints than those required by S-VBR affects the video quality. That is, if the channel constraints necessary to transport S-VBR do not produce satisfactory network utilization, or if they are too costly, to what extent does making the channel constraints stricter and using C-VBR instead of S-VBR impact both the networking performance and the video quality? We assume the delay is constant in this discussion. We also assume that network resource allocation and call admission are based on the LB parameters. Thus, two sources using the same set of LB parameters will see the same QoS, provided they both comply with the constraints.

As discussed, Table 2 indicates that C-VBR has better average quality than S-VBR for the same leaky bucket parameters. Therefore, it is clear that the same average quality can be obtained for the same delay with less demanding LB parameters (smaller bucket, lower rate) by using C-VBR. Figure 2(b) quantifies this for C-VBR; quality is maintained by increasing the leaky bucket size and decreasing the average rate.

However, a better quality measure than the PSNR averaged over the duration of the source may be the quality of the most demanding scene. During the most demanding scene, the S-VBR encoder operates near its rate constraint. Therefore, with C-VBR, reducing the LB parameters below what is necessary for S-VBR will degrade the quality of that scene, even though the overall average quality may be better.

Dual leaky bucket approaches may provide a way to increase the multiplexing gain without affecting the most demanding scene. In a dual LB scheme we use two buckets, matched to the short term and long term behavior, with small size/high rate and big size/low rate, respectively. We can select the small LB to match the required rate for the most demanding scene in the U-VBR sequence. Then the large LB can be chosen to limit the long term average of the sequence. Thus

one can maintain quality for the most demanding scene while reducing the overall average rate and thus increasing the multiplexing gain. See [34, 15] for details.

The set of results presented in this section indicate that while all the advantages (better quality, shorter delay, more calls) of VBR video are possible, there is a distinct trade-off among them. Hence, each application should be designed with a clear understanding of the specific objective and how achieving that objective will impact the other performance metrics.

9 Feedback VBR

Constrained VBR allows the video encoder to adjust its coding parameters to meet the constraints on the rate defined at the UNI. However, the agreed-to traffic descriptor constraint remains unchanged throughout the transmission and thus provides little knowledge about the state of the network at any given time (e.g. whether the network is congested). Thus, after the call has been in progress for some time, the constraint fixed at the beginning of the call may no longer be optimal (in terms of best video quality, lowest delay, and maximum number of admissible calls). The goal of feedback VBR (F-VBR) is to allow the admission of more calls by letting each call modulate the bandwidth it consumes depending on feedback received from the network. Thus F-VBR is analogous to C-VBR, except that the rate constraints are now time-varying. The constraint will vary according to the needs of the encoder when there is no congestion, but will vary according to the needs of the network during congestion intervals.

In section 2.2 we mentioned that unlike a file transfer application, video might benefit more from a low-rate, low-loss channel than from a high-rate, high-loss channel. If feedback is incorporated throughout the system, the encoder can adapt all of its parameters to match the current estimate of channel characteristics. If the exact rate at which the network can carry information without any impairments is known, then the best video quality will be achieved by compressing to fit exactly within that rate. This is because the varying quality produced by constraining the output rate by, for example, varying the quantizer during compression, is *far* less significant than that produced by any network-induced impairments incurred throughout the bitstream because of operating with excess rate.

In practice, it is unrealistic to expect that the network rate which guarantees impairment-free transmission will be exactly known. This is particularly true for large heterogeneous networks where

different segments of a particular end-to-end connection may be subject to different traffic mixes, and where indeed all the information for all the segments may not be available. In fact, except perhaps in private networks, complete knowledge of the network state is unlikely to be available. Consequently, feedback provided by each of the parties in a connection, or by intermediate nodes, will have to be relied upon to estimate network characteristics. In summary, knowledge of network state is likely to be imperfect and so losses will still occur. Thus, video systems can obtain better quality by also adjusting their coding parameters to adapt to the network's prevailing loss profile, and not just available rates.

In section 9.1, we consider explicit rate feedback using the context of the Available Bit Rate (ABR) service for ATM networks. ABR allows a very fine-grained (frequent) feedback from the network to the source. The feedback consists of information about the source rate which the network expects to transport successfully. In section 9.2, we contrast and compare ABR and Renegotiated CBR (R-CBR), for which the source rate is allowed to change but only at isolated instants in time. Finally, the example in section 9.3 illustrates that forms of feedback other than the network's available rate can be used advantageously in a packet video system. A joint design of the network and the video system, incorporating all aspects of feedback, will improve the overall video quality.

Note that the issue of supporting reliable end-to-end service over a time-varying lossy channel is also relevant to the transmission of video over a wireless link which might be subject to fading and other time-varying impairments. In a wireless scenario, variable rate transmission arises when using varying degrees of error protection (depending on the channel state) or when using Automatic Repeat reQuest (ARQ) protocols to retransmit lost data [35, 36]. [36] uses explicit feedback from the link state to modulate the output rate of a video coder and supports our argument that it is better to send fewer bits, which lowers the theoretically obtainable video quality but obtains more robust transmission and consequently better end-to-end quality than could be achieved by sending more bits with higher loss probability.

9.1 Explicit Rate Feedback

One example of an explicit rate feedback scheme is ABR, where in-band (using no separate signaling channel) Resource Management (RM) cells are periodically transmitted by each source to indicate the currently desired rate. The network may mark this rate downwards before returning the RM

cell back to the source if it is unable to provide the desired rate. The information in the returned RM cell can be used by the video encoder to constrain its output rate. The rate can be modified much more frequently than for R-CBR since RM cell processing is in-band. However, the overhead in the form of "control" information that is exchanged for each connection is higher.

Explicit rate feedback for video is considered in [52], [64], and [63]. In [52, 37], Kanakia *et al.* explore the case where the network provides a target coding rate to the video terminal, and the video encoder adjusts its quantizer to ensure that its output traffic fits into the available rate. Extensive simulations indicate that some cells are lost at the onset of a congestion period, but that in the steady state the video rate is well matched to the network-specified rate and the video quality is not significantly worse than when there is no congestion. Therefore, a graceful degradation of video quality can be achieved during congestion intervals.

One drawback of the scheme in [52, 37] is that the source is unable to specify its desired rate, but can only respond to rates provided by the network. As a result, equivalent video quality between different sources cannot be achieved if different sources have different rate requirements. Subsequently, [63] examines the case where the source is allowed to specify a desired rate. The network then computes a target rate for each source using an algorithm akin to weighted round-robin service. It then explores the robustness of the control scheme to inaccuracies in the sources' rate requests that might occur if the sources are required to request a rate prior to the actual encoding.

Using ABR, each source may request a Minimum Cell Rate (MCR) that can be used by the network for resource allocation and call admission. These processes can be simpler than computing estimates of effective bandwidth, which would be required in admission control for U-VBR services. Sources choose their MCR to provide the lowest quality video that the user will accept. If excess bandwidth is available, ABR allows sources to transmit at rates greater than MCR, thus enabling better quality.

The basic goals of an explicit rate control scheme are to manage congestion by having cooperating sources maintain a smooth output rate and to ensure that the aggregate rate never exceeds the capacity of the network. Two side-effects of this are that queue occupancies and cell loss rates are kept low for most switch service disciplines. However, to accomplish the former, it may be necessary to isolate sources that have a delay requirement from those that do not. This can be

accomplished, for example, by fair queueing.

As more calls are added to the network, the calls in progress can each reduce their rate to accommodate the new load. In this way, the quality of each degrades gracefully as more calls are added. A "cliff" effect of sudden losses when one more call is added can be avoided by careful rate allocation within the admission-controlled real-time class. Also, with if the real-time sources are isolated from the non-real-time sources as discussed, the non-real-time sources can then be used to absorb some of the bandwidth fluctuation so that only large changes will affect the real-time class. These feedback systems allow the video quality to degrade gracefully when the network undergoes unanticipated demands in the form of a sudden onset of congestion on a link. Furthermore, sources which only want to adapt minimally can request higher MCRs. For example, if the source only wants to be minimally adaptive, the MCR can be set just below the open-loop effective bandwidth.

From the network point of view, there are two other critical issues. First, the computation of the explicit rates to be fed back should satisfy some notion of fairness by providing equal allocation to equivalent sources. As a new call is added, the transient behavior of both the queues and the rate allocation may not achieve this goal. This issue is particularly difficult in the cases where different links have different source demands, different sources have different propagation delays, and different queueing and service disciplines (including FIFO queues) are used in the network. In general, the time it takes the rate allocation to converge is twice the propagation delay multiplied by the number of bottleneck rates. Since it is easier to achieve fairness if the sources' output rates are relatively constant during the convergence interval, it is desirable for the video sources to output smooth traffic.

The second issue is deciding when feedback is too late to be useful. The feedback information loses relevance if tries to capture the network (or link) state on a timescale shorter than the delay back to the source, since by the time this information reaches the encoder the network state may already have changed. The amount of feedback delay that can be tolerated is influenced by three factors: the decay of the autocorrelation of the U-VBR source traffic (which indicates whether the explicit rate is likely to match the source rate by the time it arrives), the amount of available switch buffering, and the end-to-end delay requirements of the video itself. In practice, even with a 200 ms propagation delay, packet losses are reduced by a factor greater than 4 compared to no feedback control [52]. However, longer feedback delays affect the transient behavior of the video

quality. That is, if feedback takes several frames to arrive at the encoder, some bits may be lost due to excessive rate in that time period.

Finally, there is the question of how the video system should select what rate to request from the network. The request rate must be large enough to ensure that all the traffic in the source buffer can meet the end-to-end delay requirements, just as is necessary in the channel rate selection for C-VBR. Furthermore, to assist the network in partitioning bandwidth fairly among sources, and to create a more predictable traffic stream and provide more accurate rate allocation, the source can use the source buffer and smooth its traffic to the extent allowed by the end-to-end delay constraint.

9.2 Renegotiated CBR

As far as video coding is concerned, ABR service can be viewed as R-CBR service with rate changes allowed on very small timescales. At the interface between the video encoder and the network, however, in R-CBR the source requests rates at the negotiation instants and the network accepts or rejects the rate, while in ABR the source request rates using rate management (RM) cells and the network indicates in returning RM cells what the allowed rates are. Further, the means by which the network will support the ABR and R-CBR services are different.

R-CBR [46] is based on the observation that video traffic exhibits burstiness on multiple timescales. There are variations on small timescales (of the order of buffering delays) that presumably correspond to variations within a scene, and there are rate fluctuations over periods much larger than buffering delays which presumably are due to inter-scene variations.

R-CBR's usefulness is premised on the notion of separation of timescales in multiplexing gains. Fast timescale fluctuations are absorbed by buffering whereas slow timescale fluctuations are absorbed by the multiplexing of independent sources in a bufferless system (averaging among sources). R-CBR moves all buffering to the edges of the network where it is used to absorb fast timescale fluctuations. Slow timescale fluctuations result in rate renegotiations. The network sees quasi-CBR sources. These can be multiplexed in the network using a bufferless model. (Knowledge of the distribution of rate changes is still necessary to determine renegotiation failure probabilities.)

Note that if leaky bucket shaping is used, slow timescale fluctuations cannot be absorbed without using very large buffers or without using very large token rates [14]. R-CBR allows token rates to be renegotiated, so a combination of small token rates and small buckets may be possible.

The concept is appealing provided signaling costs are low so that frequent and fast renegotiation is possible. However, a method to determine the renegotiation events is needed. The main difficulty is that detecting changes in the slow timescale is difficult since these have to be distinguished from fluctuations in the fast timescale. The changes have to be detected fast enough for new rates to be obtained before buffer overflow happens. A method for optimal renegotiation for non-real-time video is presented in [46]. However, no scheme is presented for real-time video.

The proposal in [63] to use rate management cells in ATM for rate requests can be viewed as an alternate mechanism for renegotiation (though the network is not providing CBR service). Here renegotiation is frequent and fast, and the source buffer occupancy combined with short-term traffic forecasts (which are not hard to obtain because of the high short-term correlation) can be used to set renegotiation demands. However, the scheme in [63] uses an explicit rate feedback mechanism such as ATM ABR. If a scheme like ATM ABR were used then to keep delays to levels acceptable for video it would be necessary to separate cells from real-time and non-real-time sources. This can be done easily in switches which support weighted fair queueing.

9.3 Feedback of cell loss rates

As mentioned above, even using feedback information to adjust the rates, each connection may still be subject to losses. Different loss rates have different impacts on the quality of decoded video. Moreover, different coding schemes provide different degrees of robustness to cell losses. Thus if the encoder can obtain knowledge of the channel loss characteristics, it can choose to maintain the same transmission rate but adjust its coding parameters the better to compensate for the losses. In particular, the video system can provide suitable feedback of this information for itself through a backchannel [38].

As an example, in MPEG, intra macroblocks (I-blocks) are used to limit the temporal extent of an error while short slices are used to limit the spatial extent of an error. However, both incur an overhead bit-rate that cannot then be used to send video information. Thus, if there are few losses, one would like infrequent I-blocks and greater use of long slices to improve compression efficiency. Alternatively, if there are many losses, one would like frequent I-blocks and short slices to provide resilience. Thus, the encoder can adapt the distribution of I-blocks and the length of slices based on the observed loss characteristics in the channel. Unlike in [39], where feedback regarding specific

individual cell losses is used in conjunction with sophisticated prediction techniques to completely eliminate error propagation, in this method of adaptation, the encoder adapts to the prevailing network conditions. As a result, some error propagation may occur, but the overall bitstream is more resilient to errors in the period before the feedback arrives to the encoder.

10 Conclusions and open issues

Our goals in this paper were to clarify terminology and to describe the various types of transmission modes which can be seen to be forms of VBR video. In so doing we have tried to draw some conclusions based on recent research as to what future VBR video implementations will be like. While the work we have described was primarily restricted to packet video applications, these are by no means the only applications where VBR video can play a role. Other applications include stored video, satellite transmission (channel sharing) and wireless.

In the case of playback of stored video, the maximum throughput of the storage device, e.g. a DVD player, could be much larger than either the average or peak rate of the video source, so data can be read in bursts. In this case, pure U-VBR is possible if physical memory is sufficient. This is equivalent to being guaranteed access to a high-rate channel without losses. In other applications using stored video, the access can occur over a network that limits the access bit-rate. As previously discussed, synchronous display of frames at the decoder imposes constraints on the transported traffic, and thus the same issues discussed for live video, such as decoder buffer underflow and robustness to losses, must be considered for stored video. Finally, if the rate must be modulated for stored video, it will be necessary to resort to Dynamic Rate Shaping [26], which can noticeably degrade the perceptual video quality if the rate reduction is substantial.

Channel sharing is already being used in direct satellite broadcasting systems. However most of the systems in place use rudimentary techniques, where sets of channels are grouped together and allocated an overall rate. Channels having content requiring lower rate can be matched with those requiring a higher rate. To improve these systems further will require dynamic rate allocation according to the "local content" variations, rather than the expected content (e.g. a newscast vs. a movie). While networking issues no longer apply, the problem of how to allocate rate to each of a number of VBR video channels so as to optimize the overall quality is by no means solved [40, 41, 42, 43].

Finally, we mentioned in Section 7 how wireless transmission, in particular point-to-point links, can be seen as transmission over a VBR link where the losses are induced by time-varying channel impairments rather than congestion. Thus the results on F-VBR have relevance to this problem as well.

Clearly, then, two conclusions can be drawn, not only for packet video but for the other applications of VBR video as well: (i) video applications that are aware of the channel characteristics will outperform those that are not, and (ii) feedback of channel state information as well as negotiation of traffic parameters are essential.

Considering video over the Internet, we see that little feedback is available. (This is for good reason, since in multicasting applications each user experiences completely different channel characteristics.) However, video applications such as 'vic' [44, 45] do an excellent job of supporting the service by making appropriate choices in the video coding strategy. For example, since packet losses are common, 'vic' uses an intraframe, conditional replenishment method, with frequent refreshing of all the image blocks. More sophisticated versions, including those based on scalable coding [45], are being developed. In particular, in receiver-driven layered multicasting, the layered video stream can be stripped to consist of only the video resolution that provides the best quality for the bandwidth that is available to the client.

In general, future video applications are more likely to be successful if they are aware of the channel characteristics and are able to utilize feedback information both to match their output rate to the available rate in the channel using encoder rate control, and to adapt the robustness of its coded video by intelligent use of frame refresh and adaptive error protection. In addition, layered coding can be used to provide additional resilience when the network is unable to estimate its rate accurately or when it has unanticipated demands for bandwidth. In particular, it allows selective shedding of bits when the link capacity is exceeded. It is not surprising, then, that the first published work considering feedback between the network and the video system within the context of explicit rate feedback used a layered codec [12].

There are still numerous open issues that remain to be tackled for a successful deployment of VBR video. In the source coding arena, despite the fact that layered coding has been studied in various forms (e.g. multiresolution, scalable) for a number of years, there are still concerns about its efficiency relative to single layer coding. Efficient methods that provide layered, motion

compensated coding in a lossy environment are also needed. We advocate that video systems use feedback about the network to optimize their traffic based on this time-varying information and static rate constraints. Thus, before either S-VBR or C-VBR can be effectively carried on packet networks, a means to choose appropriate traffic descriptor parameters *without* having the bitstream already compressed will be essential. The selection of constraints would ideally be driven by both networking and video criteria. Finally, on the networking side, further explorations are required into call admission and network management issues related to the transport of F-VBR and C-VBR traffic.

Acknowledgements: The authors would like thank Carlo Torniai (USC and University of Florence) for generating the R-D data for the Mission Impossible movie, Chi-Yuan Hsu (USC) for his help with the C-VBR experiments, and James H. Snyder (AT&T) and the anonymous reviewers for their detailed comments.

References

[1] W. Verbiest, L. Pinnoo, and B. Voeten, "The impact of the ATM concept on video coding," *IEEE J. on Sel. Areas in Comm.*, vol. 6, pp. 1623–1632, Dec. 1988.

[2] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Trans. on Comm.*, vol. 36, pp. 834–843, July 1988.

[3] P. Sen, B. Maglaris, N. Rikli, and D. Anastassiou, "Models for packet switching of variable-bit-rate video sources," *IEEE J. on Sel. Areas in Comm.*, vol. 7, pp. 865–869, June 1989.

[4] ATM Forum, *ATM User-Network Interface Specification, Version 3.0.* Prentice-Hall, 1993.

[5] R. Jain, S. Kalyanaraman, S. Fahmy, R. Goyal, and S.-C. Kim, "Source behavior for ATM ABR traffic management: An explanantion," *IEEE Communications Magazine*, vol. 34, pp. 50–57, Nov. 1996.

[6] M. Hamdi and J. W. Roberts, "QoS guaranty for shaped bit rate video connections in broadband networks," in *Proc. of Intl. Conf. on Multimedia Networking, MmNet'95*, (Aizu-Wakamatsu, Japan), Oct. 1995.

[7] M. Hamdi, J. W. Roberts, and P. Rolin, "Rate control for VBR video coders in broadband networks," *IEEE J. on Sel. Areas in Comm.*, vol. 15, pp. 1040–1051, Aug. 1997.

[8] "ATM forum traffic management specification version 4.0," *ATM Forum/af-tm-0056.000*, April 1995.

[9] Y. Wang, Q.-F. Zhu, and L. Shaw, "Maximally smooth image recovery in transform coding," *IEEE Trans. on Comm.*, vol. 41, pp. 1544–1551, Oct. 1993.

[10] Q. Zhu and Y. Wang, "Error control and concealment for video communications," *Proceedings of the IEEE*, Jul. 1997. Submitted.

[11] G. Karlsson and M. Vetterli, "Packet video and its integration into the network architecture," *IEEE J. on Sel. Areas in Comm.*, vol. 7, pp. 739–751, June 1989.

[12] M. W. Garrett and M. Vetterli, "Joint source/channel coding of statistically multiplexed real time services on packet networks," *IEEE/ACM Trans. on Networking*, vol. 1, pp. 71–80, Feb. 1993.

[13] A. R. Reibman and B. G. Haskell, "Constraints on variable bit-rate video for ATM networks," *IEEE Trans. on CAS for video tech.*, vol. 2, pp. 361–372, Dec. 1992.

[14] A. R. Reibman and A. W. Berger, "Traffic descriptors for VBR video teleconferencing over ATM networks," *IEEE/ACM Trans. on Networking*, vol. 3, pp. 329–339, June 1995.

[15] C.-Y. Hsu, A. Ortega, and A. R. Reibman, "Joint selection of source and channel rate for VBR video transmission under ATM policing constraints," *IEEE J. on Sel. Areas in Comm., Special Issue on Real-Time Video Services in Multimedia Networks*, vol. 15, pp. 1016–1028, Aug. 1997.

[16] J. Salehi, Z.-L. Zhang, J. Kurose, and D. Towsley, "Supporting stored video: reducing rate variability and end-to-end resource requirements through optimal smoothing," in *ACM Sigmetrics Conference on Measurement and Modeling of Compu ter Systems*, pp. 222–231, May 1996.

[17] W. S. Tan, N. Duong, and J. Princen, "A comparison study of variable bit rate versus fixed bit rate video transmission," in *Austrailian Broadband Switching and Services Symposium*, pp. 134–141, 1991.

[18] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long range dependence in variable bit rate video traffic," *IEEE Transactions on Communications*, vol. 43, pp. 1566–1579, February/March/April 1995.

[19] J. Mitchell, W. Pennebaker, C. E. Fogg, and D. J. LeGall, *MPEG Video Compression Standard*. New York: Chapman and Hall, 1997.

[20] I. Dalgic and F. A. Tobagi, "A constant quality MPEG-1 video encoding scheme and its traffic characterization," in *International Picture Coding Symposium*, (Melbourne, Australia), pp. 105–110, March 1996.

[21] M. R. Pickering and J. F. Arnold, "A perceptually efficient VBR rate control algorithm," *IEEE Transactions on Image Processing*, vol. 3, pp. 527–532, September 1994.

[22] P. J. van der Meer, R. L. Lagendijk, and J. Biemond, "Local adaptive thresholding to reduce the bitrate in constant quality MPEG coding," in *International Picture Coding Symposium*, (Melbourne, Australia), pp. 117–121, March 1996.

[23] A. Ortega, M. W. Garrett, and M. Vetterli, "Toward joint optimization of VBR video coding and packet network traffic control," in *Proc. of the 5th Packet Video Workshop*, (Berlin), March 1993.

[24] J. S. Turner, "New directions in communications (or which way to the information age?)," *IEEE Commun. Mag.*, vol. 24, pp. 8–15, October 1986.

[25] G. Niestegge, "The leaky bucket policing method in the atm network," *Int. Journal of Digital and Analog Communication Systems*, vol. 2, pp. 187–197, 1990.

[26] A. Eleftheriadis and D. Anastassiou, "Constrained and general dynamic rate shaping of compressed digital video," in *Proceedings, 2nd IEEE International Conference on Image Processing*, (Washington, DC), pp. III.396–399, October 1995.

[27] J.-J. Chen and D. W. Lin, "Optimal bit allocation for coding video signal over ATM networks," *IEEE JSAC*, Aug. 1997.

[28] W. Ding, "Joint encoder and channel rate control of VBR video over ATM networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, pp. 266–278, April 1997.

[29] A. Ortega, K. Ramchandran, and M. Vetterli, "Optimal trellis-based buffered compression and fast approximation," *IEEE Trans. on Image Proc.*, vol. 3, pp. 26–40, Jan. 1994.

[30] D. M. Lucantoni, M. F. Neuts, and A. R. Reibman, "Traffic models for variable-bit-rate video," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 176–180, April 1994.

[31] B. Voeten, F. V. der Putten, and M. Lamote, "Preventive policing in video codecs for ATM networks," in *Fourth International Workshop on Packet Video*, pp. G1.1–G1.6, 1991.

[32] P. Skelly, M. Schwartz, and S. Dixit, "A histogram-based model for video traffic behavior in an atm multiplexer," *IEEE/ACM Trans. on Networking*, vol. 1, pp. 446–459, Aug. 1993.

[33] H. Heeke, "A traffic-control algorithm for ATM networks," *IEEE Trans. on Circ. and Sys. for Video Tech.*, vol. 3, pp. 182–189, Jun. 1993.

[34] A. Ortega, M. W. Garrett, and M. Vetterli, "Rate constraints for video transmission over ATM networks based on joint source/network criteria," *Annales des Télécommunications*, vol. 50, pp. 603–616, Jul.-Aug. 1995.

[35] M. Khansari, A. Jalali, E. Dubois, and P. Mermelstein, "Low bit-rate video transmission over fading channels for wireless microcellular systems," *IEEE Trans. on Circ. and Sys. for Video Tech.*, pp. 1–11, Feb. 1996.

[36] C.-Y. Hsu, A. Ortega, and M. Khansari, "Rate control for robust video transmission over wireless channels," in *Proc. of Visual Comm. and Image Proc., VCIP'97*, (San Jose, CA), Feb. 1997.

[37] H. Kanakia, P. P. Mishra, and A. R. Reibman, "An adaptive congestion control scheme for real time packet video transport," *IEEE/ACM Transactions on Networking*, vol. 3, pp. 671–682, Dec. 1995.

[38] C. Horne and A. R. Reibman, "Adaptation to cell loss in a 2-layer video codec for ATM networks," in *1993 International Picture Coding Symposium*, (Lausanne, Switzerland), March 1993.

[39] M. Wada, "Selective recovery of video packet loss using error concealment," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 807–814, June 1989.

[40] G. Keesman and D. Elias, "Analysis of joint bit-rate control in multi-program image coding," in *SPIE Visual Communications and Image Processing vol. 2308*, pp. 1906–1917, 1994.

[41] B. G. Haskell and A. R. Reibman, "Multiplexing of variable rate encoded streams," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, pp. 417–424, August 1994.

[42] M. Balakrishnan and R. Cohen, "Global optimization of multiplexed video encoders," in *Proc International Conference on Image Processing*, (Santa Barbara, CA), Oct 1997.

[43] D. T. Hoang and J. S. Vitter, "Multiplexing VBR video sequences onto a CBR channel with lexicographic optimization," in *Proc International Conference on Image Processing*, (Santa Barbara, CA), Oct 1997.

[44] S. McCanne and V. Jacobson, "`vic`: a flexible framework for packet video," in *Proc. of ACM Multimedia'95*, (San Francisco, CA), pp. 511–522, Nov 1995.

[45] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE Journal on Selected Areas in Communications*, August 1997.

[46] M. Grossglauser, S. Keshav, and D. Tse, "RCBR: A Simple and Efficient Service for Multiple Time-Scale Traffic", Proceedings of the ACM SIGCOMM'95 Conference, Sept. 1995.

[47] D. Wrege, E. Knightly, H. Zhang, and J. Liebeherr, "Deterministic Delay Bounds for VBR Video in Packet Switching Networks: Fundamental Limits and Practical Trade-Offs", *IEEE/ACM Transactions on Networking*, pp. 352-362, June 1996.

[48] A. Elwalid, and D. Mitra, "Effective Bandwidth of General Markovian Sources and Admission Control of High Speed Networks", *IEEE/ACM Transactions on Networking*, pp. 329-343, 1993.

[49] A. Elwalid, D. Heyman, T. V. Lakshman, D. Mitra, and A. Weiss, "Fundamental Bounds and Approximations for ATM Multiplexers with Applications to Video Teleconferencing", *IEEE Journal on Selected Areas in Communications: Special Issue on Fundamental Advances in Networking*, pp. 1004-1016, August 1995.

[50] A. Elwalid, D. Mitra, and H. Wentworth, "A New Approach for Allocating Buffers and Bandwidth to Heterogeneous Regulated Traffic in an ATM Node," *IEEE J. Select Areas of Communications: Special Issue on Fundamental Advances in Networking*, Vol. 13, No. 6, pp. 1115-1127, August, 1995.

[51] M. Garrett, and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic", Proceedings of ACM SIGCOMM 1994, August, 1994.

[52] H. Kanakia, P. P. Mishra, and A. R. Reibman, "An Adaptive Congestion Control Scheme for Real-Time Packet Video Transport", Proceedings of the ACM SIGCOMM'93 Conference, Sept. 1993.

[53] H. Kanakia, P. P. Mishra, and A. Reibman. Packet Video Transport in Atm Networks with Single-Bit Feedback. In *Proceedings of the Sixth International Workshop on Packet Video*, Portland, Oregon, September 1994.

[54] D. Heyman, "The GBAR Source Model for VBR Videoconferences", *IEEE/ACM Transactions on Networking*, Vol. 5, No. 4, pp. 554-560, Aug. 1997.

[55] D. Heyman, and T. V. Lakshman, "What Are the Implications of Long Range Dependence on Video Traffic Engineering" *IEEE/ACM Transactions on Networking*, Vol. 4, No. 3, pp. 301-317, June 1996.

[56] D. Heyman, and T. V. Lakshman, "Source Models of Broadcast-Video Traffic", *IEEE/ACM Transactions on Networking*, Vol. 4, No. 1, pp. 40-48, Feb. 1996.

[57] D. Heyman, T. V. Lakshman, and A. Tabatabai, "Statistical Analysis of MPEG2-Coded Video Traffic" Proceedings of Symposium On Multimedia Communications and Video Coding: A Celebration of the Centennial of Marconi's Invention of Radio Transmission, Brooklyn, NY, Oct. 1995.

[58] D. Heyman, T. V. Lakshman, A. Tabatabai, and H. Heeke, "Modeling Teleconference Traffic from VBR Video Coders", Proceedings of ICC '94, pp. 1744-1748, May 1994.

[59] D. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical Analysis and Simulation Study of VBR Video Teleconference Traffic in ATM Networks", *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 49-59, March 1992.

[60] M. Melemed, "TES: A Class of Methods for Generating Autocorrelated Uniform Variates", *ORSA Journal on Computing*, 3, pp. 317-326, 1991.

[61] D. Mitra, and J. Morrison, "Multiple Time Scale Regulation and Worst Case Processes for ATM Network Control", *Proceedings of CDC*, 1995.

[62] T. J. Ott, T. V. Lakshman, and A. Tabatabai, "A Scheme for Smoothing Delay Sensitive Traffic Offered to ATM Networks", Proceedings of IEEE Infocom 1992, pp. 776-785, May 1992.

[63] T. V. Lakshman, P. Mishra, and K. K. Ramakrishnan, "Transporting Compressed Video over ATM Networks with Explicit Rate Feedback Control" *Proceedings of Infocom 1997*, April 1997.

[64] P. P. Mishra. Fair Bandwidth Sharing for Video traffic sources using Distributed Feedback Control. In *Proceedings of IEEE GLOBECOM*, Singapore City, Singapore, Nov. 1995.

[65] P. Pancha, and M. El Zarki, "MPEG coding for variable bit rate video transmission", IEEE Communications Magazine, pp. 54-66, May 1994.

[66] A. K. Parekh, and R. G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: the Single-Node Case." *IEEE/ACM Transactions on Networking*, June 1993, Vol. 1, No. 3, pp. 344-357.

[67] A. K. Parekh, and R. G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: the Multiple Node Case." *IEEE/ACM Transactions on Networking*, April 1994, Vol. 2, No. 2, pp. 137-150.

[68] D. M. Cohen and D. P. Heyman "Performance Modeling of Video Teleconferencing in ATM Networks", *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 408-420, Dec. 1993.

[69] M. R. Frater, P. Tan, J. F. Arnold, "Variable Bit Rate Video Traffic on the Broadband ISDN: Modeling and Verification", *in The Fundamental Role of Teletraffic in the Evolution of Telecommunications, L. Labetoulle and J. W. Roberts Eds.*, Elsevier 1994.

[70] F. Yegenoglu, B. Jabbari, Y. Zhang, "Motion-classified autoregressive modeling of variable bit rate video", *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 42-53, Feb. 1993.

[71] M. Krunz and A. Makowski, "A Source Model for VBR Video Traffic Based on M/G/infinity Input Processes," University of Arizona, Department of Electrical and Computer Engineering CENG-TR-97-112, 1997.

[72] M. Krunz and H. Hughes, "A Traffic Model for MPEG-Coded VBR Streams", *Proceedings of the ACM SIGMETRICS Conference on the Measurement and Modeling of Computer Systems*, pp.47-56, 1995.

[73] R. Guerin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks", *IEEE JSAC* **9**, pp. 968–981, 1991.

[74] R. J. Gibbens and P. J. Hunt, "Effective bandwidths for the multi-type UAS channel", *Queueing System* **9**, pp. 17–28, 1991.

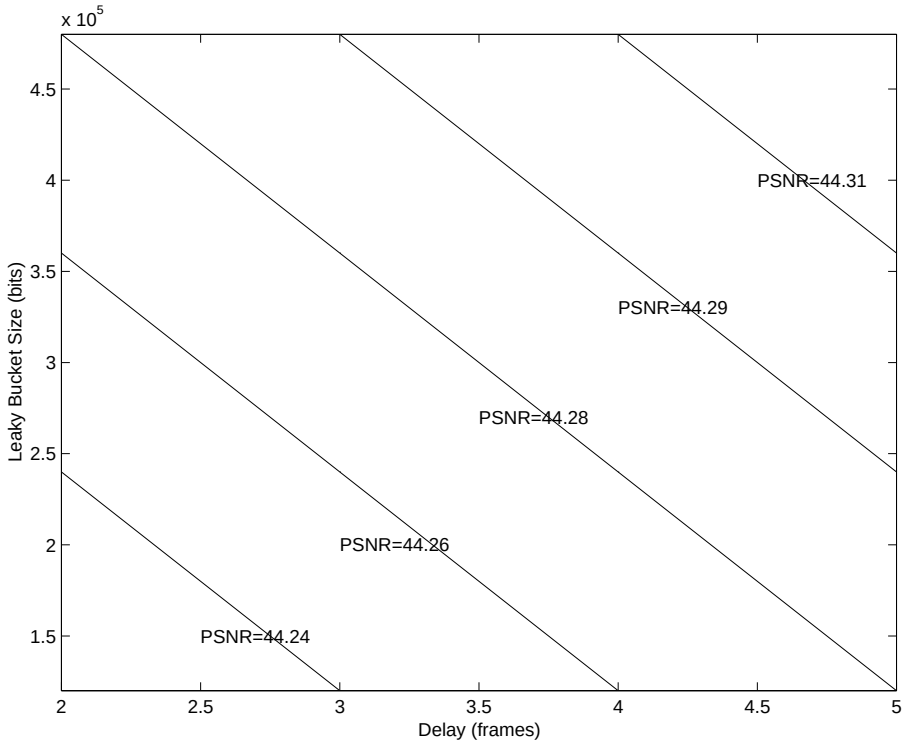
[75] J. Y. Hui, *Switching and Traffic Theory for Integrated Broadband Networks* Boston: Kluwer, 1990.

[76] F. P. Kelly, "Effective bandwidths at multi-type queues", *Queueing Syst.* **9**, pp. 5–15, 1991.

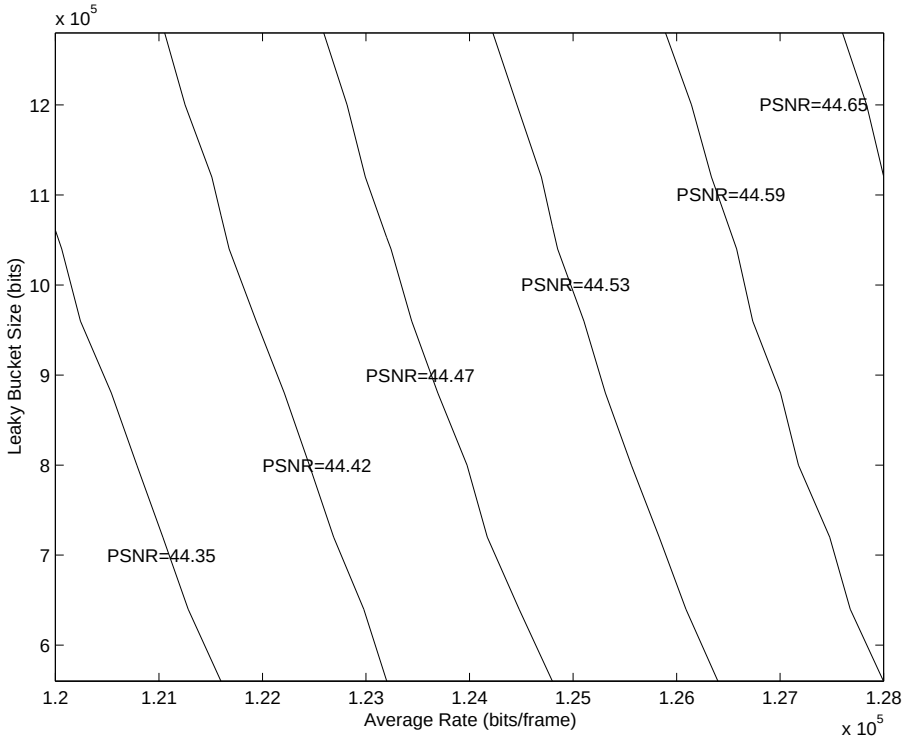
[77] G. Kesidis, J. Walrand and C. S. Chang, "Effective bandwidth for multiclass fluids and other ATM sources", *IEEE/ACM Trans. Networking*, **1**(4), pp. 424–428, 1993.

[78] J. W. Roberts, "Performance evaluation and design of multiservice networks", Final Report of the COST 224 Project, Commission of the European Communities, 1992.

[79] W. Whitt, "Tail probabilities with statistical multiplexing and effective bandwidths for multi-class queues", *Telecommun. Syst.* **2**, pp. 71–107, 1993.



(a)



(b)

Figure 2: (a) Contours of constant PSNR for different combinations of end-to-end delay and leaky bucket size. The LB rate is kept constant at 120,000 bits/frame. (b) Contours of constant PSNR for different combinations of LB size and average rate. The delay is kept constant at two frames.