End-to-end Delay of Videoconferencing over Packet Switched Networks

Mario Baldi
Politecnico di Torino, Dipartimento di Automatica e Informatica
Yoram Ofek
IBM, T. J. Watson Research Center

Abstract— **Videoconferencing is an important global application - it enables people around the globe to interact when they are far from one another. In order for the participants in a videoconference call to interact naturally, the end-to-end delay should be below human perception - about 100 ms. Since the global propagation delay can be about 100 ms, the actual end-to-end delay budget available to the system designer (excluding propagation delay) can be no more than 10 ms. We identify the components of the end-to-end delay in various configurations with the objective of understanding how it can be kept below the desired 10 ms bound.**

We analyze these components going step-by-step through six system configurations obtained by combining three generic network architectures with two video encoding schemes. We study the transmission of raw video and variable bit rate (VBR) MPEG video encoding over (*i*) circuit switching, (*ii*) synchronous packet switching, and (*iii*) asynchronous packet switching. In addition, we show that constant bit rate (CBR) MPEG encoding delivers unacceptable delay, which is on the order of the group of pictures (GOP) time interval.

This study shows that having a global common time reference, together with time-driven priority (TDP) and VBR MPEG video encoding, provides adequate end-to-end delay, which is (*i*) below 10 ms, (*ii*) independent of the network instant load, and (*iii*) independent of the connection rate. The resulting end-to-end delay (excluding propagation delay) can be smaller than the video frame period, which is better than what can be obtained with circuit switching.

Keywords— **Time-driven Priority, MPEG, Videoconference, Quality of Service End-to-end Delay, Performance Guarantees.**

I. Introduction

Interactive real-time applications over packet switched networks are challenging. One of the key problems is keeping the end-to-end delay below human perception. Various studies concluded that for natural hearing this delay should be below 100 ms [1]. The problem that we study in this work is videoconferencing in which voice and video should be synchronized (a.k.a. lip-sync), thus, the end-to-end delay of the video should be below 100 ms as well. Since the global propagation delay can be about 100 ms the actual end-to-end delay budget available to the system designer (excluding propagation delay) can be **no more than 10 ms**.

The video stream requires high capacity and since network resources are limited, the video pictures should be compressed. Compression can be costly in terms of end-to-end delay. In this work we assumed MPEG encoding [2], since it is one of the most popular compression techniques. Other encoding techniques such as Moving JPEG [3], [4] and H.261 [5] are possible, but they are not in the scope of this paper and are left for further research. Here we show that using MPEG in its "standard" way, at constant bit rate (CBR), introduces unacceptable delay, as discussed in Section III-B. From the delay viewpoint, Moving JPEG is a better choice for CBR encoding, even though it generates a higher bit rate.

Intuitively, for small end-to-end delay, a picture (video frame) is captured, compressed, and sent once there are enough data units to send them in packets over the network. This approach can result in a short end-to-end delay. However, it raises some key questions: (*i*) *when* the compressed video data units will be ready to be sent (in general, the data units generation during compression is difficult to predict), and (*ii*) *how many* data units will be generated after compression of each picture. Both pieces of information are needed in order to reserve communication resources inside the network. In other words, the difficulty arises since the time data units are produced and the amount of data units produced may change from picture to picture. In this paper, we discuss the two problems in details and suggest some solutions which provide global quality of service (QoS) guarantees.

The heart of these solutions is the way in which packet queuing and forwarding is managed by nodes. Some solutions are based on nodes having a common time reference obtained from the GPS (global positioning system) [6] and using it to control packet forwarding; this allows queuing to be reduced in the network nodes. Other approaches are based on asynchronous packet forwarding and require special techniques, like weighted fair queuing (WFQ) [7], [8], [9], to be deployed in the management of queues inside the network nodes. These queue management schemes guarantee a bound on the end-to-end delay which is: (*i*) **inversely proportional** to the bandwidth allocated to each connection or session, and (*ii*) **proportional** to the packet length (in IP networks packets are rather long). This is particularly a problem for a low bit rate service, since a short delay bound can be provided only at the expense of an over-allocation of the bandwidth.

Designing the network for stringent delay requirements can be beneficial to other system aspects. One of the main advantages of *small delay jitter are small buffers* inside the network and at the receiver side. Networks with high speed links require buffers with short access time, which can be expensive, and therefore, small delay jitter will save money.

In the future, when *virtual reality* applications will become real (rather than virtual) the system constraints, such as delay and loss, will be even more rigid. Therefore, in the coming years, the competition among various network vendors will be more than just the capability to provide the service: it will become primarily a competition to provide better quality of network services. Some of the techniques discussed in this manuscript will indeed provide better quality for a lower price (because of less buffers requirements, for example).

II. The Model

In this section we identify the components of the end-to-end delay of a videoconferencing system for a number of relevant system configurations. In Section III, we analyze each delay

Visiting IBM, T. J. Watson Research Center while performing this work.

0-7803-4383-2/98/$10.00 © 1998 IEEE.

component for the various configurations. The model focuses on video rather than audio because (1) compression (not required by the latter due to its lower bandwidth requirements) and (2) the sampling rate, which is much lower for video (5-30Hz vs. 8KHz.), can be key factors in determining the and-to-end delay.

A. High-level Delay Components

Figure 1 shows the model of a videoconferencing system. The end-to-end delay of the system is the time elapsed from when a video image is captured by the video camera at the sender side until when it is displayed on the monitor at the receiver (upper arrow in Figure 1). In order for the participants in the videoconference call to be able to interact naturally we have the following objective:

Objective 1: The end-to-end delay (including propagation delay) should be below 100 ms.

For delivering high visual quality, video frames should be displayed on the receiver's monitor at the same fixed pace they have been captured. This leads to the second objective (see Figure 1):

Objective 2: **Continuous play.** The receiver displays pictures (plays audio samples) continuously and at the same rate they have been captured by the sender.

Objective 2 means that the end-to-end delay between capture and display is constant.

The end-to-end delay is modeled with four high-level components, whose values depend on the system configuration. The segmented arrow in Figure 1 shows which function in the system introduces each delay component.

1. A *processing* delay (**P** in Figure 1) is introduced on both the sender and receiver sides. It may encompass, for example, the time spent in compressing and decompressing of pictures.
2. The *network* delay (**N** in Figure 1) is the time needed to move data units from a source to the other videoconference participant(s). The network delay includes also the protocol processing in both sender and receiver(s).

The above two delay components can vary during a videoconference call. In order to meet Objective 2 (i.e., constant end-to-end delay) these variations should be "smoothed out" before pictures are displayed at the receiver. We identify two *resynchronization* delay components, which are typically realized by some sort of *replay buffer*. All pictures when exiting this buffer have experienced the same delay from the time they were captured.

3. The *processing resynchronization* delay (**PR** in Figure 1) cancels the delay variations in generating the compressed video data units.
4. The *network resynchronization* delay (**NR** in Figure 1) cancels the variations of the delay experienced in the network (e.g., the delay jitter due to queuing in network nodes).

Thus, Objectives 1 and 2 can be summarized using the four delay components in the following way: after resynchronization the end-to-end delay, including propagation delay - Pr, is:

$$\mathbf{P} + \mathbf{N} + \mathbf{PR} + \mathbf{NR} + Pr = \mathbf{CONSTANT} \leq \mathbf{100\ ms}.$$

B. The Analysis Methodology

In this work we analyze the end-to-end delay by going through a number of configurations, each adding one or more components to the end-to-end delay. The system configurations differ for the network architecture and the video encoding technique exploited. We consider three network architectures and three video coding techniques; since one of the latter is proven not to be suited for videoconferencing, we end up with six system configurations which are studied in Section III. The network architectures are described in Section II-C. The three video encoding schemes are described below.

Raw video: the three color components of each picture are digitally sampled and the resulting data units are transmitted over the network with no delay. Since the capture card provides all the bits within few milliseconds and they are sent immediately, the traffic generated is bursty.

Constant Bit Rate (CBR) MPEG: pictures are encoded according to the The Motion Picture Expert Group (MPEG) [2], [4], [10] standard. Compression is obtained by eliminating spatial (within each picture) and temporal (between subsequent pictures) redundancy. The amount of bits needed to encode each picture (a.k.a. *picture size*) is not known in advance and is highly variable. Since the picture rate is constant, bits are produced at a variable rate. A buffer is used to smooth the production bit rate: bits exit this buffer (and enter the network) at a constant rate. The encoder is controlled according to the fill level of the buffer in order to prevent it from either overflowing or underflowing. The delay introduced by the buffer renders CBR MPEG encoders not suited for videoconferencing, as it is discussed in details in Section III-B.

Variable Bit Rate (VBR) MPEG: only a small buffer is exploited at the output of the encoder for assembling data units which may exit the encoder shortly after they are produced. The rate of the resulting compressed stream is highly variable.

C. The Network Architectures

Three network architectures are considered. One is circuit switching which is a fully synchronous network: routing and flow control are done using time. The second is asynchronous packet switching with no notion of global common time reference. And the third one is a combination of the previous two, called time-driven priority (TDP); it has a global common time reference that is used only for flow control and not for routing.

C.1 Circuit Switching

A fixed amount of link capacity is assigned to each videoconference call by means of time division multiplexing. At the time a data unit (e.g., a byte) is transmitted from its source, it is possible to predict deterministically when it will exit any switch along its route. The time resolution of this advanced knowledge is *much shorter than the data unit transmission time.* As a result, network nodes introduce a small (few microseconds) delay.

C.2 Asynchronous Packet Switching

In packet switched networks data are gathered in packets which are sent to an ingress switch or router. Switches forward packets toward the destination while statistically multiplexing packets from different sources which are forwarded over the same link. When a packet happens to have to be forwarded on a busy link, it is delayed until the link is available. This delay is called (network) queuing delay.

Queuing delay has high variability since the time spent in an output buffer depends on the packets already in this buffer and in other buffers of the same output port. Thus, the distribution of the queuing delay experienced by packets throughout the videoconference call is determined by the resource allocation policy, the scheduling algorithms used for buffer management, and the overall network traffic characteristics.

Queuing delay often counts for a large portion of the network delay (see Section III-A.3 for details on network delay compo-

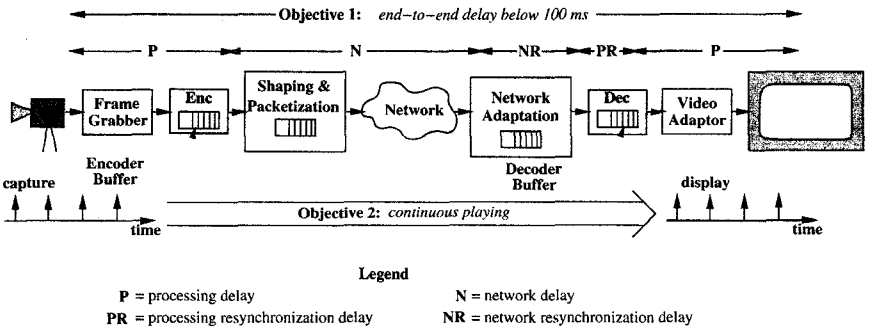

Fig. 1. Model of a Videoconferencing System.

nents) and its high variability gives a major contribution to the network delay variation, a.k.a. *network jitter*. The maximum jitter is defined, in the context of this work, as the difference between maximum and minimum delay. The network resynchronization delay should be between zero (for packets having experienced maximum network delay) and the maximum jitter (for packets having experienced minimum network delay).

Actually, the replay buffer introduces an *excess (network) resynchronization* delay up to the maximum network jitter (see [11] for a detailed explanation). As a result, the network delay possibly contributes with its maximum value plus its maximum jitter.

C.3 Time-driven Priority

Time-driven priority (TDP) [12], [13] gives higher priority to real-time traffic in a *periodic fashion* in order to provide the following properties for real-time traffic: (*i*) bounded delay - independent of the best-effort data traffic, (*ii*) constant bound on the jitter, which is independent of the network size, and (*iii*) either *deterministic* no-loss or *probabilistic* control of the loss due to congestion inside the network. For example, for real-time service it is possible to ensure *deterministically no (loss due to) congestion inside the network*. Moreover, this can be achieved under a full link utilization and without adversely affecting the quality of service. TDP is a multiplexing scheme aimed at sharing link capacity while guaranteeing users against uncontrolled delays (or even losses) due to contention in accessing network's links.

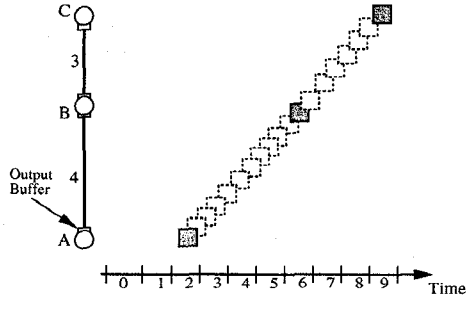

Fig. 2. TDP Packet Forwarding.

The time is divided into *time frames* (TFs) of fixed duration T_f (a typical choice is $T_f = 125\,\mu s$). Given the link capacity C, in each TF a fixed amount of bits $T_f \cdot C$ can be sent on a link. Assuming small propagation delay, a real-time packet is forwarded one hop every TF. Since packets are not buffered for uncontrolled time, in the more general case with arbitrary propagation delay, each packet takes a fixed number of TFs to move from the output buffer of an intermediate node to the output buffer of the following one on the path to the destination, as shown in Figure 2. This is also called *RISC-like forwarding*.

III. End-to-end Delay Analysis

This section presents the end-to-end delay of the six system configurations following the delay model presented in Figure 1. For clarity, two of the delay elements shown in Figure 1 are not included in the analysis, since they are the same in all configurations. They are briefly described in the following.

Capture delay. The frame grabber introduces a constant delay on the order of few milliseconds.

Presentation delay. As a picture is ready to be displayed on the receiver side, it is inserted into the video frame buffer which is periodically scanned by the video adaptor to trace the image on the screen. This introduces a presentation delay which, depending on the video refreshing frequency, can be up to 17 ms. The presentation delay can be eliminated by synchronizing the decoder, the video controller inside the receiver, and the capture card, as shown in Figure 1. This requires synchronization between network and decoder, receiver network interface and sender network interface, network and encoder, encoder and capture card. This end-to-end synchronization, otherwise very hard to implement, can be easily obtained by using a common time reference, e.g., from the GPS [6].

A. Raw Video

The delay analysis of raw video is interesting since it concerns a reduced set of delay components, which are the network (**N**) and the network resynchronization (**NR**) delays. There are only two delay components since there is no compression, and therefore, the processing (**P**) and processing resynchronization (**PR**) delays are null[1].

A.1 Circuit Switching

In circuit switched networks it is assumed that the video transmission is continuously using the bandwidth, B^{CS}, allocated to this circuit. Figure 3 shows, for each frame, the resulting timing (and the rate) of the production of bits by the capture card and of their transmission over the network. This introduces a *network shaping* delay

$$S^{CS}_{Raw} = \frac{F_r}{B^{CS}},$$

where F_r is the picture size in bits. The end-to-end delay is constituted by a single component (the network delay **N** depicted in Figure 1) given by

$$\Delta^{CS}_{Raw} = S^{CS}_{Raw} + Pr + Sw, \qquad (1)$$

[1] Analog-to-digital and digital-to-analog conversions of pictures requires few milliseconds which is a short time in comparison to the other delay components.

where Pr is the propagation delay, Sw is the circuit switching delay (typically, a few microseconds). The minimum circuit bandwidth required for the transmission of raw video is,

$$B^{CS} \geq \frac{F_r}{T}.$$

The end-to-end delay can be reduced by allocating larger bandwidth to the circuit. In this case, the circuit is busy only for a time S^{CS}_{Raw} in each video frame period T. As a result, the remaining time $T - S^{CS}_{Raw}$ is wasted and no other connection can exploit the reserved resource left unused.

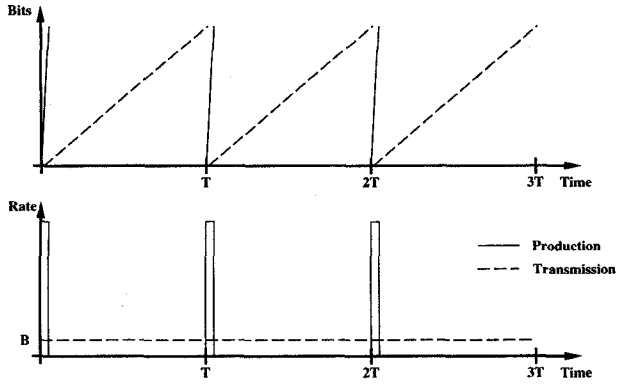

Fig. 3. Raw Video Encoding and Transmission over a Circuit Switched Connection.

If minimum bandwidth is to be allocated for the videoconference call, the end-to-end delay is larger than one video frame period. Therefore, the lower the video frame rate, the larger the end-to-end delay. For example, the minimum bandwidth required to send QCIF pictures at 15 frames per second, is 4.5 Mb/s and the resulting shaping delay is 67 ms. However, if more bandwidth is allocated to decrease the network shaping delay to 30 ms, more than 50% of the allocated bandwidth is wasted because the circuit is idle for half of the video frame period.

A.2 Time-driven Priority

Raw video can be sent over a packet switched network with TDP by inserting each picture into one or more packets which are transmitted during a TF[2]. During the TFs between the transmission of two subsequent pictures of the same session, both real-time and best-effort traffic of other connections can be transmitted, see Figure 4.

The network delay **N** is the only component of the end-to-end delay. It can be expressed as $L \cdot T_f$ (where L is a function of the number of nodes and the processing delay inside each node) and the propagation delay - Pr. The end-to-end delay is:

$$\Delta^{TDP}_{Raw} = L \cdot T_f + Pr. \quad (2)$$

In TDP the presentation delay is zero since the frame grabber and the video display adaptor are synchronized.

Resource reservation is based on the definition of a *time cycle* which encompasses a predefined number of TFs: all the nodes share the same knowledge of the ordinal position of the current TF inside the time cycle[3]. Bandwidth is allocated to a sender/receiver pair, by properly reserving (a fraction of) the link capacity during a number of TFs per time cycle on each link

[2]If the link capacity is not large enough to allow a picture to be transmitted in a single TF, it is sent over more time frames. This introduces a network shaping delay given by the number of TFs needed to transmit each picture.

[3]This can be easily implemented with GPS [6].

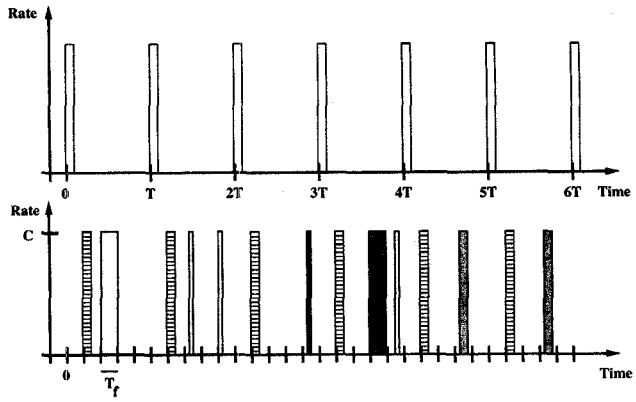

Fig. 4. Generation of Raw Pictures and Transmission over a Network with Time-driven Priority (TDP).

on the path from sender to receiver[4]. In order for intermediate nodes to perform the RISC-like forwarding, the TFs on a link must be chosen according to the TFs reserved on the upstream link, and the time needed for a packet to be transmitted from the output buffer of the upstream node to the output buffer of the considered node, see more details in [13].

A.3 Asynchronous Packet Switching

A picture, split over one or more packets, is sent through the network from the source to a packet switching node introducing a *transmission* delay F_r/C, where F_r is the picture size and C is the link bandwidth or capacity. In the network, packets experience a fixed propagation delay, Pr, and a variable queuing delay; all the delays mentioned so far are part of the network delay component **N**.

Due to the real-time requirements of the video stream, a replay buffer is needed at the receiver to compensate for the variation in the queuing delay. The compensation is obtained by delaying the samples that have experienced a queuing delay shorter than the maximum Q_M; the delay introduced is part of the network resynchronization delay **NR**. As a result of the compensation, the sum of the queuing delay and its compensation experienced by each sample is the maximum queuing delay Q_M.

Actually, the compensation of the queuing delay introduces also the *excess resynchronization* delay $E_r \in [0, \Delta Q]$ constant over the duration of the whole conference; ΔQ is the maximum variation of the queuing delay, i.e., the difference between the maximum and the minimum queuing delay. The excess resynchronization delay is introduced because when a packet is received the actual delay it experienced in the network is not know, as explained in detail in [11].

Thus, if sender and receiver do not share a common time reference, the end-to-end delay experienced by the user of the videoconferencing system is

$$\Delta^{Async}_{Raw} = \frac{F_r}{C} + Pr + Q_M + E_r.$$

The compensation delay introduced by the replay buffer and the related error constitute the network resynchronization delay component **NR** identified in Figure 1, while the other terms are part of the network delay component **N**. It can be worth pointing out that the contribution Q_M accounts partly for the **N** component (since it contains the queuing delay) and partly for

[4]The TFs during which capacity is allocated to a videoconference call are said to be reserved to the call.

the **NR** component (since it contains the compensation delay). The excess resynchronization delay E_r can be eliminated if the sender and the receiver have a common time reference.

The maximum queuing delay is usually much larger than all other network delay components; also, Q_M is much larger than the minimum queuing delay and, consequently, $\Delta Q \simeq Q_M$. Thus, due to jitter compensation, *the maximum queuing delay can contribute twice* (namely, by itself and as excess resynchronization delay) to the end-to-end delay [11]. Schemes, like WFQ and PGPS [7], [8], [9], are being proposed for preventing loss and bounding the queuing delay. Such schemes provide a bound that is inversely proportional to the bandwidth allocated to the session, and proportional to the packet size and the number of hops[5].

The contribution to the end-to-end delay due to the queuing delay and its compensation can be reduced by underdimensioning the replay buffer, thus having the queuing delay and the excess resynchronization delay contributing with a value $\hat{Q}_M < Q_M$. As a consequence, all the (parts of) pictures experiencing a network delay larger than $F_r/C + P + \hat{Q}_M + E_r$ are discarded at the receiver at the expenses of the visual quality of the received video stream. $\hat{Q}_M$ is some percentile of the queuing delay chosen as to guarantee that the percentage of discarded packets does not affect visual quality. Some videoconferencing applications explicitly designed for operation over asynchronous packet switched networks adapt the resynchronization delay introduced by the replay buffer to the instantaneous distribution of the network delay experienced by packets [14]. This results in variable visual quality and user perceived delays, and does not comply with Objective 2.

Losses in network nodes due to congestion and buffer overflow also decrease the perceived quality of a videoconference call. In order to reduce buffer overflows and control the distribution of queuing delay over the duration of videoconference calls, resources are reserved in the nodes along the connection path and access to the network is controlled. This limits the amount of guaranteed traffic routable over the same link. Since transmission of large amounts of data at link speed (*bursts*) makes queues grow suddenly, bursty sources require a large amount of resources to be allocated and significantly reduce the overall amount of real-time traffic the network can support. Source burstiness can be reduced through *traffic shaping* at the network boundaries. Traffic shaping mechanisms like, for example, the *leaky bucket* [15], guarantee an average bandwidth B to the source while keeping the burstiness below a predefined value[6]. This introduces a shaping delay

$$S_{Raw}^{Async} = \frac{F_r - A}{B}, \qquad (3)$$

where A is the largest burst size, i.e., the maximum number of bits which can be sent at the full link speed. On one hand, the traffic shaping at the boundary of the network reduces the buffer requirements in the nodes, the queuing delay in the network and its variability (i.e., both the **N** and **NR** components of the end-to-end delay model proposed in Figure 1); on the other hand it introduces a variable shaping delay that is compensated on the receiver side (i.e., contributes to both the **N** and **NR** components of the end-to-end delay model proposed in Figure 1). In summary, the end-to-end delay can be expressed as

$$\Delta_{Raw}^{Async-TS} = S_{Raw}^{Async} + \frac{P_s}{C} + Pr + \hat{Q}_M + E_r,$$

where P_s is the size of packets sent into the network.

B. Why VBR MPEG?

Compression, although reduces the transmission delay, can have large processing (**P**) and processing resynchronization (**PR**) delays. Moreover, if the naturally variable rate of a compressed stream is to be converted into a constant one, a significant contribution to the processing resynchronization delay must be further introduced, thus adversely affecting the end-to-end delay. The objective of the following discussion is to present and justify the rationale for recommending the use of VBR MPEG, rather than CBR MPEG, for videoconferencing applications. The motivation is that VBR encoding can keep very short the contribution to the end-to-end delay due to the processing resynchronization (**PR**) component[7].

The Motion Picture Expert Group (MPEG) [2], [4], [10] standard encodes video frames in one of two different ways[8]: *Intra-frame Coding* eliminates spatial redundancy inside pictures producing the so called *I-frames*; *Predictive Coding* eliminates temporal redundancy between a picture and the previous one and produces *P-frames*. P-frames are typically from 2 to 4 times smaller than an I-frame: the more similar two subsequent pictures are, the more temporal redundancy can be eliminated. If a scene is slow moving, subsequent pictures are similar and predictive coding delivers high compression, i.e., I-frames are much larger than P-frames.

A video sequence is compressed by encoding one picture out of N as an I-frame, and the remaining $N-1$ pictures as P-frames; the sequence of N pictures is called a *Group Of Pictures* (GOP). The larger N, the smaller the amount of bits needed to encode each video sequence. If the network introduces an error in an encoded I-frame, the error will propagate into the entire GOP. The next I-frame is the first picture not to be affected by such error. Thus, the larger N the more damage an error can have.

Due to the difference between I-frames and P-frames, the rate of the bit stream produced by the encoder has high variability. Figure 5 shows the amount of bits produced by the software MPEG encoder **dvdenc** [16] during the encoding of the "Cheerleaders" video sequence; an image from the sequence is shown in Figure 6.

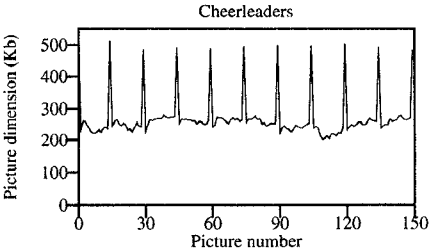

Fig. 5. Natural Dimension of MPEG Encoded Pictures from the "Cheerleaders" Sequence.

[5]Note that some of the results reported in [7] are based on several timing assumptions, such as, that the delay between nodes is zero or constant. Realizing such an assumption will require the synchronization of the local clocks of all the nodes, which is equivalent to the global common time reference used for TDP. However, the general result that is relevant to our work is Section X on *PGPS Network* (page 146 [7]). In particular, Equations 37, 38 and 39 on page 148, which have the following general structure:
$PGPS - Delay - bound - connection_i \leq \frac{2(K-1)L_i}{\rho_i}$, where L_i is the packet size, K is the number of hops and ρ_i is the rate of connection i.

[6]If a leaky bucket is exploited, B is its token generation rate and A is the token pool size.

[7]A detailed discussion can be found in [11].

[8]Actually, a third type of encoding, called bidirectional predictive coding, exists. Before a picture can be coded, a reference subsequent picture must be captured and coded. This introduces a delay of some frame periods that we consider not acceptable given the 100 ms end-to-end delay bound. Thus, this type of compression is not considered here.

A CBR MPEG stream is obtained by filling a buffer with the output of the basic encoding process and retrieving bits at a constant *target rate* B, as shown in Figure 7. This buffering process introduces a sensible variability of the processing delay component **P** of each picture which must be compensated in the decoder thus introducing a large processing resynchronization delay component **PR**. A *rate control* function (in principle) monitors the fullness of the encoder buffer and adjusts the quantization step-size to change the natural bit rate in order to prevent it from underflowing or overflowing. The rate control function tunes the bit production of the encoder to grant the stream compliance according to a model of a system target decoder buffer (see Figure 7) whose dimension is included in the MPEG stream [2]. The bit production can be modified acting on the step-size of a quantizer used at the last stage of the encoding process[9].

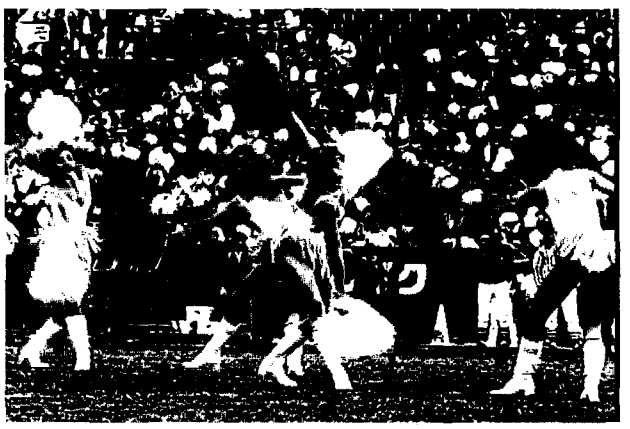

Fig. 6. Picture from the "Cheerleaders" Sequence.

The CBR encoder contributes to the processing delay with F/B, being F the size of an encoded picture. Since picture size is not constant, this contribution is variable and must be compensated at the receiver. The resulting overall contribution (after resynchronization) to the end-to-end delay is called *coding shaping delay* S_c:

$$S_c \geq \frac{\max_{seq} F}{B},$$

where $\max_{seq} F$ is the maximum picture size over the all sequence. In order for the video stream to be continuous (actually having a constant rate), $S_c \geq T$; otherwise there would be a time interval between two subsequent frames during which no bits exit the encoder. As a consequence, *CBR MPEG encoding always introduces a delay larger than the video frame period.* With reference to the model depicted in Figure 1, S_c contributes to the processing (**P**) and processing resynchronization (**PR**) delay components.

[9]See [10] for further information on the MPEG encoding process.

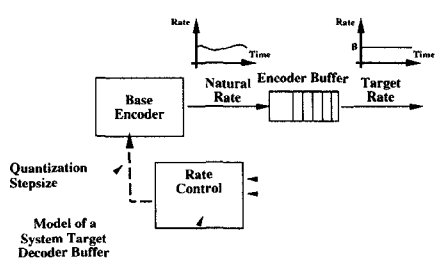

Fig. 7. High Level Model of a CBR MPEG Encoder.

Figure 8 shows the picture size obtained when encoding four video sequences with the `dvdenc` [16] CBR MPEG encoder. Each sequence is encoded at three different target rates. The maximum picture size in each sequence determines the lower bound on the coding shaping delay. In order to keep constant the rate of the video stream (i.e., avoid the encoder buffer to underflow), the encoder must grow larger I-frames as P-frames become smaller.

P-frame dimension is determined by the motion in the scene: the slower the scene, the smaller the size of P-frames. Videoconferencing scenes are usually quite static: P-frames are expected to be small and I-frames consequently large. In order to obtain a confirmation from experimental data, we encoded two completely static scenes built by replicating 120 times the same picture of the "Cheerleaders" and "Hockey" sequences, respectively. Figure 9 shows the resulting picture size for the same target rates used in the previous experiment. As expected, the maximum picture size, i.e., the lower bound on the coding shaping delay, has significantly increased with respect to Figure 8, especially at low bit rates.

The following of this section is devoted to identifying the maximum picture size given the characteristics of the CBR MPEG encoder. Since the rate control function aims at avoiding the encoder buffer to overflow, picture size is upper bounded by the buffer dimension. It follows that the coding shaping delay can be reduced by exploiting a small encoder buffer. Hence, the issue becomes how to reasonably dimension the encoder buffer. Whenever a picture smaller than $B \cdot T$ is produced, the buffer must contain a backlog large enough to guarantee the continuity of the stream. Thus, the smaller the buffer, the less picture size can vary.

In principle, the buffer should be large enough to allow P-frames to be encoded with no bits when the image is completely static. This is particularly important when dealing with videoconferencing because the camera can be pointed, for example, over a blackboard thus capturing a completely static scene. The less bits are used to encode P-frames, the more are dedicated to I-frame (thus creating the backlog in the buffer) and the higher is the quality of the resulting image. Quality of static images is particularly critical since human eye is more sensitive to errors on static images, than on moving scenes.

Null size P-frames can be produced if an I-frame can be as large as the whole amount of bits sent during a GOP. I.e., in order to deliver maximum quality of static images, the encoder buffer size must be at least the GOP size; this results in a *coding shaping delay on the order of the GOP duration.* Such a coding shaping delay is not acceptable when aiming at Objective 1: for example, if operating at 30 frames per second with a GOP size of 15 pictures, the coding shaping delay is 500 ms.

In summary, the CBR MPEG encoding scheme is not suited to high quality videoconferencing. In the following the exploitation of VBR MPEG is evaluated.

C. Transmission of VBR MPEG

While the CBR encoder introduces an unacceptable delay in the encoder, a VBR video stream may impact the network performance leading to either high delay, or high loss, or the need for an over-allocation of communication resources in some of the configurations.

C.1 Circuit Switching

In principle, if the videoconference call is allocated a circuit with bandwidth larger than the maximum instantaneous rate of the encoder, bits are transmitted as soon as they are produced

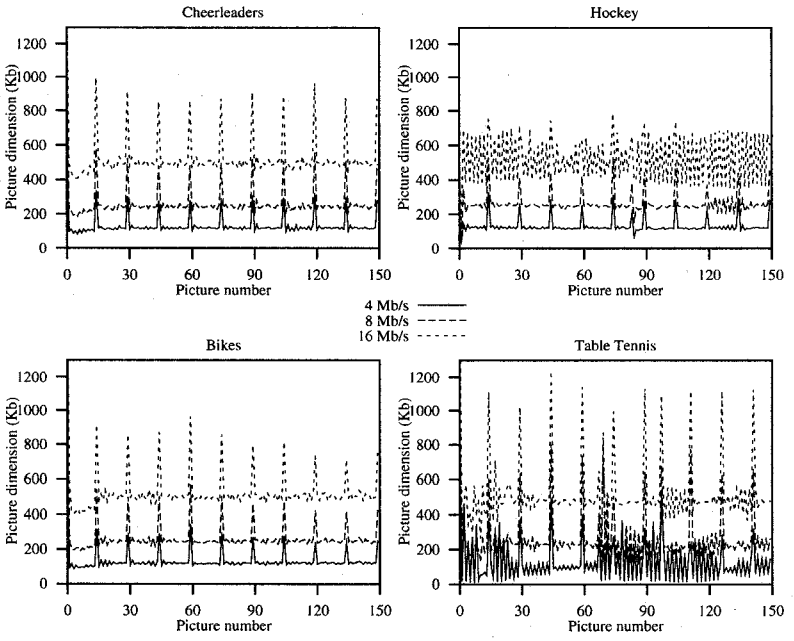

Fig. 8. Dimension of Pictures in CBR MPEG Encoding with $N = 15$.

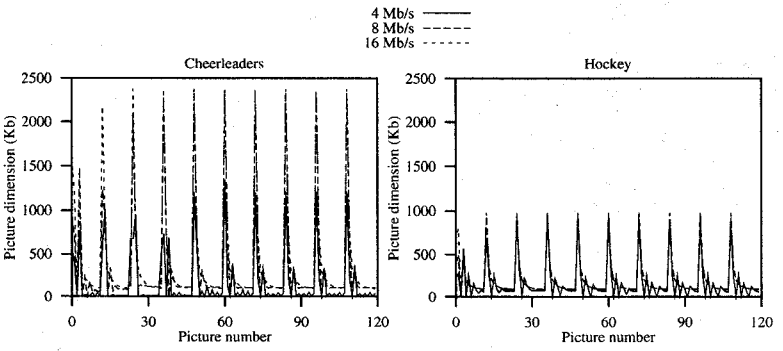

Fig. 9. Dimension of Pictures in CBR MPEG Encoding of a Static Scene

and bits get to the decoder at the same rate they had been produced after having experienced the constant propagation (Pr) and switching (Sw) delays. The end-to-end delay is given by

$$\Delta_{VBR}^{CS} = CD_M + Sw + Pr,$$

where the first term is the overall coding-decoding-resynchronization delay and contributes to the processing delay[10] **P**, while the other two terms contribute to the network delay component **N**.

Since we are dealing with real-time video a picture should be encoded (decoded) within the video frame period, i.e., $CD_M \leq 2 \cdot T$. Thus, if a scene is captured at 30 frames per second and the network delay is 20-40 ms (like in a transoceanic call), the end-to-end delay is about the 100 ms bound.

The system configuration discussed in this section provides a lower bound on the end-to-end delay in a videoconferencing system exploiting MPEG compression. Nevertheless, it is not practical since the bandwidth of the circuit allocated to the videoconference call is only partially used and the unused fraction is wasted.

Actually, when the bandwidth of the circuit is larger than the peak rate of the encoder, the end-to-end delay can be reduced by exploiting only intra-frame coding. Since motion estimation is the most time consuming function of the encoding process, CD_M is significantly reduced using a GOP of one picture[11]. A circuit having bandwidth smaller than the peak rate of the encoder can be allocated to a videoconference call introducing a network shaping delay S_{VBR}^{CS}.

C.2 Time-driven Priority

As soon as all the bits for encoding a picture are produced by the encoder, they are inserted into a packet and sent at the full speed of the ingress link. The end-to-end delay of the system is given by

$$\Delta_{VBR}^{TDP} = CD_M + L \cdot T_f + Pr, \quad (4)$$

where L is the number of TFs a packet takes to travel from sender to receiver. With reference to the model of the videoconferencing system depicted in Figure 1, the first term contributes to the processing delay component **P** (and in a negligible way to **PR**), while the second term contributes to the network delay component **N**.

Equation (4) is the actual end-to-end delay only if the nodes on the path from sender to receiver perform RISC-like forwarding of packets. To guarantee the fixed $L \cdot T_f$ network delay and

[10] Actually, CD_M contributes also to the processing resynchronization delay component **PR**, but since the variability of the **P** component is relatively small, the contribution of **PR** can be neglected. See [11] for details.

[11] Encoding all the pictures as I-frames is equivalent to using Moving JPEG encoding.

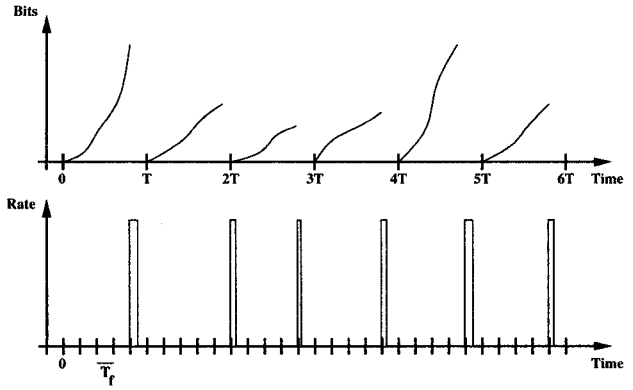

Fig. 10. Transmission of VBR MPEG Video with Time-driven Priority.

loss free delivery, resources must be allocated in the network and video frames sent during reserved TFs. To reserve resources in packet switched networks with TDP, the amount of data to be sent and their timing must be known at reservation time so that the proper fraction of link capacity can be reserved during the proper TFs. The amount of bits reserved should be larger than (and as close as possible to) the dimension of the picture being sent. As was shown in Figure 5, picture dimension is not known in advance and resource reservation may not be accurate.

If during a TF a user sends more bits than the reserved amount, the network does not provide any guarantee on the delivery of the excess data units. If, on the other hand, the videoconferencing application uses only a fraction of the reserved capacity, the leftover bandwidth can be used by best-effort traffic and is not wasted (unlike circuit switching). Even though this is acceptable from the network point of view, the solution is not optimal for the user that is possibly paying for the allocated bandwidth and would like to use it all by himself.

Choosing a Bound on Picture Dimension. Since videoconferences are expected to be slow moving scenes, we propose to reserve different amounts of bits for transmission of I-frames and P-frames. These amounts determine the bandwidth reserved to the videoconference call as

$$B^{TDP} = \frac{F^I + (N-1)\cdot F^P}{N \cdot T}, \tag{5}$$

where F^I and F^P are the amount of bits reserved for I-frames and P-frames, respectively, N is the number of pictures per GOP, and $N \cdot T$ is the GOP duration.

As discussed, the relative dimension of I-frames and P-frames yielded by a natural MPEG encoder depends on the amount of motion in the scene. The *picture ratio*

$$\alpha = \frac{F^I}{F^P} \tag{6}$$

must then be chosen wisely depending to the amount of motion expected in the scene to be encoded and transmitted. This is, in general, a difficult task, but in the particular case of videoconferencing, scenes are likely to be slow and α consequently large.

Combining Equations (5) and (6), the amount of bits to be reserved to each frame can be expressed as a function of the bandwidth to be allocated as

$$\begin{cases} F^P = \frac{B^{TDP} \cdot N \cdot T}{N+\alpha-1} \\ F^I = \alpha \cdot F^P \end{cases} \tag{7}$$

Controlling Picture Dimension. The reserved bandwidth is used more efficiently if the encoding process is controlled and picture size is kept below (and as close as possible to) the amount of bits reserved. The encoder is extended with a *rate control function*, as shown in Figure 11, which tunes the parameters of the basic MPEG coding process so that the dimension of each picture is best fitted to the target size associated with its type. Among the parameters of the MPEG encoding process the quantization step size is the most suited to this purpose. Various control functions can be devised to maximize the quality of encoded pictures [11].

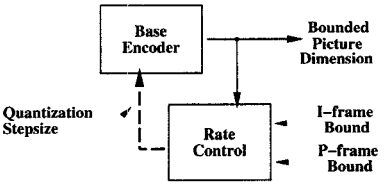

Fig. 11. MPEG Encoder for Controlling Dimension of Frames.

Complex Periodicity Scheduling. Scheduling, i.e., the choice of the TFs to be reserved to a video-conference call, is simplified by considering the nature of the application generating traffic. Different amounts of bits should be reserved in the TFs intended for sending I-frames and those for P-frames. The time cycle must be set to an integer multiple M of the GOP period and $N \cdot M$ TFs must be reserved within the time cycle. The choice of the TFs to be reserved on each link on the path between sender and receiver is called *complex periodicity scheduling*[12]. The choice of the TFs impacts both network performance (in terms of maximum number of real-time connections concurrently supported) and the end-to-end delay of the video-conference call.

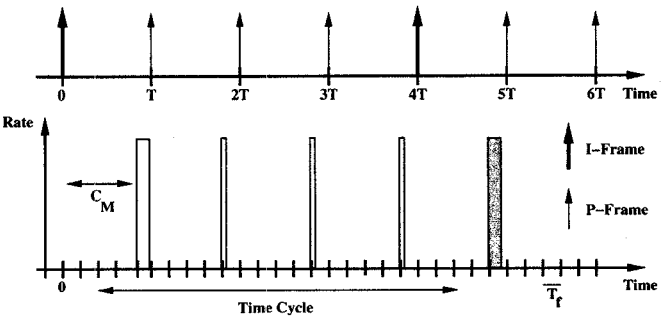

Fig. 12. Time-driven Priority and Complex Periodicity Scheduling.

Figure 12 shows a sample reservation with $M = 1$ and $N = 4$. The upper diagram depicts the frame grabbing time and the lower one shows the amount of bits reserved in the TFs; the TF reserved for sending a picture is allocated right after the maximum coding delay C_M form the picture grabbing. This is the optimal schedule which leads to minimum delay as given by Equation (4). If the optimal schedule is not feasible, a *scheduling shaping delay* [11] is introduced which contributes, together with $L \cdot T_f$, to the network delay component **N** identified in the model depicted in Figure 1.

C.3 Asynchronous Packet Switching

The end-to-end delay is given by

$$\Delta_{VBR}^{Async} = CD_M + \frac{P_s}{C} + Pr + \hat{Q}_M + E_r,$$

where CD_M is the maximum time required to encode and decode a picture (contributing to **P** and to a small **PR**), $\hat{Q}_M$

[12]This scheduling is said complex because not the same capacity is allocated during all the reserved TFs.

	Circuit Switching	Time-driven Priority	Asynchronous Packet Switching
Raw Video	$\Delta_{Raw}^{CS} = S_{Raw}^{CS} + Sw + Pr$ **N**	$\Delta_{Raw}^{TDP} = L \cdot T_f + Pr$ **N**	$\Delta_{Raw}^{Async} = \frac{F_r}{C} + Pr + \hat{Q}_M + E_r, E_r \in [0, \Delta Q]$ $\Delta_{Raw}^{Async-TS} = S_{Raw}^{Async} + \frac{P_s}{C} + Pr + \hat{Q}_M + E_r$ **N + NR**
VBR MPEG	$\Delta_{VBR}^{CS} = CD_M + Sw + Pr$ **P + N**	$\Delta_{VBR}^{TDP} = CD_M + L \cdot T_f + Pr$ **P + N**	$\Delta_{VBR}^{Async} = CD_M + \frac{P_s}{C} + Pr + \hat{Q}_M + E_r$ $\Delta_{VBR}^{Async-TS} = CD_M + S_{VBR}^{Async} + \frac{P_s}{C} + Pr + \hat{Q}_M + E_r$ **P + N + NR**

TABLE I

END-TO-END DELAY FOR THE SYSTEM CONFIGURATIONS CONSIDERED IN THIS WORK.

is some percentile of the maximum queuing delay, and $E_r \in [0, \Delta Q]$ is the excess resynchronization delay introduced by the replay buffer (contributing to **NR**). The end-to-end delay is dominated by $\hat{Q}_M$ that, with reference to the videoconferencing system model depicted in Figure 1, contributes to the network delay component **N** and to the network resynchronization delay component **NR**.

Resources are reserved in the network in order to bound the queuing delay (i.e., to reduce **N** and **NR**). As discussed in Section III-A.3, resource reservation is more efficient if traffic shaping is performed at the boundaries of the network, even though it introduces a network shaping delay (i.e., it increases **N** and **NR**). The delay globally experienced by a picture due to the traffic shaper depends on the natural bit generation rate of the encoder, the implementation of the traffic shaper, and the characteristics of the shaped traffic. The receiver has to compensate it by means of the network resynchronization delay introduced by the replay buffer. Thus, each packet experiences a network shaping delay S_{VBR}^{Async} partly in the traffic shaper and partly in the replay buffer. The end-to-end delay of the videoconferencing system is given by

$$\Delta_{VBR}^{Async-TS} = CD_M + S_{VBR}^{Async} + \frac{P_s}{C} + Pr + \hat{Q}_M + E_r.$$

IV. DISCUSSION

In this work we analyze the end-to-end delay of videoconferencing in six system configurations obtained by combining three network technologies with two encoding schemes. The results are summarized in Table I. We study the transmission of raw video and variable bit rate MPEG video over (*i*) circuit switching, (*ii*) time-driven priority packet switching, and (*iii*) asynchronous packet switching. In addition, we show that constant bit rate (CBR) MPEG encoding delivers long delay, which is on the order of the group of pictures (GOP) duration. Thus, if the sampling rate is of 30 frames per second and a GOP of 15 pictures is used (i.e., each I-frame is followed by 14 P-frames), the resulting delay is on the order of 500 ms.

Given the long distance and high bit rate requirements of videoconferencing, video compression should be used. Since CBR encoding has unacceptable delay, variable bit rate (VBR) encoding should be used. In the case of VBR encoding, circuit switching is not practical since the network utilization is very low. *Thus, packet switching should be used.* However, relying on asynchronous packet switching with statistical multiplexing and first come first serve queuing discipline can result in high loss and delay jitter under high load conditions. Other queuing disciplines, such as weighted fair queuing, can only guarantee deterministic no loss to CBR traffic, which as mentioned would result in an unacceptably large delay bound.

This study shows that having a global common time reference can be used for implementing time-driven priority with complex periodicity scheduling for transporting variable bit rate MPEG encoding. This will provide adequate delay bound which are *independent of the network load and the connection rate.* In such system configuration, the end-to-end delay (excluding propagation delay) can be smaller than the video frame period. Furthermore, time-driven priority with complex periodicity scheduling can *deterministically ensure no loss (due to congestion) of VBR traffic.* These unique results cannot be obtained with circuit switching or any other known alternative schemes.

ACKNOWLEDGMENTS

We thank Prasoon Tiwari for providing us with the software MPEG encoder **dvdenc** and Peter Westerink for his kind and useful help in understanding, modifying, and operating the encoder.

REFERENCES

[1] G. Karlsson, "Asynchronous transfer of video," *IEEE Communications Magazine*, pp. 118 – 126, Aug. 1996.

[2] ISO/IEC, *Information technology - Coding of moving pictures and associated audio for digital storage media up to about 1.5 Mbit/s*, International Organization for Standardization, 1993.

[3] B. Fufht, "A survey of multimedia compression techniques and standards. Part I: JPEG standard," *Real Time Imaging*, , no. 1, pp. 49 – 67, 1995.

[4] R. Steinmetz and K. Nahrstedt, *Multimedia: computing, communications & applications*, Prentice Hall, Upper Saddle River, NJ 07458, 1995.

[5] ITU-T, *Recommendation H.261*, Video codec for audiovisual services at p x 64 kbit/s, Mar. 1993.

[6] P. H. Dana, "Global positioning system overview," http://www.utexas.edu/depts/grg/gcraft/notes/gps/gps.html.

[7] 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*, vol. 2, no. 2, pp. 137–150, Apr. 1994.

[8] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queuing algorithm," *ACM Computer Communication Review (SIGCOMM'89)*, pp. 3–12, 1989.

[9] S. Shenker, C. Partridge, and R. Guerin, "Specification of guaranteed quality of service," Standard Track RFC 2212, Internet Engineering Task Force, Sept. 1997.

[10] D. Le Gall, "MPEG: A video compression standard for multimedia applications," *Communications of the ACM*, vol. 34, no. 4, pp. 47 – 58, Apr. 1991.

[11] M. Baldi and Y. Ofek, "End-to-end delay of videoconferencing over packet switched networks," Research Report RC 20669 (91480), IBM, Dec. 1996.

[12] C-S Li, Y. Ofek, and M. Yung, ""Time-driven priority" flow control for real-time heterogeneous internetworking," in *IEEE INFOCOM'96*, 1996.

[13] C-S. Li, Y. Ofek, A. Segall, and K. Sohraby, "Pseudo-isochronous cell switching in ATM networks," in *IEEE INFOCOM'94*, 1994, pp. 428–437.

[14] T. Turletti and C. Huitema, "Videoconferencing on the Internet," *IEEE/ACM Transactions on Networking*, vol. 4, no. 3, pp. 340 – 351, June 1996.

[15] M. Butto, E. Cavallero, and A. Tonietti, "Effectiveness of the 'leaky bucket' policing mechanism in ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 3, pp. 355 – 342, Apr. 1991.

[16] E. Linzer, "A robust MPEG-2 rate control algorithm," Unpublished technical report, IBM – T. J. Watson Research Center, department 924A.