An Adaptive Congestion Control Scheme for Real Time Packet Video Transport

Hemant Kanakia, Partho P. Mishra, and Amy R. Reibman, *Member, IEEE*

***Abstract*— In this paper we show that modulating the source rate of a video encoder based on congestion signals from the network has two major benefits: the quality of the video transmission degrades gracefully when the network is congested and the transmission capacity is used efficiently. Source rate modulation techniques have been used in the past in designing fixed rate video encoders used over telephone networks. In such constant bitrate encoders, the source rate modulation is done using feedback information about the occupancy of a local buffer. Thus, the feedback information is available instantaneously to the encoder. In the scheme proposed in this paper, the feedback may be delayed by several *frames* because it comes from an intermediate switching node of a packet-switched network. This paper shows the proposed scheme performs quite well despite this delay in feedback. We believe the use of such schemes will simplify the architecture used for supporting real time video services in future nationwide gigabit networks.**

I. Introduction

TWO DISTINCT approaches have emerged recently for supporting real time video over packet-switched networks. One approach is for the network to offer a quality of service similar to that currently available over the telephone network. This requirement has directed the design of traffic control schemes that guarantee constant or nearly constant bandwidth to each video connection [10], [12]. A connection of this type is sometimes referred to as a constant bitrate (CBR) channel. The primary advantage of this approach is the hardware and software required for the coding and decoding video signals is identical to that used currently for transmission of video signals over (circuit-switched) telephone networks. An alternative approach, explored in this paper, is for the network to make a variable bitrate (VBR) transmission capacity available to a video connection. This approach uses network resources more efficiently by exploiting packet-switching and the inherent burstiness of the data generated by a video encoder.

The actual video encoder and decoder mechanisms (or video codec) required for each approach are essentially identical. The rate control systems, however, are somewhat different. In each case, the actual bit-stream produced by the video encoder has a variable bitrate. For CBR, however, the encoder uses a buffer to smooth the generated variable rate stream into a constant rate stream. A decoder buffer is also required at the receiving end. To ensure that neither the encoder nor the decoder buffers overflow or underflow, the encoder uses the buffer occupancy levels to modulate its rate of data generation. Since the buffer contents can be known locally, this information is available instantaneously.

VBR encoders have the potential to provide a constant quality video signal. This is possible if the values of the encoding parameters remain constant and there is no loss of information due to packet losses. However, the bandwidth provided by a packet-switched network is generally time-varying. As long as a network is lightly loaded, it has the bandwidth available to carry the variable bitrate stream at a constant quality. Under heavy loads, however, there could be loss of information due to buffer overflows or excessive delays. The degradation of signal quality due to such overloads can be reduced by providing special overload control mechanisms in the network and/or in the coder.

In this paper, we study the performance of an overload control strategy that uses feedback from the network to modulate the source rate. We show that the feedback control mechanism performs quite well under a wide range of conditions, for example with scene changes, addition and deletion of controlled and uncontrolled traffic sources etc. In particular, the results show that during periods of congestion, the proposed mechanism can reduce the input rate from video sources substantially with a very graceful degradation in the image quality. For example, in one of our experiments a fully loaded link suffers a sudden reduction of 40% of its total capacity. When the proposed feedback scheme is used, the degradation in the quality of video transmissions carried over the link is hardly noticeable to users. However, in the absence of feedback control, the quality of the video sequences is quite poor. Our results disproves the prevailing belief that feedback-based control is not useful for real time video traffic in high speed wide area networks.

The feedback available from the network is qualitatively different from the feedback information used in CBR video encoders. First, when feedback information is sent from network switches, there is a significant delay before this information is available to the encoder. Second, the goal of the feedback control mechanism of a CBR encoder is simply, to control the buffer occupancy level of a single queue drained at a constant rate; the system we consider is more complex in that there are multiple queueing points and the service rate at each queue for a single stream varies over time depending on the intensity of cross-traffic.

The notion of using feedback information to modulate the source rates of video streams in a packet switched network has been proposed earlier in the context of sending slow-

Manuscript received May 5, 1994; revised January 16, 1995; approved by IEEE/ACM Transactions on Networking Editor D. Raychanduhuri.

The authors are with AT&T Bell Laboratories, Murray Hill, NJ 07974 USA.

IEEE Log Number 9415912.

1063-6692/95$04.00 © 1995 IEEE

scan video pictures (6–8 frames/s) of fairly small size over relatively low-speed networks that do not allow bandwidth reservations. In this environment, the quality of the generated images is quite poor and the effectiveness of using any type of feedback mechanism is hard to evaluate. The contribution of our work is in demonstrating that feedback control mechanisms result in a graceful degradation in the perceptual quality under periods of congestion. Our results also show the usefulness of the proposed mechanisms even when networks provide resource reservation/admission control facilities.

This study differs in one important respect from previous studies of video transport in a packet-switched network [9], [11], [14], [17], [19] in that, we use actual video sources as traffic generators and we use the perceptual quality of images as the main performance metric. In order to derive the perceptual quality we use actual video sources to drive a network simulator. The network is simulated as a collection of switches, links and hosts. In the simulation experiments, we principally focus on studying the *transient response*, namely how the signal quality degrades when congestion occurs and how well the control mechanism reacts to allow recovery from congestion. We also study the long-term behavior in terms of the average packet-loss probabilities and link utilization for representative image sequences comprising many scene changes and examine the impact of increasing the feedback delay in the system. We do not attempt to make a statistical characterization of the multiplexing behavior in terms of the long-term average packet-loss probabilities or the average number of video connections that could be supported for a given bandwidth, when the proposed scheme is used. Measurements of such statistical parameters would require the use of appropriate analytical models or representative long traces for video sources. We believe that no such models/traces have gained widespread enough acceptance to be useful.

The rest of the paper is organized as follows. In Section II, we discuss the statistical properties of coded video sources and their service quality requirements. We then discuss the different overload control mechanisms that have been proposed for VBR video transport over packet switched networks. In Section III, we describe the adaptive feedback control mechanism used in this paper. In Section IV, we discuss our experimental methodology, the topology of the network, and experimental results. Section V concludes the paper with a summary and description of future work.

II. Real Time Packet Video

The transmission of uncompressed, digitized, full-motion color video pictures, e.g., NTSC quality signals, is expensive. The average bandwidth requirement per video connection is about 166 Mbps. While the optical fibers that provide this kind of capacity are cheap, the relatively high cost of other factors such as disk storage capacity, high-speed electronic components, and installation of long-distance wires continue to favor using compression technology for video transport. For this reason, only compressed video traffic is considered in this paper. Although most of the current applications are of the talking heads variety, future applications are likely to use entertainment quality signals. We believe these types of applications will dominate the workload. Accordingly, the video sequences used in these experiments are of entertainment quality, with a picture size of 352 by 240 pixels and a frame rate of 30 frames per second.

A. *Service Quality Requirements*

The main metric of service quality for video connections, as far as a user is concerned, is the perceptual quality of the received images. A subjective measure of perceptual quality, called the mean opinion score (MOS), has been used for years to compare and tune coding algorithms. The MOS is calculated from the ratings given by a sample of human observers under controlled conditions. In our experiments, we have relied primarily on perceptual quality, although, the judgement was more informal than in the MOS computation. The image quality was judged by us along with a small group of our colleagues.

The best objective (although imperfect) measure of the spatial signal quality is the signal-to-noise ratio for the reconstructed video images. The SNR is an imperfect measure of perceptual quality because the perceptual quality of a sequence of frames depends, in a complex way, on the quality of each frame in the sequence. For example, it has been observed that the overall perceptual quality of a long sequence is often determined by the poorest quality in the sequence. We use the per-frame SNR and the average SNR value over a sequence of frames as crude indicators of the perceptual quality one is likely to observe.

B. *Statistical Characteristics*

The modeling of coded video traffic sources may prove invaluable in the effort to design integrated services networks. General models that reflect the characteristics of codecs, however, are not yet available. This restricts the ability to generalize observations made on the basis of specific analytical models. However, certain traffic characteristics of VBR sources have been generally accepted [14], [17], [18], [21]. We use the sample traces shown in Figs. 1 and 2 to demonstrate some of these features. These figures show the number of bits per frame generated by an motion picture experts group (MPEG) coder for a particular image sequence, over 50 and 850 frames, respectively [8].

We make the following observations about the source traffic characteristics:

1) *Burstiness:* From Fig. 1, it is obvious that the video traffic sources are bursty. In general, it has been observed that the burstiness of the sources depends on the information content of the images and the particular coding algorithm employed.
2) *Abrupt changes:* A change of scene or a change in the background of the picture can cause the data generation rate to change abruptly. For example, in Fig. 2, there is an abrupt drop in the number of bits generated from 190 kbits per frame to 75 kbits per frame between frame 629 and frame 634.

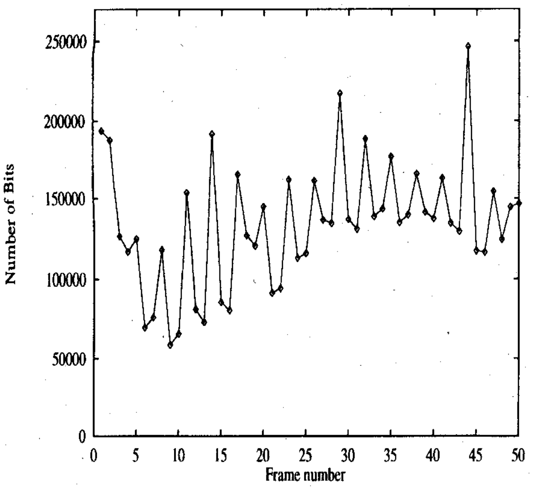


Fig. 1. Short output trace for an MPEG coder.

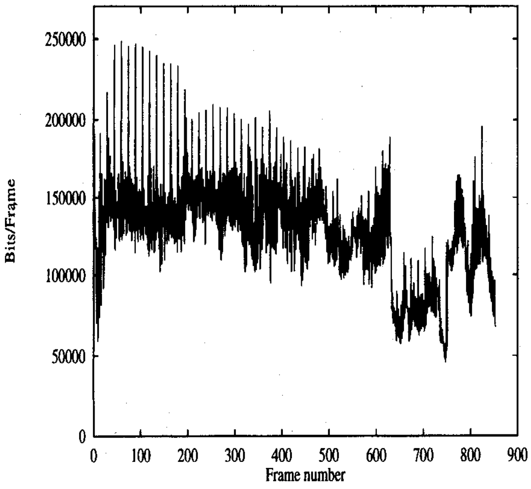


Fig. 2. Long output trace for an MPEG coder.

3) *Persistence:* A scene can last for several hundred frames. An implication of this is that the number of bits generated per frame is likely to be highly correlated. This may be demonstrated by computing the autocorrelation coefficients. Fig. 3 shows the autocorrelation coefficients for increasing inter-frame distances for a long sequence of frames generated by an MPEG coder. The oscillation observed here is because the coder generates three different types of frames – I, B, and P. The autocorrelation for only B frames of the same trace, shown in Fig. 4, is much stronger.

4) *Drifts:* There are also long-term variations (over several thousand frames) in the statistical parameters of the video sources. The superposition of these long-term drifts with the burstiness observed over smaller time intervals complicates the modeling and estimation of the parameters of a real time video source.

C. Transport Issues

The ISO MPEG has proposed standard algorithms that can be used to compress entertainment or educational video material for storage or transmission. In the experiments reported in this paper, an MPEG-1 video codec was used to generate the traces for two reasons. First, we wanted to use entertainment quality pictures. Second, the three different frame types and the resultant fluctuation in the data generation rate over successive frames makes the problem of designing an effective feedback control system harder. Since both JPEG and H.261 algorithms generate bit streams showing a high degree of autocorrelation, the proposed scheme should work well with those standards as well.

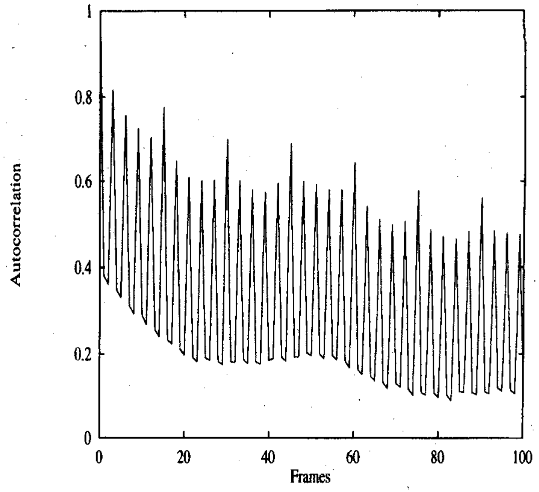


Fig. 3. Sample autocorrelation plot for all frames of an MPEG coded video data trace.

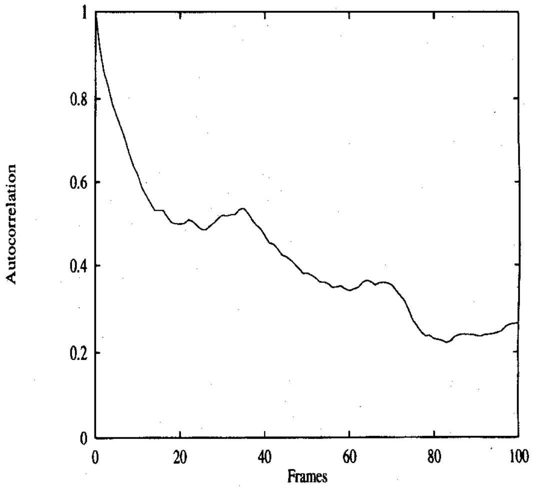


Fig. 4. Sample autocorrelation plots for only B frames of an MPEG coded video data trace.

The burstiness of variable bitrate traffic along with the need to provide certain performance guarantees implies the necessity for overload control and admission control mechanisms when transporting this type of traffic in a packet-switching network. Various overload control strategies have been proposed in the literature. This includes using complex scheduling mechanisms at network switches, error concealment and forward error recovery techniques, prioritized data transmission, traffic shaping, and multi-layered coding. A complete discussion and comparison of these approaches is outside the scope of this paper. Our approach to overload control involves a video source adjusting its rate based on explicit feedback signals received from network switches.

Admission control mechanisms are used to accept or reject new connections at connection setup time. The choice of overload control mechanism strongly affects the admission control policies for real time traffic. When network feedback is used to adapt the traffic offered to the network, resource reservations could be made based on the minimum bandwidth required by a connection. There is no need to estimate the peak or effective bandwidth required by a connection. Intuitively, such an approach would allow more connections to be admitted than when the effective or peak bandwidth has to be taken into account. If connections do not require a minimum transmission quality, then an admission control mechanism is not required for use with the feedback based congestion control mechanism.

III. Proposed Control Mechanism

In this section, we describe the control mechanism used for adapting the encoding rate. The goal of the mechanism is to minimize the impact of traffic overloads on the perceptual quality of image transmissions. We do not simply reuse one of the many feedback control mechanisms proposed earlier for the control of bursty data traffic. This is because the goals of feedback control for data and video traffic are different. Existing schemes proposed for data traffic attempt to prevent congestion so that the average throughput or end-to-end delay is optimized. Using such a scheme would not necessarily maximize the quality of image transmissions, since the signal quality for two streams with the same average throughput and delay could be quite different.

The proposed scheme is based on predicting the evolution of the system state over time. Such a predictive control scheme has been shown to work quite well in preventing congestion for bursty data traffic [15]. The predicted system state is used to compute the target sending rate for each frame of video data. The adaptation policy strives to keep the bottleneck queue size for each connection at a constant level.

In our design of the control mechanism, we make the following assumptions. We assume the video encoder has a rate control mechanism that can be used to match the target rate computed by the feedback mechanism, in terms of the number of bits per frame. The encoder rate control mechanism modulates the sending rate by adjusting the values of parameters used in the encoding process. We assume all packets belonging to a particular frame are transmitted at regularly spaced intervals over the frame duration. To allow this spacing of packets, an entire frame needs to be stored at a transmitter before the packets are created and injected into the network. We also assume that video packets from all connections are serviced in first-in-first-out (FIFO) order from a single queue at a switch. Although in this section we only consider a point-to-point video connection, the extensions required for a multipoint connection are briefly described in Section V.

Before describing the controller, we first define a few terms. The parameters defined are per connection. We denote the target sending rate for the nth frame in the image sequence by λ_n and the service rate at the bottleneck for the nth frame by μ_n. The bottleneck is defined as the switch with the minimum service rate available for the given connection.

The amount of data waiting at the bottleneck at the beginning of the nth frame is denoted by x_n. We assume that the frame rate (number of frames displayed per second) of a video connection does not vary once the connection is established and that this rate is given by $1 \ / \ F$. We denote the target queue size for the connection at the bottleneck by x^*. Two additional parameters, δ and *gain*, are used to tune the controller characteristics.

Since the feedback information is not available instantaneously, neither μ_n nor x_n are known at the instant at which the transmission of the nth video frame begins. Hence, estimates of these quantities, denoted by $\hat{\mu}_n$ and $\hat{x}_n$, are used to calculate the target sending rate. The target sending rate for the nth frame, λ_n, is calculated as

$$\begin{aligned} \lambda_n &= \lambda_{n-1} + \delta, \ \ if \ \ x_{n-k} = 0 \\ &= \hat{\mu}_n + \frac{x^* - \hat{x}_n}{\text{gain} * \text{F}}, \ \ \text{otherwise.} \end{aligned} \quad (1)$$

The goal of the controller equation described above is to keep the buffer occupancy level for the controlled connection, at the bottleneck, close to the target value, x^*. When the reported buffer occupancy level is nonzero, the sending rate λ_n is calculated so the buffer occupancy level is driven to x^*, over $\frac{1}{\text{gain}*\text{F}}$ time units. Thus, the value of *gain* controls how rapidly the buffer occupancy value approaches the target buffer size. When the reported buffer occupancy level is zero (the first case of (1)), the sending rate is increased linearly until the reported queue size becomes nonzero. This is referred to as the linear start-up phase with the value of δ controlling the increase in the sending rate. x^*, δ, and *gain* are system-wide parameters which provide control over the dynamics of (1).

The mechanism used to provide feedback information to the sources should be robust enough to handle situations where the location of the bottleneck might change rapidly over time. Each switch monitors the buffer occupancy and the service rate per connection. The buffer occupancy information is a count of the number of queued packets for the connection at the instant the feedback message is sent. The rate information is the number of packets transmitted for the connection in the time interval between two feedback messages. We describe two possible implementations of the feedback transmission mechanism. In the first implementation, the per-connection state information is appended periodically to a data packet for the corresponding connection. At the destination, this information is extracted and sent back to the source. A switch updates the information fields in a packet only if the local service rate is lower than that reported by a previous switch along the path. An alternative implementation for the feedback transmission mechanism is to use a separate control packet sent back along the path of the connection toward the source. The feedback information is reported multiple times per frame interval. The latter implementation is used in our simulation experiments.

The estimated buffer occupancy, $\hat{x_n}$, is calculated as

$$\hat{x}_n = x_{n-k} + \sum_{i=n-k+1}^{i=n-1} \lambda_i - k * \hat{\mu}_{n-k} \tag{2}$$

where x_{n-k} is the last known buffer occupancy level at the start of the $n-k$th frame. The value of k depends on how old the feedback information is. We calculate the exact value of k from the timestamp information included in a packet. The buffer occupancy values are available from the feedback messages sent multiple times per frame interval. In our implementation, a switch does not have explicit knowledge about frame boundaries. Consequently, the reporting period is not phase-synchronized to a frame interval. Since the reporting period is not synchronized to the start of a frame, a linear interpolation between the two buffer occupancy values reported around the start of the $n-k$th frame time is used to derive the value of x_{n-k}.

The estimated service rate, $\hat{\mu}_n$, is calculated as

$$\hat{\mu}_n = \mu_{n-1} * \alpha + \hat{\mu}_{n-1} * (1 - \alpha). \tag{3}$$

This is a first order auto-regressive filter with a forgetting factor α. The value of α could be either constant or variable [4]. Our experience suggests the estimator tracks the actual service rate much better with α computed as

$$\begin{aligned} \alpha &= 0.25 * E^2/\sigma_{k+1} \\ E &= \mu_k - \hat{\mu}_k \\ \sigma_{k+1} &= E^2 * 0.25 + \sigma_k * (1 - 0.25) \end{aligned} \tag{4}$$

where E and σ are the estimation error and the estimate of the squared estimation error, respectively.

When α is computed in this manner, the estimated service rate is not affected by small changes in the service rate. However, if the actual service rate changes abruptly, the estimated service rate tracks the change quite rapidly. In contrast, when a fixed value of α is used, a choice of constants is necessary such that either a controller reacts quickly to large changes or has better immunity to noise in steady state.

For the JPEG or H.261 coding standards, there is no difference between the frames generated by the encoder, in terms of the coding mechanism used for each frame. For such coding schemes, the above procedure works well. A slightly different procedure is used for the MPEG standard which generates data in three types of frames – I, B, and P frames. As evidenced by our experiments, the performance of the controller improves when it explicitly distinguishes between each type of frame by keeping a separate service rate estimator for each. The main change needed in the procedure described above comes while computing the target rate using (1), the estimator $\hat{\mu}_n$ used depends on the type of the nth frame. We would like to emphasize that the switches need not be aware which coding standard is used.

The three separate estimators of service rates are maintained as follows. Say that the received information applies to frame k. Each time an information packet is received, a counter is incremented by the number of packets sent out in that period. When the value of the counter exceeds the value of λ_k (the number of packets sent out by the source for that frame) the value of the service rate for the frame is computed as $\mu_k^x = \frac{\lambda_k}{t}$, where t is the time taken to send the λ_k packets at the bottleneck and x is the type of the kth frame.

The encoder tries to match the target sending rate by adjusting the values of the parameters used in the encoding algorithm. In our implementation of the encoder, the adjusted parameter is the quantization (Q) factor whose value can vary from 1 to 31. A Q-factor value of three produces very high quality images. Lower values of the Q-factor increase the data generation rates substantially but produce only marginal improvements in the image quality. At the other end of the scale, the human eye can generally distinguish visual artifacts for Q-factor values greater than 20. Thus, the rate matching algorithm chooses values of Q within these bounds. In general, the encoder should not allow the value of the Q-factor to exceed a maximum value determined to provide the minimum acceptable picture quality. We have also introduced a mechanism in the encoder that tracks the changes in the Q-factor and damps the fluctuation in its value. There are two desirable effects in this type of damping. First, the fluctuation in the perceptual quality across successive frames of the image sequence is reduced . Also, it leads to better perceptual quality when there is a change of scene. For the same value of the Q-factor, a complex image has a higher data generation rate than a simpler image. Thus, when a scene change occurs, the data rate remains higher than the target for the next few frames due to the damping in the change of the Q-factor value. As a result, a scene change to a more complex image frame will have a higher data generation rate for the source and force other sources sharing the network path to reduce their sending rates.

IV. Our Methodology

We have studied the performance of the proposed mechanism by using actual video traces to drive simulations. This appears to be the only sound methodology available at the moment to study the effects of traffic overloads on the quality of video transmissions. Although the modeling of video sources has received a great deal of attention in the recent past [11], [13], [14], the generality of these models has not yet been established. Moreover, none of these models can provide a measure for the perceptual quality of images.

The main performance metrics used in this study are the signal to noise ratio (SNR) and the perceptual quality of the image. The per-frame SNR is calculated by comparing the decoded frame with the original unencoded image. The average SNR is computed by averaging the per-frame SNR over the entire sequence of frames. Our comparison of the perceptual quality is based on an informal approach with ratings given by a small number of colleagues. The other performance measures used are the queue occupancy, the link utilization and the number of packets lost. The queue occupancy affects the end-to-end delay which is important for interactive applications while the utilization of bottleneck resources affects the efficiency with which network resources are used. The packet losses provide a crude indication of the degradation in signal quality.

In order to allow very general network topologies and workload patterns and to provide application, transport and network layer functionality, we use a network simulator. The application protocol component at a sender reads a trace and passes data to the transport protocol component. This component packetizes the data into variable size packets so an integral number of macroblocks is carried in a single packet with the maximum packet size being 500 bytes. The packetization process begins after data for an entire frame has been passed to the transport protocol component. The transmission of the packets is spaced apart equally over the frame duration. Feedback information received from the network is used by the transport protocol component to calculate the target sending rate for the next frame.

The target rate is matched by assuming the number of bits generated is linearly proportional to the number of macroblocks processed. The procedure is simple enough to be implemented in hardware. Our experience with the above method for target matching indicated the actual rate generated by the encoder is almost always within 5% of the target. This difference is insignificant for the outcome of our experiments, as verified by rerunning experiments with these rates. The transport protocol component at a receiver collects the arriving packets into a buffer. At the start time of every frame (as indicated by the display system), packets belonging to that frame are removed from the buffer and passed up to the decoder.[1] A trace of the received data is then decoded to produce real time images as well as to compute the SNR values.

For the initial set of experiments, we compare the performance of the proposed feedback control scheme with that of two nonadaptive schemes differing in the way they drop packets when the buffers are full. In the first scheme, henceforth referred to as the *random loss* scheme, an arriving packet finding the buffer full is dropped. This scheme is expected to provide the worst quality for a given level of congestion in the network. In the second scheme, referred to as the *priority loss* scheme, a two-level priority mechanism is used to drop packets. If the buffers are full when a high-priority packet arrives, a low-priority packet is ejected to make room for the high-priority packet. In our experiments, priority is assigned to a packet based on the type of frame to which it belongs. Packets from B frames are assigned a low priority while packets from P and I frames are assigned a high priority. This priority assignment is based on the observation that the loss of B frame packets does not affect the signal quality in other frames, while the loss of P and I frame packets adversely affects the signal quality in other frames. This mechanism is perhaps the only way to prioritize packets in conjunction with the MPEG–I standard. The comparison between the different schemes is based on the SNR and the perceptual quality of pictures obtained with these schemes under identical traffic conditions. For comparison, we also provide the quality that would be obtained if the original image sequences were coded using a constant quantization factor and transmitted using a loss free quantization factor and transmitted using a loss free network. We refer to this as the *perfect channel* case.

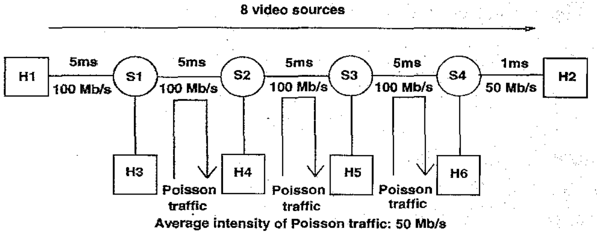


Fig. 5. Network configurations used in the simulations.

A. Network Topology and Traffic Description

In this section, we describe the network topology, the values of parameters and the various traffic scenarios used. Currently, an accurate model of traffic on future networks that will carry video, data and voice applications is not available. Thus, it would be impossible to justify a particular traffic model for inducing congestion. We have used three different plausible models for how traffic overload might occur. In the first of these models, the congestion is induced by abruptly reducing the bandwidth available to the existing video traffic. This models a situation where a group of new connections, routed through the bottleneck link, switch on simultaneously. This kind of sharp transition stresses the transient behavior of feedback based control mechanisms. In the second model, the increase and decrease of the available bandwidth is more gradual with new connections joining and leaving the congested path at periodic intervals. In the third model, congestion is caused by the statistical fluctuations in the traffic intensity of video traffic. The focus is on the long-term statistical variations due to scene changes in the video sequences and the impact this has on video quality. We also use this last model to address the question of what the limitations of using network-generated feedback are, i.e. when is feedback too late to be useful?

In summary, the four sets of experiments presented in this section are as follows:

1) Sudden reduction in the available bandwidth at the bottleneck link.
2) Gradual increase/decrease in traffic at the bottleneck.
3) Traffic fluctuations at the bottleneck due to scene changes.
4) When will feedback be too late to be useful?

The first two sets of experiments use the network configuration, shown in Fig. 5. The link bandwidths are all 100 Mbps except for the bottleneck link which is 50 Mbps. The link propagation delays are chosen so the round trip propagation delay is 42 ms, equivalent to the coast-to-coast propagation delay in the U.S. The receiver buffers up to 100 ms of video data in its playback buffer. For the third and fourth set of experiments we use a simpler configuration in which the end-to-end propagation delay is kept the same but the intermediate switches S1, S2, and S3 are removed to speed up the simulations. The duration of the simulations for the first two sets of experiments is 5 s, while the duration for the latter two sets is 250 s. In all the simulations, multiple video sources share a common path. The multiplexing helps the study of robustness and the fairness of the (distributed) feedback control mechanism.

[1] Packets arriving later than their display time are discarded.

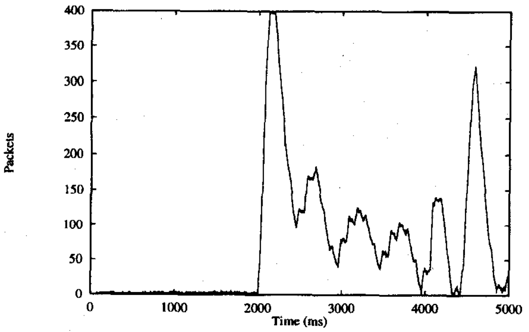


Fig. 6. Buffer occupancy at the bottleneck for experiment described in Section IV.B.1.

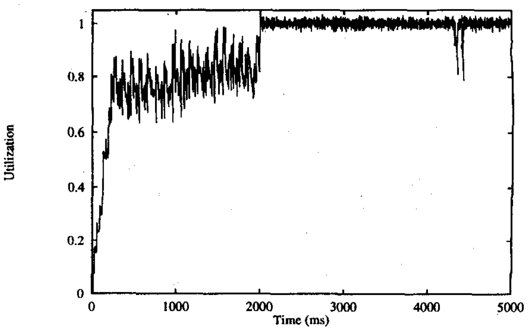


Fig. 7. Link utilization at the bottleneck for experiment described in Section IV.B.1.

TABLE I
AVERAGE SNR'S OF VIDEO STREAMS FOR EXPERIMENT DESCIBED IN SECTION IV.B.1

Stream	Perfect Channel	Feedback Control	Priority Loss	Random Loss
autumn	38.11	37.02	36.80	30.15
ball of wool	40.10	39.25	33.57	28.33
bicycles	38.63	34.99	30.46	24.86
birches	38.08	35.02	32.19	25.13
cheer	38.44	33.83	30.64	24.21
ferris wheel	40.13	38.10	33.03	25.69
horses	39.52	38.71	33.76	28.39

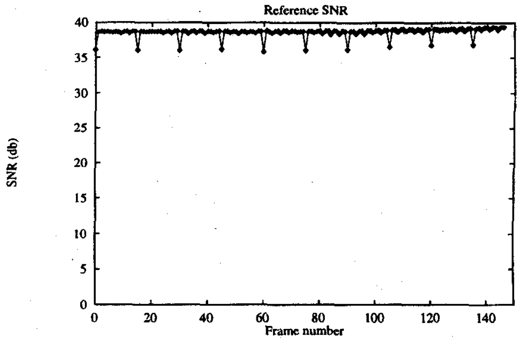


Fig. 8. SNR plots for bicycle sequence with *perfect channel* for experiment described in Section IV.B.1.

The source traces chosen by us to drive the simulations correspond to a set of video image sequences typically used for evaluating coding algorithms. For the first two sets of experiments, we use eight different sequences for each connection, while for the last two sets, we use the same (long) sequence indexed at different points. The sequences used include two bicyclists riding through a forested path, an automobile driving through a forest in full autumn colors, a ferris wheel and so on. Due to a disk error, we lost the trace recorded at a receiver for one of the eight sources, after the simulations. Thus, we have only calculated SNR values for seven sources. The video traces used in these experiments were produced with a Q factor value of 4, which resulted in very good quality images. In the experiments in which the feedback control scheme was used, the Q factor values ranged from six to nine after the reduction in bandwidth. The data rate for the coded video sources, when they were not throttled due to network conditions, varied from 2 to 8 Mbits/s at 30 frames/s.

In the first two sets of experiments, we use bursty data sources with Poisson arrivals to provide cross-traffic on each link. The aggregate bandwidth of these sources is about 50 Mb/s and packets from these data sources are serviced with the same priority as those from video traffic sources. The traffic from these sources is not flow-controlled. These sources allow us to study the effect of uncontrolled sources with stochastic packet arrivals on the service quality seen by the controlled connections; this also serves to test how well the service rate and buffer estimation techniques work.

B. Sudden Jumps in Traffic Intensity

In this scenario, the congestion is induced by reducing the available bandwidth at the bottleneck link at time 2000 ms from 50 Mbps to 30Mbps. We will present results observed for three groups of experiments each differing in the amount of buffering available at switches and the degree of synchronization among the video sources. Each group consists of three experiments with identical traffic conditions in which either the *feedback-based, priority loss* or *random loss* schemes is used. The results are compared to those for the perfect channel.

1) Large Buffers and Out-of-Phase Sources: In this group of experiments, each of the connections comes on at intervals of 33 ms (namely one frame interval). There are 400 packet buffers of size 500 bytes shared among all the active connections at each of the output queues. Fig. 6 shows the evolution of the queue occupancy at the bottleneck. The average SNR values for each video source is shown in Table I. The first column in the table is for the perfect channel. The next three columns show, respectively, the values for the feedback-based scheme, the priority loss scheme and the random loss scheme, respectively. Figs. 9–15 show the variation in the per frame SNR for the adaptive scheme for the *bicycles* sequence.

It is shown in Fig. 6 that the queue occupancy does not start reducing until almost five round-trip times after the reduction in bandwidth. This is a result of the heavy damping built into the feedback controller. As a result, in this experiment a total of 107 packets are lost across all the connections before the sending rates are reduced with a concomitant decrease in the buffer occupancy. Although all the lost packets belong to two frames, these losses cause the signal quality to drop to a lower level and stay there until an I frame is sent. This is a result of the use of inter-frame coding for B and P frames. This effect,

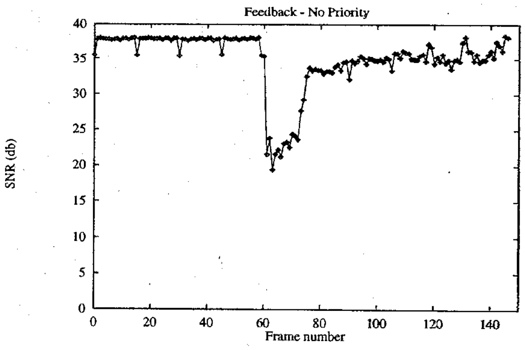


Fig. 9. SNR plots for bicycle sequence with *feedback control* for experiment described in Section IV.B.1.

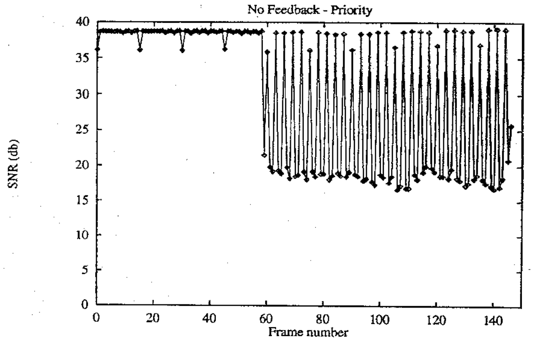


Fig. 10. SNR plots for bicycle sequence with *priority loss* for experiment described in Section IV.B.1.

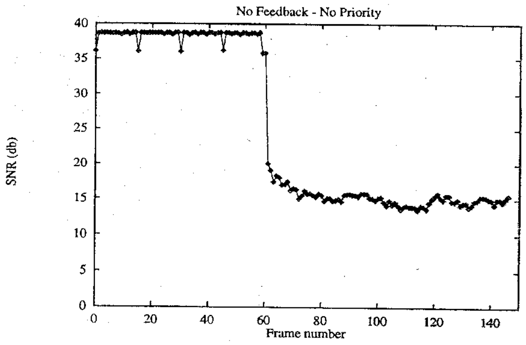


Fig. 11. SNR plots for bicycle sequence with *random loss* for experiment described in Section IV.B.1.

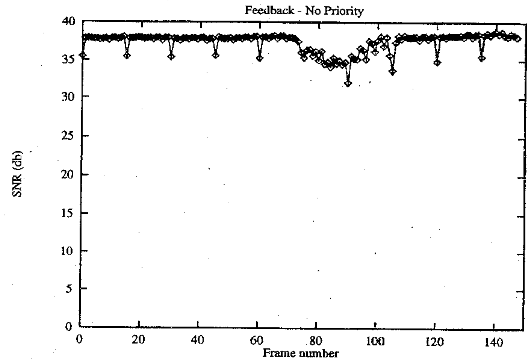


Fig. 12. SNR plots for bicycle sequence with *feedback control* for experiment described in Section IV.C.

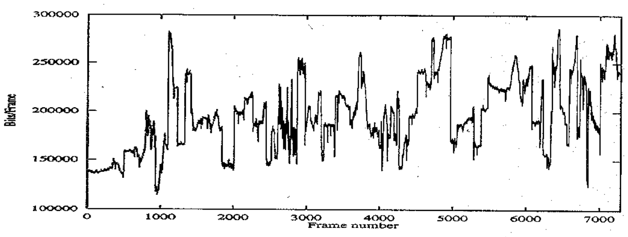


Fig. 13. Input frame rates for a long sequence with many scene changes.

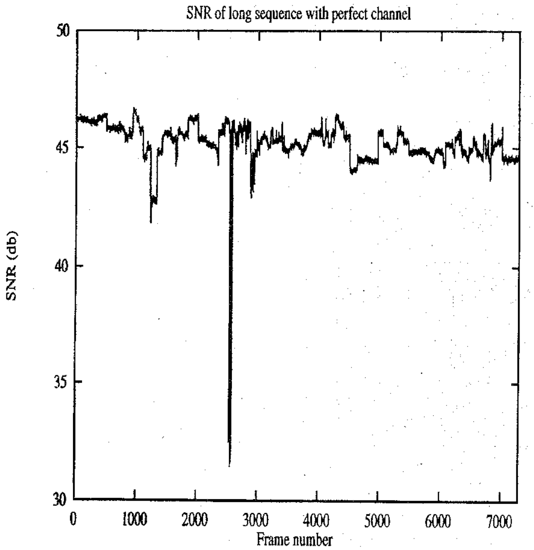


Fig. 14. SNR plots for long sequence with *perfect channel* for experiment described in Section IV.D.

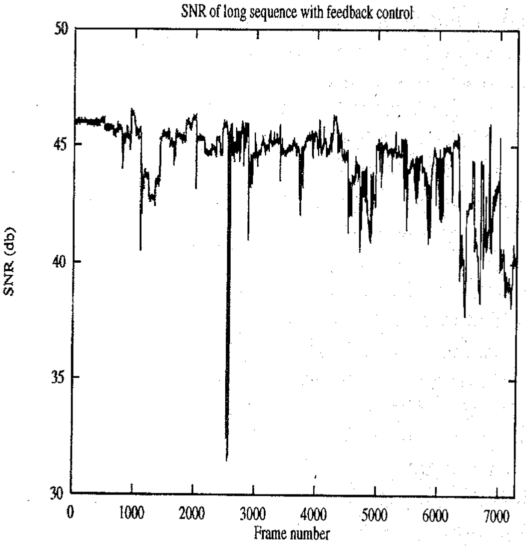


Fig. 15. SNR plots for long sequence with *feedback control* for experiment described in Section IV.D.

which we call loss persistence (Fig. 9), implies that measures such as bit error rates or packet loss probabilities are not good indicators of the perceptual quality. The link utilization at the bottleneck, shown in Fig. 7, remains high throughout the time period.

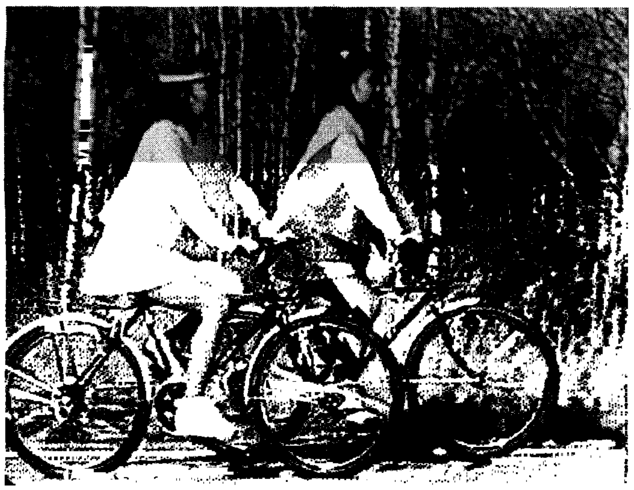


Fig. 16. Frame #110 for bicycle sequence with *perfect channel.*

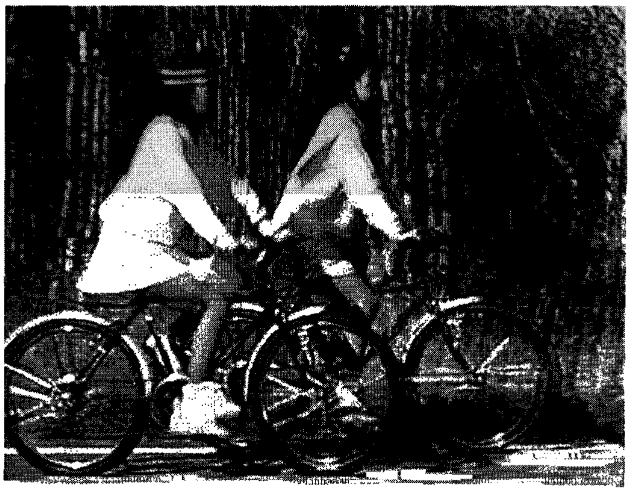


Fig. 18. Frame #110 for bicycle sequence with *priority loss.*

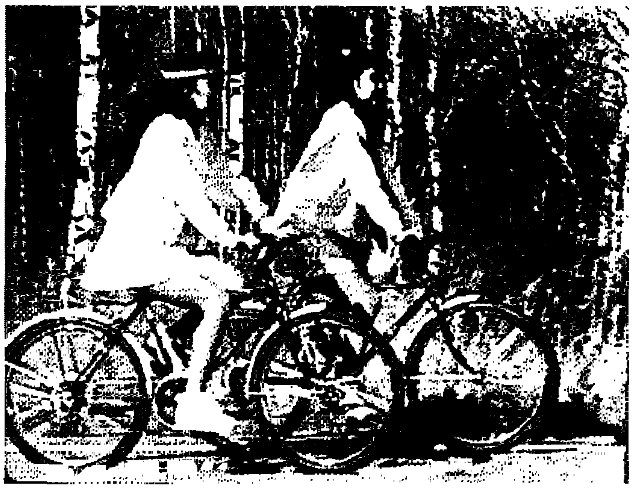


Fig. 17. Frame #110 for bicycle sequence with *feedback control.*

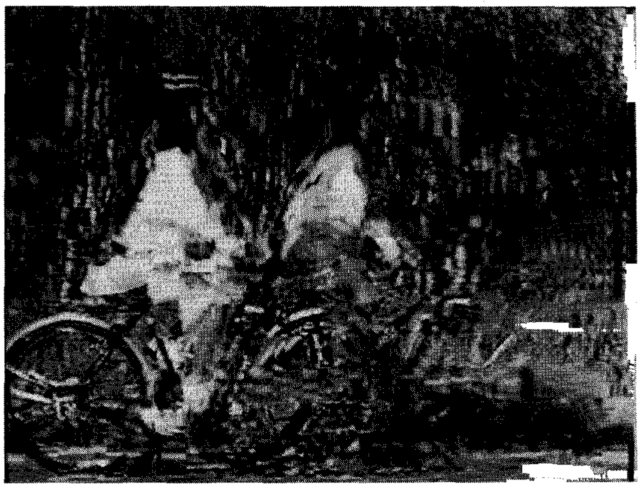


Fig. 19. Frame #110 for bicycle sequence with *random loss.*

Table I shows the average SNR values with the feedback based scheme are much closer to those for a perfect channel than for either of the other two schemes. A visual comparison of the decoded sequences corroborates this observation. In general, the perceptual quality observed with the feedback scheme was much better than that observed with the priority loss scheme. Figs. 16–19 show the picture quality for the *bicycles* sequence for frame #110 as decoded at the receiver for all four schemes. This B-frame is received approximately 2 s after the bandwidth reduction has taken place. These photographs are a fair snapshot of the differences in the perceptual quality observed when the complete video sequence was played back at the receiver. As expected, the perceptual quality observed with the random loss scheme was the worst with the visual effect of the bandwidth reduction similar to watching the sequence through a waterfall.

The visual comparison also confirmed the average SNR is an imperfect measure of the perceptual quality. For example, although the difference in the average SNR value between the perfect channel and the feedback scheme is about the same as that between the feedback scheme and the priority scheme, there was hardly any perceptible difference between the first two sequences but a huge difference between the latter two. Thus with feedback control, after the reduction in bandwidth, there is a barely noticeable flicker (due to the packet losses) followed by near perfect quality while for the priority loss case there is a highly visible flashing and blurring in the picture quality. The variation in the per frame SNR values, shown in Fig. 9, seems to be a somewhat better indicator of perceptual quality, although as we have pointed out, even a degradation in SNR for as long as 15 frames is barely perceptible.

Table I also indicates the feedback control scheme does not achieve an equal reduction in the quality of pictures transmitted for users sharing the bottleneck resources. Although all video sources take the same path and are controlled identically, the average SNR values are different.

2) Small Buffers and Out-of-Phase Sources: In the second group of experiments, there are only 100 packet buffers of size 500 bytes shared among all the active connections. The other experimental parameters are kept the same as in the

TABLE II
COMPARISON OF AVERAGE SNR FOR BICYCLES FOR EXPERIMENTS DESCRIBED IN SECTIONS IV.B.1, IV.B.2, AND IV.B.3

Group	Perfect	Feedback	Priority	Random
	Channel	Control	Loss	Loss
Large buffers and out-of-phase sources	38.63	34.99	30.46	24.86
Small buffers and out-of-phase sources	38.63	34.08	30.29	24.69
Large buffers and in-phase sources	38.63	36.10	27.64	24.94

experiments of Section IV.A.1. At a speed of 50 Mbps and a round-trip time of 42 ms, one hundred buffers (of size 500 bytes) can store data equal to one fifth the round-trip delay-bandwidth product. Due to the smaller number of buffers in this scenario, a greater number of packets are lost when the bandwidth reduces abruptly. As a result, the degradation in signal quality is slightly worse than when a greater number of buffers are available. The average SNR values observed for the *bicycle* sequence for this scenario are shown in Table II. The table also shows the SNR values observed for the other groups. The average link utilization, not shown here due to space constraints, is slightly lower than in the previous group.

3) Large Buffers and In-Phase Sources: In the third group of experiments, there are 400 buffers available per output queue but the video connections start up at the same time. As a result, the I frames of all the video sources will arrive *in-phase*. This experiment is motivated by previous studies that show that synchronization among traffic sources can produce a significant degradation in the performance and potentially cause unstable behavior [5], [7]. No such effects were observed in our experiments. The phase synchronization was observed to cause some oscillations in the buffer occupancy; however, the effect was not significant enough to affect the overall utilization or packet losses and thereby the perceptual quality.

C. Gradual Increase/Decrease in Traffic Intensity

In this set of experiments, the congestion is induced by gradually opening new video connections. The network topology and connection characteristics are similar to that of the experiments described in Section IV.B.1. However, instead of downsizing the available link bandwidth, five new video connections (controlled using the same feedback mechanism) are added at intervals of 200 ms each. Each of these connections transfers 33 frames so each is active for approximately one second. The aggregate bandwidth of these sources, without any feedback control, is 20 Mb/s. Thus the net effect from the point of view of Connections one to eight is the available bandwidth decreases gradually from 50 Mb/s to approximately 30 Mb/s then back to 50 Mb/s. All the new traffic connections are added at host H3 and traverse the congested link, exiting at host H4. We observe the SNR values for seven video connections.

The results shown in Table III show the gradual increase and decrease of the offered load at the bottleneck point have much less impact on the picture quality as measured by the SNR when compared to the case of the sudden traffic changes described in the previous section. No packets are lost since the adaptive controls have enough time to adjust the encoder rate to a new connection before the next change occurs. When the five connections terminate, the existing connections quickly adapt to the increase in bandwidth. This is clearly demonstrated in Fig. 12 which shows the SNR of the *bicycle* sequence degrading around Frame 65 as the new connections come on and increasing back to the original value around Frame 100.

TABLE III
COMPARISON OF THE AVERAGE SNR FOR GRADUAL CHANGES IN TRAFFIC

Stream	Perfect Channel	Feedback Control
autumn	38.11	37.92
ball of wool	40.10	40.02
bicycles	38.63	37.38
birches	38.08	37.48
cheer	38.44	36.65
ferris wheel	40.13	39.91
horses	39.52	39.52

D. Effect of Scene Changes

In this set of experiments, the congestion is caused by the statistical fluctuations in the load imposed by a fixed set of video connections. In general, if video connections remain active for long time periods and the admission control policy ensures new connections are denied at high loads, then the only possible source of congestion will be due to statistical fluctuations in the offered load of video connections. These variations occur over both a short-term and long-term time frame. Short-term fluctuations are observed due to the nature of the coding algorithm as well as the motion observed in any sequence of video frames. The long-term variation occurs due to the subject matter of video frames changing over time. Sometimes the change is abrupt, such as, when a scene change occurs. We intuitively expect scene changes to be the major cause of congestion, especially when the network is using admission control to regulate the number of active video connections.

Fig. 13 shows the frame rate distribution of the source encoded with MPEG–1 scheme. This sequence is relatively smooth with a peak to average ratio of 2:1 and an average rate of approximately 5.9 Mb/s. The video, a montage of outdoor scenes of wildlife, has approximately 7300 frames with about 12 scene changes. This video sequence is used to generate eight video sources. As before, the simulation has eight video sources traversing the main path from nodes H1 to H4. The bottleneck capacity of 50 Mb/s is approximately equal to eight times the average rate of the source. To speed up the simulations, however, we use a simpler configuration in which the propagation delay is kept the same but the intermediate switches S1, S2 and S3 are removed. We use two traffic scenarios. In one, each source comes on at the same time – this corresponds to the worst possible combination of sources from the point of view of statistical multiplexing gain. In the second, each source switches on at intervals of 200 frames starting at time zero. At 200 frames, the autocorrelation coefficient observed for this video sequence is less than 0.1, hence, we can assume the "new" sources generated are independent. As a point of comparison we repeat these experiments with no feedback control used during the transmission, i.e., the *random loss* case. In all cases, the duration of the simulation is 250 s and approximately three million packets are handled by the bottleneck link.

TABLE IV
COMPARISON OF UTILIZATION AND LOSSES

Set	Random loss		Feedback control	
	total losses	link utilization	total losses	link utilization
In phase	151546	0.869	1297	0.849
Staggered	28481	0.833	0	0.829

For these experiments, Table IV summarizes the results in terms of the number of packets lost and the average utilization. Clearly, the feedback control scheme performs very well with virtually no losses and only marginally lower utilization than the maximum possible value (obtained when no control is used). Figs. 14 and 15 show the SNR of the sequence over the duration of the experiment for one of the connections, with feedback control compared to the perfect case, when the start time of the sources are staggered apart. The sudden drop of 15 db observed around frame 2500 is due to an imperfection in the coding scheme and not a result of dropped packets due to congestion. The average SNR values for each of the seven sources is within about 1–2 dB of the perfect SNR. These results show the proposed feedback scheme is capable of dealing quite well with congestion caused by scene changes.

E. When Will Feedback be Too Late to be Useful?

The last set of experiments is directed at addressing the question of the limitations of network-generated feedback because of the inherent propagation delays. In other words, when is feedback too late to be useful? The autocorrelation observed in a video stream is one factor that indicates how much delay in the feedback path is acceptable. An additional factor limiting the maximum tolerable delay is the amount of available buffering at switches. Moreover, even if there is unlimited buffering available, the maximum end-to-end delay requirements of real time video traffic provides a natural upper bound. The effect of increasing delay on the performance of the feedback control scheme is the focus of this set of experiments. The simulation is run with propagation delays on the bottleneck link ranging from 20–200 ms, so the round trip propagation delays varied between 42 ms and 402 ms. The playback delay time at the receiver is chosen to be the sum of the propagation delay from source to receiver plus a *constant* delay of 80 ms. In all the earlier set of experiments the value of playback delay time was 100 ms, so the maximum tolerable variation in the queueing delay (80 ms) is the same in all the experiments. The configuration chosen is the same as that used in the experiment of Section IV.D. All the sources are chosen in phase in order to stress the feedback control mechanism as much as possible.

Table V shows the average utilization and number of packet losses for each of the scenarios. The degradation in performance is quite gradual with an increase in the propagation delay causing a greater number of losses with slightly more than on percent packet loss at a round trip propagation delay of 400 ms. However, even with a propagation delay of 200 ms the number of packet losses is less than 25% of the loss observed when no control is exercised (see Table IV). Note that we kept all the controlled parameters at the same values throughout the experiments, including the number of available buffers. In real life, if we knew the ratio of the available buffering to the round trip delay was relatively small, one would choose to have less damping in the control equation, leading to somewhat fewer losses at the higher propagation delays. In general, the extent of the degradation observed will depend on the exact statistical multiplexing characteristics and the choice of parameter values for the control equation.

TABLE V
COMPARISON OF UTILIZATION AND LOSSES FOR VARIOUS VALUES OF PROPAGATION DELAY

Round trip propagation delay	Total losses	Link utilization
42 ms	1297	0.849
162 ms	10245	0.829
282 ms	26538	0.828
402 ms	36132	0.827

V. CONCLUSIONS

In this paper, we have proposed a scheme in which explicit feedback information from the network is used to control the data generation rate of a video source. The robustness of the feedback control and the effectiveness in achieving graceful degradation was tested with different models for how congestion occurs. Although this feedback information was available only after a significant delay, our results show the perceptual quality degraded gracefully. The transition was hardly noticeable from the perceptual quality of the pictures even when each source had to reduce its sending rate by as much as 30%.

One of the compelling reasons to use feedback control for packet video transport in future integrated service networks, such as ATM networks, is that the scheme allows use of a simply implemented, and yet quite aggressive admission control policy. With feedback, admission control can make resource reservations for the lowest acceptable quality of service for video traffic. This is much simpler than having to make a-priori assumptions about the average or peak bandwidth that a video source would have. More video connections could be supported by the relaxed admission control scheme, feasible with the feedback scheme. The feedback control mechanism allows all connections to still be carried even when overload occurs, albeit at somewhat lower picture quality. In this sense, the feedback scheme should be compared more with those schemes where similar reduction in picture quality is achieved, either by reducing frame rates or simply losing data in the network followed by data recovery methods implemented at a receiver. We plan to pursue this comparison in our future efforts.

Ensuring fairness in servicing VBR video traffic appears to be a difficult problem. In the proposed scheme, the available bandwidth is not reduced equally among all video streams multiplexed through the bottleneck. What is appropriate is reducing the available bandwidth to each connection proportional to their current quality, with more bandwidth to a video source and a more complex sequence of images. This issue has been investigated further in [16], [22] and is an area of continuing work.

References

[1] R. Blake, personal communication.

[2] E. W. Biersack, "Performance evaluation of forward error correction in ATM networks," in *Proc. ACM SIGCOMM*, 1989, pp. 248–257.

[3] D. Clark, S. Shenker, and L. Zhang, "Supporting real time applications in an integrated services packet network: architecture and mechanism," in *Proc. ACM SIGCOMM*, 1992, pp. 14–27.

[4] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control.* Englewood Cliffs, NJ: Prentice–Hall, 1984.

[5] A. Erramilli and L. J. Forys, "Traffic synchronization effects in teletraffic systems," in *Teletraffic and Datatraffic in a Period of Change*, A. Jensen and V. B. Iversen, Eds. Amsterdam, The Netherlands: North–Holland, 1991, pp. 201–206.

[6] D. Ferrari and D. Verma, "A scheme for real time channel establishment in wide-area networks," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 368-379, Apr. 1990.

[7] S. Floyd and V. Jacobsen, "The synchronization of periodic routing messages," in *Proc. of ACM SIGCOMM*, 1993.

[8] D. Le Gall, "MPEG: a video compression standard for multimedia applications," *Commun. ACM*, pp. 47–58, Apr. 1991.

[9] M. Gilge and R. Gussela, "Motion video coding for packet-switching networks—an integrated approach," in *Proc. SPIE*, 1991.

[10] S. J. Golestani, "A framing strategy for congestion management," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1064–1077, Sept. 1991.

[11] D. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Trans. Circuits Syst.*, vol. 2, no. 1, pp. 49–59, Mar. 1992.

[12] C. R. Kalmanek, H. Kanakia, and S. Keshav, "Rate-controlled servers for very high-speed networks," in *Proc. GLOBECOM*, 1990, pp. 12–20.

[13] D. Lee, B. Melamed, A. R. Reibman, and B. Sengupta, "TES modeling for analysis of a video multiplexer," *Performance Evaluation*, pp. 21–34, 1992.

[14] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Trans. Commun.*, pp. 834–844, July 1988.

[15] P. P. Mishra and H. Kanakia, "A hop-by-hop rate-based congestion control scheme," in *Proc. ACM SIGCOMM*, 1992, pp. 112–123.

[16] P. P. Mishra, "Fair bandwidth sharing for video traffic sources using distributed feedback control," in *Proc. IEEE GLOBECOM*, 1995.

[17] S. P. Morgan and A. R. Reibman, "Statistical multiplexing comparison of one and two-layer video codecs for teleconferencing," in *Proc. 5th Int. Workshop Packet Video*, 1993.

[18] P. Pancha and M. El-Zarki, "MPEG coding for variable-bit-rate video transmission," *IEEE Commun. Mag.*, May 1994.

[19] P. Pancha and M. El–Zarki, "Prioritized transmission of variable bit rate mpeg video," in *Proc. GLOBECOM*, 1992, pp. 1135–1139.

[20] A. R. Reibman and A. Berger, "Traffic descriptors for VBR video teleconferencing over atm networks," in *Proc. GLOBECOM*, 1992, pp. 314–319.

[21] D. Reininger, B. Melamed, and D. Raychaudhuri, "Variable bitrate MPEG video: characteristics, modeling and multiplexing," in *Proc. Int. Teletraffic Congress*, 1994, pp 314–319.

[22] D. Reininger and W. Kwok, "Rate control for VBR MPEG video on local area networks," in *Proc. SPIE High Speed Networking Multimedia Computing*, 1994, pp 153–161.

[23] E. Rathgeb, "Modeling and performance comparison of policing mechanisms for ATM networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 325–334, Apr. 1991.

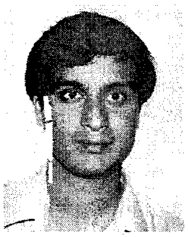
[24] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: a new resource reservation protocol," *IEEE Network Mag.*, Sept. 1993.

[25] R. Yavatkar and L. Manoj, "Optimistic strategies for large scale dissemination of multimedia information," in *Proc. ACM Multimedia*, 1993, pp. 13–20.

[26] XTP Protocol Definition Revision 3.6. Protocol Engines Incorporated, PEI 92-10, Mountain View, CA, 1992.

Hemant Kanakia received the B.Tech. degree from the Indian Institute of Technology, Bombay, in 1975, and the Ph.D. degree from the Stanford University, Palo Alto, CA, in 1990.

He is currently a Member of the Technical Staff in the Computing Science Research Center at AT&T Bell Laboratories, in Murray Hill, NJ. His research interests are in the design of integrated service networks, packet video services, and packet switching systems.



Partho P. Mishra received the B.Tech. degree from the Indian Institute of Technology, Kharagpur, in 1988, and the M.S. and Ph.D. degrees from the University of Maryland, in 1991 and 1993, all in computer science.

He is currently a Member of the Technical Staff in the Networked Computing Research Department at AT&T Bell Laboratories, in Murray Hill, NJ. His research interests are in the design and analysis of integrated service networks.

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

From 1988 to 1991, she was an assistant professor in the Department of Electrical Engineering at Princeton University. She is currently a Distinguished Member of the Technical Staff in the Visual Communcations Research Department at AT&T Bell Laboratories, in Murray Hill, NJ. She was the Technical Program Chair for the Sixth International Workshop on Packet Video in Portland Oregon in 1994, and she is currently an Associate Editor for IEEE Transactions on Image Processing. Her research interests include video compression, packet video, and wireless video.

Dr. Reibman is a member of Sigma Xi, Eta Kappa Nu, and Tau Beta Pi.