

# Performance Models of Statistical Multiplexing in Packet Video Communications

BASIL MAGLARIS, MEMBER, IEEE, DIMITRIS ANASTASSIOU, MEMBER, IEEE, PRODIP SEN, MEMBER, IEEE, GUNNAR KARLSSON, AND JOHN D. ROBBINS

**Abstract**—Statistical or asynchronous time division multiplexing of variable bit rate moving image sources is a first step toward packetized video networks. Its goal is to efficiently utilize a common communications channel while maintaining uniform picture quality at the receiver. In this paper, we present queuing models of such schemes, and use them to assess their probabilistic behavior. First, we analyze the coding bit rate of a single video source as a function of time based on experimental data of a video telephone scene. Various statistical quantities are derived from bit rate data collected with a conditional replenishment interframe coding scheme. Two correlated Markov process models (one in discrete time and one in continuous time) are shown to fit the experimental data and are used to model the input rates of several independent sources into a statistical multiplexer. The continuous-time model is used with a flow-equivalent approximate queuing analysis to obtain the common buffer length distribution. The analysis is validated with computer simulations that use the discrete-time source model and take into account the discrete nature of the packet queue. Numerical results are presented for variable channel utilization and number of multiplexed video sources. Our analyses demonstrate the statistical smoothing effect of packetized video in a single link. The multiplexer performance, in terms of the probability of buffering or delaying beyond a certain limit, is shown to improve dramatically as the number of multiplexed sources increases beyond one.

## I. INTRODUCTION

DIGITAL video is expected to become a major traffic component on integrated services digital networks (ISDN's). Applications such as video conferencing, video telephone, and switched TV, impose very large bandwidth requirements on public and private networks. The installation of optical fibers down to the subscriber local loop will encourage the proliferation of video services, but as more bandwidth becomes available new services will tend to overload them. Apart from loading communication lines, switched broad-band services impose large loads on the switches; this is of great concern if switching in the networks of the future will employ an integrated flexible technology (such as fast packet switching) for voice, data, and video alike. It is, therefore, of particular interest to study the implication of bandwidth-reducing video compression algorithms [1] on

Paper approved by the Editor for Image Processing of the IEEE Communications Society. Manuscript received April 27, 1987; revised September 18, 1987. This work was supported in part by the New York State Center for Advanced Technology in Telecommunications, Polytechnic University, by the New York State Center for Advanced Technology on Computer and Information Systems and the NSF Center for Telecommunications Research, Columbia University, and by Bell Communications Research. This paper was presented in part at the First International Packet Video Workshop, Columbia University, New York, May 1987, and at the IEEE Global Communications (GLOBECOM) Conference, Tokyo, Japan, November 1987.

B. Maglaris and P. Sen are with the Department of Electrical Engineering, Polytechnic University, Brooklyn, NY 11201.

D. Anastassiou and G. Karlsson are with the Department of Electrical Engineering, Columbia University, New York, NY 10027.

J. Robbins is with Bell Communications Research, Morristown, NJ 07960. IEEE Log Number 8820873.

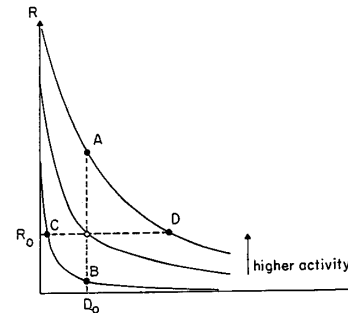


Fig. 1. Rate-distortion functions.

sharing network resources, specifically on multiplexing and switching in a broad-band ISDN.

A PCM-coded video signal sequence may result in a large bit rate (hundreds of Mbits/s). In limited bandwidth communication networks, a few video sources can easily fill all line capacities. Data compression techniques are currently employed to transmit coded full motion video within digital lines of lower capacity, typically 1.544 Mbit/s, but even as low as 56 kbits/s. We will consider a particular interframe coding scheme that encodes significant differences between successive frames of a moving sequence. The rate of such codes depends on the instantaneous activity of the scene, increasing during active motion and decreasing during inactive periods.

There is a complicated tradeoff between the minimum achievable coding bit rate  $R$  and the distortion  $D$  of the decoded images, described in information theory by the rate-distortion function  $R(D)$ , see, e.g., [2]. The entropy rate (in bits/s) of a source determines the maximum compression (or minimum bit rate) achievable for lossless ( $D = 0$ ) coding. For a moving image it will be time varying, depending roughly on the instantaneous activity or motion. Higher activity sources will have larger  $R(D)$  for the same  $D$  (Fig. 1). If some distortion  $D_0$  can be tolerated, the coding bit rate can be decreased, but it will remain time varying for a constant quality video output, as shown by the segment  $AB$  in Fig. 1. Similarly, if the coding bit rate is set constant, equal to  $R_0$ , the video quality will be time varying, worse for high-activity scenes as shown by the segment  $CD$  of Fig. 1. Such is the case for video encoders that feed a constant rate communication line or share network resources via synchronous time division multiplexing (TDM) and circuit switching. To tailor variable rate codes into a channel of speed lower than the coding rate at very high activity, *multimode* encoders equipped with buffers are used. The buffer absorbs statistical peaks of the coding rate by temporarily storing data in excess. If the amount of data in the buffer exceeds a certain threshold, the encoder is instructed to switch into a coding *mode* that has lower rate but worse quality to avoid buffer overflow. Similarly, when buffer underflow is approached, the channel speed is maintained by

increasing the bit rate, thus improving the quality. The efficiency of the multimode scheme increases with lower fixed speed on the output line; unfortunately, the quality of the received image will start suffering from highly visible variations that can be rather unpleasant to the human eye.

Variable bit rate video coding exhibits the type of statistical variations that made statistical multiplexing and packet switching attractive for bursty computer communications. These network architectures can dynamically support variable rate sources by smoothing out the aggregate of several independent streams in common buffers within the network. It is expected that such technologies may provide efficient video transport without the unpleasant variations in quality of multimode coders. In statistical or asynchronous time division (ATD) multiplexing, several independent sources share a line of capacity less than the sum of their peak rates. Instead of using individual buffers, all sources feed a common buffer and their cumulative bit rate tends to smooth out around the average rate as indicated by the law of large numbers. Packet switching is a network extension of statistical multiplexing, with data from individual sources segmented into small packets that are stored and forwarded from switch to switch toward their destination. The statistical smoothing of variable rate sources is achieved as packets are buffered in the network switches. Both statistical multiplexing and packet switching introduce variable delays in delivering data due to the buffering stages. In addition, they may introduce data losses due to buffer overflow. Thus, it is very important to select parameters and line speeds to minimize these effects while maintaining the efficiency of statistical averaging.

Statistical multiplexing of video was proposed and simulated by Haskell at Bell Laboratories in 1972 for *Picturephone* interframe coders [3]. For the same application and coding, Limb [4] simulated the behavior of a buffer that smooths out the variable bit rate of a single source. Recently, there has been a revival of this concept due to the increasing popularity of video communications, advances in video coding development and implementation, and the spreading interest in broad-band ISDN's in the US and abroad, e.g., the RACE program of the European Economic Community. A particular project of the RACE definition phase entitled ATDM deals, among other items, with packetized video. A discussion on the advantages of statistical or asynchronous time division (ATD) video multiplexing was reported by Verbiest [5] in 1986. Packetized video has also been simulated and related protocols have been devised within Columbia University's MAGNET local area network testbed [6], [7].

Packet switching was introduced in the 1970's, mainly for bursty computer communications that tolerate variable delays of the order of 1 s. Applications that require timely and synchronous delivery, such as telephone voice and real-time video, currently use dedicated circuit switching. This may change with the new concept of fast packet switching [8] that promises to convey large volumes of packetized digital voice. The asynchronous packetized transmission of conversational voice requires very stringent end-to-end delay; voice packets exceeding a given time threshold may not be used to synthesize the continuous voice stream at the receiver. Late packets thus result in voice losses that can be tolerated up to a certain degree. A similar problem must be considered in full motion video and particular attention must be paid to recovery from packet losses.

Several queueing analyses and simulations were recently reported to assess the buffering statistics of packetized voice [9]–[12]. Their complexity stems from the correlated nature of the stochastic process that models the rate of a voice source. A video source generates correlated bit rates but of a different statistical nature. The output of a digital voice phone alternates between talkspurts and silent periods. This corresponds to two states for the source output rate; the state corresponding to

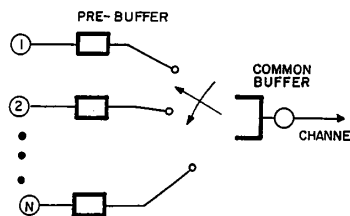


Fig. 2. Schematic of a statistical multiplexer.

zero bit rate occurs 60 percent of the time. In contrast, the output of a variable bit rate video coder exhibits continuous variations. Thus, new models and analytic techniques are needed to evaluate queueing statistics in packetized video. This paper is a first step in this direction.

Some measured data on the nature of the video coder rate were reported in 1972 by Haskell and Limb for the simulations on *Picturephone* buffering mentioned above [3], [4], and in 1981 by Koga *et al.* [13] for broadcast television. We generated our own data in the course of this work using a modified version of the *conditional replenishment* scheme proposed in [14]. In Section II, we present a statistical analysis of our data and introduce two Markov models that match their basic statistics. The first model (Model *A*) is a continuous-state autoregressive discrete-time Markov process which is used for our simulation experiments. The second model (Model *B*) is a quantized-state continuous-time Markov process which is used in our fluid-flow queueing analysis. In Section III, we present numerical results that show agreement between simulation and analysis. The analytic model is used to analyze the queueing behavior of statistical multiplexing of several independent, identically distributed video sources. The results show that the probability of buffering or delaying data beyond a certain threshold decreases dramatically as the number of multiplexed video sources increases.

## II. MODELING AND ANALYSIS ASSUMPTIONS

We assume video sources generating 30 frames/s. Each frame consists of approximately 250 000 pixels that are digitally coded. The conditional replenishment compression algorithm encodes and transmits the difference between pixel levels of subsequent frames if this difference exceeds a given threshold. As in [3] we assume that  $N$  independent video sources are multiplexed into a high-speed trunk. The unbuffered coded bits from each source are first stored in separate prebuffers and then join a common buffer as in Fig. 2. The prebuffers may perform a presmoothing of the source data over a frame period as will be explained later. The multiplexer assembles the data in the common buffer into blocks that are transmitted over the high-speed communication line. The blocks may be asynchronous time frames that combine portions of data from each source. The frame length depends on the instantaneous amount of data from the sources. Frame delimiters and source identification information must be included to enable the demultiplexing at the destination. A block may also be a packet of data assembled from a single source. Packets are stored and forwarded in a FIFO (first-in-first-out) mode in the same order as they are assembled. The packet length may be variable due to temporal changes in the source bit rate, or fixed, in which case the variable source rates will cause varying packet arrival rates. Each packet needs a header identifying the source, the destination, its sequence number, and possibly a time stamp to alert the network in case of excessive buffering delays. Error detection and correction overhead bits may be added.

We will model the statistical multiplexer as a queue that receives the encoded bit streams from the prebuffers; its

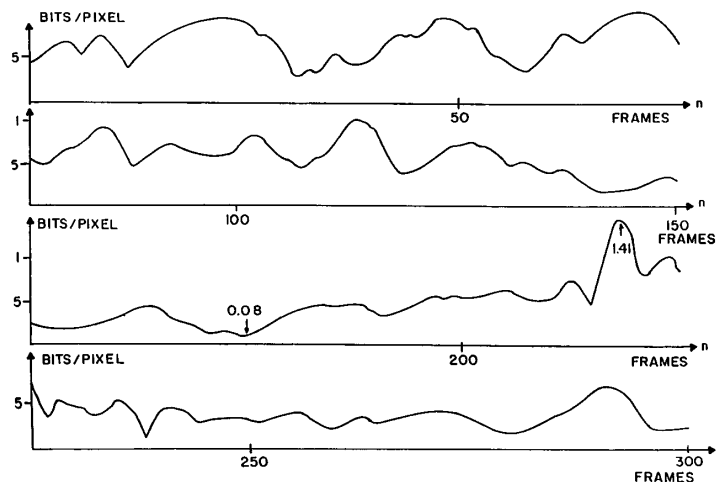


Fig. 3. Coding bit rate of the captured sequence (in bits/pixel).

service rate is determined by the speed of the channel. Issues related to the specific multiplexing technique, e.g., frame or packet formats and related protocols, are beyond the scope of this study. Our analysis is based on a continuous fluid-flow approximation of the traffic that does not take into account the discrete nature of packets or frames. The queue behaves like a reservoir of water that is fed from water supplies of time-varying rates and empties through a fixed-rate sink. The fluid-flow approximation is a powerful analytic tool which is able to capture correlated input models; it is accurate if the packet size is small compared to the speed of service [15]. The analysis agrees with our simulation model, in which bit streams from individual sources are assembled into packets of fixed length. Packets join the common buffer and are served in a FIFO order.

#### A. Source Model Inference from Experimental Results

In order to analyze the common buffer statistics, we need a model for the coded source rate. The rate depends on the compression algorithm and the nature of the video scene. For a scene without abrupt movement, such as the head of a person in a picturephone, we expect the rate to have a bell-shaped stationary probability density and to exhibit significant correlations for an interval of several frames. We verified this behavior by experimenting with a test image sequence depicting the head of a talking person. The sequence had a duration of 10 s or 300 frames. We measured the instantaneous bit rate  $\lambda(t)$  in bits/pixel. Recall that there are about 250 000 pixels per frame and 30 frames/s, thus 1 bit/pixel corresponds to 7.5 Mbits/s.

A key parameter in evaluating the bit rate is the number of pixels,  $N_p$ , which, combined in a group, give rise to a coded bit stream of length  $N_b$  into the prebuffer. The average bit rate is then equal to  $N_b/N_p$  bits/pixel. A large  $N_p$  leads to more averaging at the prebuffering stage of Fig. 2; it requires large individual prebuffers and introduces long delays. A small  $N_p$  preserves the variations of individual source rates at the input of the multiplexer. Statistical multiplexing in the common buffer will then average the overall bit rate in a more efficient manner. On the other hand, too small  $N_p$  values will introduce quasiperiodic variations with the period equal to a frame duration. For example, if  $N_p$  is the number of pixels in a horizontal scan line,  $N_b$  will be influenced by the variable activity of the lines and will tend to be repeated after a frame for the same line. These periodic variations are not worth

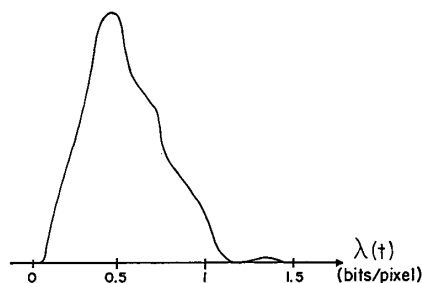


Fig. 4. Bit rate histogram.

exploiting in statistical multiplexing, and disappear with prebuffering of a whole frame. The prebuffers will introduce a one-frame delay of  $1/30$  s at the encoder side. Therefore, we have chosen  $N_p$  equal to the number of pixels in a frame. This choice also makes  $\lambda(t)$  dependent only on the varying activity of the frame sequence, and eliminates complicated properties from its statistics. Note that  $N_p$  is not necessarily related to the packet size, as there may be several packets formed from an  $N_b$  bit stream, and the packetization mechanism need not be synchronized with the frame sequence.

In Fig. 3, we show  $\lambda(t)$  as the bit rate (averaged for each frame) for all 300 frames (10 s) of the test sequence measured in bits/pixel. Although the measured rate is fixed for the duration of a frame ( $1/30$  s), we treat it as a continuous-time function since the frame period is very small compared to our time scale. The average value  $\mu$  over all 300 frames and the standard deviation  $\sigma$  were found to be  $\mu = 0.52$  bits/pixel and  $\sigma = 0.23$  bits/pixel. The maximum value of the bit rate was 1.41 bits/pixel and the minimum 0.08 bits/pixel. Fig. 4 shows a histogram of the values for the bit rate, an indication of the bell-like shape of its probability density function. Note that the rate cannot be negative, and that the density function is not exactly symmetric around its average  $\mu$ .

The autocovariance  $C(\tau) = E\{\lambda(t)\lambda(t + \tau)\} - \mu^2$  of the sequence was evaluated and is shown in Fig. 5 as a function of either time ( $\tau$ ) or frame ( $n$ ) difference. The dotted line corresponds to an exponential fit of the form  $\hat{C}(\tau) = \sigma^2 e^{-a\tau}$ , with  $a = 3.9 \text{ s}^{-1}$ , which is accurate up to  $n = 10$ , except that  $C(1)$  is slightly larger than  $\hat{C}(1)$ . The mismatch for  $n > 10$  is

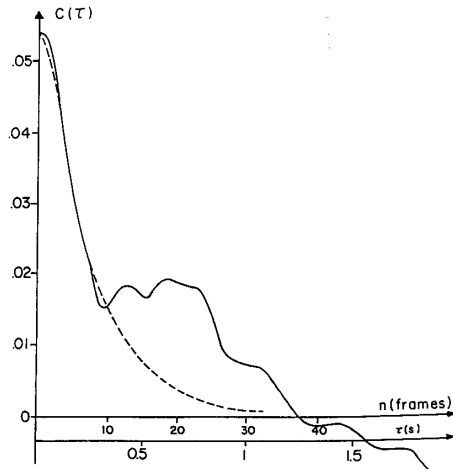


Fig. 5. Autocovariance function.

less meaningful. It is partly due to truncations in computing  $C(\tau)$  and would be eliminated if the sequence were much longer. Interestingly, it may partly be due to the fact that the head of the talking person in the scene was moving in a roughly periodic pattern.

We expect the average rate and its variance to be quite different in another test sequence, depending on the amount of motion. Indeed, in a video telephone conversation a talker will produce a higher bit rate than a listener. We separated the data into two segments of 5 s each and found slightly different variances. The minimum, average, and maximum rate values for the first segment are 0.10, 0.61, and 1.04 bits/pixel, while for the second that correspond to less movement, these values are 0.08, 0.43, and 1.41 bits/pixel, respectively. In both cases, however, the best fit exponent of the autocovariance was found again equal to  $a = 3.9 \text{ s}^{-1}$ .

We will present in the sequel two models of the encoded bit rate of a video source. The first model is suitable for queueing simulations, while the second leads to a simple queueing analysis. In both cases, we match the first- and second-order statistical properties of the measured data, as well as some features of the steady-state distribution. In particular, our models have the same mean and variance as the experimental data. The steady-state distributions of our models are unimodal and bell-shaped, which reflects the nature of the experimental data in Fig. 4. The queueing behavior at the multiplexer is not very sensitive to the specific nature of the distribution as our results will show.

A key feature of our models is that they match the exponential autocovariance fit mentioned above. Apart from our experimental evidence in Fig. 5, the exponential nature of the autocovariance is also supported from early results, obtained for video scenes that lasted much longer than our limited time experiment. In Fig. 6, we reproduce a curve reported by Haskell in [3, p. 270]. It portrays the correlation coefficient of the number of changes per field (a video frame can be transmitted in two interlaced fields) measured for a video telephone sequence lasting 1 h. The number of changes is related to the bit rate in interframe coding, although it is not directly proportional. We plotted in the figure three exponential curves,  $e^{-0.8\tau}$ ,  $e^{-0.7\tau}$ , and  $e^{-0.6\tau}$ . The experimental curve is bounded by the first and third curve. For  $\tau < 1/3 \text{ s}$ , or 20 fields at 60 fields/s, the first exponential is an almost perfect fit. For  $\tau > 1/2 \text{ s}$ , or 30 fields, the second curve seems to be a better fit. His results support our assumption of an exponential autocovariance. An agreement in the parameters of the

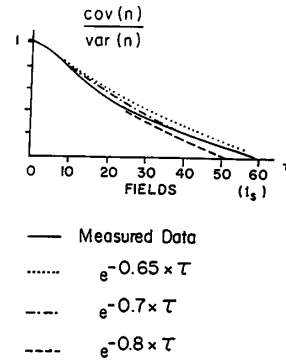


Fig. 6. Normalized correlation coefficient of the number of changes per field (Haskell, [3, p. 270, Fig. 7]).

autocovariance cannot be expected due to the difference between the measured quantities. The exponential autocovariance assumption was recently verified by Verbiest [16] in experiments involving a variety of moving scenes.

### B. Source Model A: Continuous-State Autoregressive Markov Model

We first model the coder rate as a continuous-state, discrete-time stochastic process. Let  $\lambda(n)$  represent the bit rate of a single source during the  $n$ th frame. A first-order autoregressive Markov process  $\lambda(n)$  is generated by the recursive relation

$$\lambda(n) = a\lambda(n-1) + bw(n) \quad (1)$$

where  $w(n)$  is a sequence of independent Gaussian random variables and  $a$  and  $b$  are constants. Assume that  $w(n)$  has mean  $\eta$  and variance 1. Further, assume that  $|a| < 1$ ; thus, the process achieves steady state with large  $n$ . The steady-state average  $E(\lambda)$  and discrete autocovariance  $C(n)$  are given by [17]

$$E(\lambda) = \frac{b}{1-a} \eta \quad (2)$$

$$C(n) = \frac{b^2}{1-a^2} a^n \quad n \geq 0. \quad (3)$$

The autocovariance is exponential and can fit the experimental data. The steady-state distribution of  $\lambda$  is Gaussian with mean  $E(\lambda)$  and variance  $C(0)$ . The histogram in Fig. 4 shows a bell-shaped density, truncated to zero. Assuming that the negative tail of the density of  $\lambda$  is very small, there is a reasonable matching of experimental data and the autoregressive model. From our measured data we obtain

$$E(\lambda) = 0.52 \text{ bits/pixel}$$

$$C(n) \approx 0.0536 \times (e^{-0.13})^n \text{ (bits/pixel)}^2.$$

The discrete autocovariance  $C(n)$  is obtained from the experimental fit  $\hat{C}(\tau) = 0.0536 \times e^{-3.9\tau}$  by sampling at  $n/\tau = 30 \text{ frames/s}$ . Matching (2) and (3) with the measured data, we have that

$$a = 0.8781 \quad b = 0.1108 \quad \eta = 0.572. \quad (4)$$

We believe that the continuous-state autoregressive model provides a rather accurate approximation of the bit rate. A better matching may be achieved if we increase its order to include the influence on  $\lambda(n)$  of several past values  $\lambda(n-k)$ ,  $k \geq 1$ . In this case, the autocovariance will be the sum of

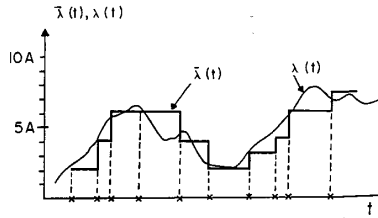


Fig. 7. Poisson sampling and quantization of the source rate.

several exponentials. Nevertheless, since a single exponential fit was found to dominate the decay of the autocovariance, we trust the first-order model.

### C. Queueing Simulation Using Model A

The model in (1) with the parameters (4) was used to generate the bit rates of each source in queueing simulation experiments. Instances of negative  $\lambda(n)$  were corrected to zero. This occurred rarely enough and did not significantly alter the measured moments of the simulated rates. In the simulation model,  $N$  sources generate independent bit streams with rate  $\lambda(n)$  for the duration of their  $n$ th frame. We assumed that the sources need not be synchronized in their frame sequences. Thus, the first frame occurrence of the  $N$  sources is randomized over the interval of a frame. Once initialized, the sources keep their individual frame synchronizations and their rate is generated according to the autoregressive formula (1). Bits generated from a single source over a frame period join a prebuffer as in Fig. 2. At the end of the frame, the bits in the prebuffer are packetized into fixed length packets that proceed into the common queue. Leftover bits that could not fill a full packet remain in the prebuffer, and are packetized within the next frame period. Packets in the common buffer are served in a FIFO order. Packet transmission times are fixed, proportional to the packet length.

We collect statistics (as time averages) of the common buffer length to estimate the average and variance of the queue length, and the probability that the queue exceeds certain levels. Results are presented in Section III.

### D. Source Model B: Discrete-State, Continuous-Time Markov Process

Model A is easy to simulate but cannot lead to a queueing analysis of manageable complexity. Even a continuous flow approximation of the queueing process with an input as in (1) leads to two-dimensional diffusion partial differential equations, with reflecting barriers at zero for both the input rate and the queue size. Such problems are encountered in stochastic storage theory; the case of queues with continuous correlated inputs is of particular analytic complexity [18]. We developed instead, a discrete-state Markov model that allows simple analytic tools to be employed.

In Model B the bit rate is quantized into finite discrete levels. Transitions between levels are assumed to occur with exponential transition rates that may depend on the current level. Thus, unlike Model A, we will approximate the rate by a continuous-time process  $\tilde{\lambda}(t)$  with discrete jumps at random Poisson times. The rate of these Poisson times and the probability of the jump size may change depending on the level of the bit rate. Model B can be obtained from the continuous-state bit rate in Fig. 3 by sampling the latter at random Poisson time instances (with appropriate rates) and quantizing the state at these points as illustrated in Fig. 7. Recall that we have treated the *measured* rate as a continuous-time, continuous-state process  $\lambda(t)$ , since the frame period is very small compared to our time scale. We can improve the approxima-

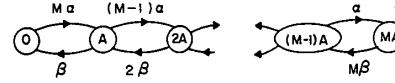


Fig. 8. State transition diagram—Model B.

tion in Model B by decreasing the quantization step and increasing the sampling rate. These parameters are related, for example, to take advantage of a smaller quantization step we should increase the sampling rate.

Model B is a discrete finite-state, continuous-time Markov process. Its state space is the set of the quantized levels up to a maximum level. The quantization step, the number of states, and the transition rates can be tuned to fit the average variance and autocovariance function of the measured data as before. Model B will be used to analyze the statistical multiplexer as a continuous-state queue (a fluid reservoir) that is filled from  $N$  variable rate sources each with rate  $\lambda(t)$ . Thus, we will tune its parameters to the aggregate instantaneous input rate  $\lambda_N(t)$ , rather than the single source bit rate  $\lambda(t)$ . The total rate is the sum of  $N$  independent random processes each with mean  $E(\lambda)$  and autocovariance  $C(\tau) \approx C(0)e^{-a\tau}$  at steady state. The steady-state mean and autocovariance of  $\lambda_N(t)$  will then be

$$E(\lambda_N) = N \times E(\lambda) = 0.52 \times N \text{ bits/pixel}$$

$$C_N(\tau) = N \times C(0) \times e^{-a\tau} \approx 0.0536 \times N \times e^{-3.9\tau} \text{ (bits/pixel)}^2.$$

(5)

There is an infinite number of choices of Markov models that can fit the parameters in (5). The actual choice should be guided by further examination of the video coder output and the complexity of the corresponding queueing analysis. As an example, if the coding rate involves sudden increases due to error recovery procedures or abrupt change of scene, this can be modeled by allowing big jumps in  $\tilde{\lambda}_N(t)$  with a certain transition rate. In this paper, we are not considering such phenomena. In the recorded data, Fig. 3, we did not observe discontinuities and very steep changes. We concluded, therefore, that a birth-death Markov model will accurately describe the aggregate source bit rate (a birth-death process allows only transitions between neighboring states or quantization levels [17]). We further expect the tendency of the bit rate toward higher levels to decrease at high levels, and inversely, the tendency of the bit rate toward lower levels to increase at high levels. This will result in a bell-shaped stationary distribution of the state as the experimental data indicate (Fig. 4). A simple birth-death process that exhibits this behavior, and has exponential autocovariance, is given by the transition diagram in Fig. 8. The state  $\tilde{\lambda}_N(t)$  of the process in Fig. 8 represents the quantized level of the aggregate bit rate of  $N$  sources. We assume uniform quantization step  $A$  bits/pixel, and  $M + 1$  possible levels,  $(0, A, \dots, MA)$ . The exponential transition rates  $r_{i,j}$  from state  $iA$  to state  $jA$  are given by

$$r_{i,i+1} = (M-i)\alpha \quad i < M$$

$$r_{i,i-1} = i\beta \quad i > 0$$

$$r_{i,i} = 0$$

$$r_{i,j} = 0 \quad |i-j| > 1.$$

(6)

It can be easily shown [19, p. 107] that  $\tilde{\lambda}_N(t)$  at steady state will have a binomial distribution with mean  $E(\tilde{\lambda}_N)$ , variance  $\tilde{C}_N(0)$ , and exponential autocovariance  $\tilde{C}_N(\tau)$

$$P\{\tilde{\lambda}_N(t) = kA\} = \binom{M}{k} p^k (1-p)^{M-k}, \quad p = \frac{\alpha}{\alpha + \beta} \quad (7)$$

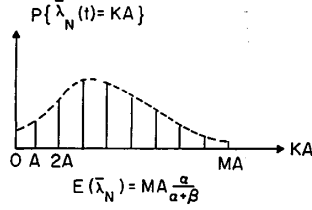


Fig. 9. Steady-state probabilities of the aggregate source rate—Model B.

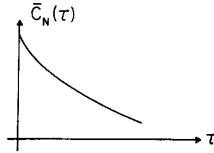


Fig. 10. Steady-state autocovariance of the aggregate source rate—Model B.

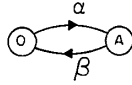


Fig. 11. Minisource model.

$$E(\bar{\lambda}_N) = MAp \quad (8)$$

$$\bar{C}_N(0) = MA^2p(1-p) \quad (9)$$

$$\bar{C}_N(\tau) = \bar{C}_N(0)e^{-(\alpha+\beta)\tau}. \quad (10)$$

The distribution and autocovariance of  $\bar{\lambda}_N(t)$  are shown in Figs. 9 and 10. Notice, that the distribution is bell shaped, but unlike Model A it does not allow negative bit rates. The parameters of the model  $M$ ,  $A$ ,  $\alpha$ , and  $\beta$  are obtained by matching (7)–(10) with the measured values in (5). With the number of quantization levels  $M$  as a parameter, and for a given number of multiplexed sources  $N$ , they are

$$\beta = a \left( 1 + \frac{N \times E^2(\lambda_N)}{M \times C_N(0)} \right) = 3.9 \left( 1 + \frac{5.04458N}{M} \right)$$

$$\alpha = a - \beta = 3.9 - \beta$$

$$A = \frac{C_N(0)}{E(\lambda_N)} + \frac{E(\lambda_N)}{M} = 0.1 + 0.52 \frac{N}{M}. \quad (11)$$

The bit rate can span the interval  $0 \leq \bar{\lambda}_N(t) \leq MA$ ;  $M$  should be selected large enough to span all likely bit rate values. Interestingly,  $\lim_{M \rightarrow \infty} A = C_N(0)/E(\lambda_N) = 0.1$  bits/pixel. Thus, for a finite number of sources  $N$ , the binomial model does not converge to the continuous-state Model A by decreasing the step size. Nevertheless, we found that with  $M = 20 \times N$ , the analytic results using Model B are in close agreement with simulations that use Model A (see Section III).

In the binomial model, the rate  $\bar{\lambda}_N(t)$  can be thought of as the aggregate rate from  $M$  independent *minisources*, each alternating between transmitting 0 bits/pixel (called the OFF state) and  $A$  bits/pixel (the ON state) according to a Bernoulli distribution. As shown in Fig. 11, a minisource turns ON with exponential rate  $\alpha$  and OFF with rate  $\beta$ . Such a source model was proposed in analyzing statistical multiplexing of voice sources with speech activity detection, e.g., [10]. The ON state represents active talkspurts and the OFF period to silence intervals, both assumed exponentially distributed. The aggregate

rate out of  $M$  voice sources corresponds exactly to our quantizing the aggregate bit rate of  $N$  video sources into  $M$  levels.

### E. Queueing Analysis Using Model B

In [20], Anick, Mitra, and Sondi presented a very elegant analytic solution for the queue statistics of a continuous-state queue (fluid-flow equivalent) with input as in the binomial model above. The same problem was considered earlier by Kosten [21], but with the number of levels  $M$  approaching infinity. His analysis involves numerical complexities that are avoided in [20]. The methodology in [20] found a natural application in analyzing statistical multiplexing of voice sources, and Daigle and Langford [12] compare it favorably against other methods of greater numerical complexity. A different approach was taken by Stern [9] who uses the binomial model to represent voice sources, but treats voice data as discrete packets rather than assuming a fluid-flow model. In the course of this work, we used the fluid-flow model because it is simpler to implement and produced results consistent with simulations of discrete packet models. In what follows, we summarize the fluid-flow queueing formulation for a general continuous-time, discrete-state Markov process approximation of the aggregate input bit rate. The binomial source model is a special case.

Let a continuous-state queue be fed by an input source with rate  $\bar{\lambda}_N(t)$  units of flow (bits/s). The input flow rate can assume discrete levels  $(0, A, 2A, \dots, MA)$ . Let  $r_{i,j}$  denote the exponential transition rate from level  $i$  to level  $j$ . The queue empties with fixed rate  $c$  units of flow per time unit. We denote the size of the queue by  $q(t)$ . A complete description of the queueing system requires a two-dimensional state  $\{q(t), \bar{\lambda}_N(t)\}$ . The first component is a continuous variable, while the second is discrete, assuming finite values. The joint statistics of the state at a time instance  $t$  can be described in terms of

$$P_i(t, x) = P\{\bar{\lambda}_N(t) = iA, q(t) \leq x\}.$$

The forward transition equations from time  $t$  to  $t + \Delta t$  are

$$P_i(t + \Delta t, x) = \sum_{j \neq i} r_{j,i} \Delta t P_j(t, x) + \left( 1 - \Delta t \sum_{j \neq i} r_{i,j} \right) \cdot P_i\{t, x - (iA - c)\Delta t\} + O(\Delta t^2).$$

As  $\Delta t \rightarrow 0$ , and ignoring second-order terms  $\Delta t^2$ , the evolution of the process is governed by the system of linear differential equations

$$\begin{aligned} \frac{\partial P_i(t, x)}{\partial t} + (iA - c) \frac{\partial P_i(t, x)}{\partial x} \\ = \sum_{j \neq i} r_{j,i} P_j(t, x) - P_i(t, x) \sum_{j \neq i} r_{i,j}, \quad 0 \leq i \leq M. \end{aligned}$$

If the utilization  $\rho$  is less than 1

$$\rho = \frac{E(\bar{\lambda}_N)}{c} < 1$$

the process achieves steady state with limiting distribution  $\lim_{t \rightarrow \infty} P_i(x, t) = F_i(x)$ . Then, the system is described by a system of ordinary differential equations

$$\begin{aligned} (iA - c) \frac{dF_i(x)}{dx} &= \sum_{j \neq i} r_{j,i} F_j(x) - F_i(x) \sum_{j \neq i} r_{i,j}, \quad 0 \leq i \leq M \\ F_i(x) &= 0 \quad x < 0 \\ F_i(0) &= 0 \quad \text{for } iA > c. \end{aligned} \quad (12)$$

The initial conditions follow from the observation that the buffer cannot be empty if the instantaneous rate  $iA$  is larger than the service rate  $c$ . An additional condition is obtained as  $x \rightarrow \infty$

$$F_i(\infty) = P\{\bar{\lambda}_N(t) = iA\}.$$

The set of equations (12) can be written in matrix form by using the vector  $F(x) = (F_0(x), \dots, F_M(x))^T$

$$D\dot{F}(x) = RF(x) \quad (13)$$

where  $D$  is a diagonal matrix with elements  $(iA - c)$  and  $R$  is the rate transition matrix from (12). Assuming that the multiplexer rate  $c$  is not equal to any of the input levels  $iA$ ,  $D$  is nonsingular. Let  $\Phi_i$  and  $z_i$  denote the eigenvectors and eigenvalues of  $D^{-1}R$ . The solution of (13) is given in terms of its known value at  $x = \infty$ , the eigenvalues and corresponding eigenvectors of  $D^{-1}R$  as follows (see, for example, [22]):

$$F(x) = F(\infty) + \sum_i k_i \Phi_i e^{z_i x}. \quad (14)$$

The sum in (14) is taken over all eigenvalues in the strict left half-plane for the solution to be a steady-state probability distribution ( $\rho < 1$  is required for this). The constants  $k_i$  in (14) are determined from the initial conditions

$$F_j(0) = 0 = F_j(\infty) + \sum_i k_i \phi_{ij} \quad \text{for } c/A < j \leq M$$

where  $\phi_{ij}$  denotes the  $j$ th element of  $\Phi_i$ . The steady-state distribution of the buffer length  $x$  is given by

$$F(x) = P\{q(t) \leq x\} = \sum_{i=0}^M F_i(x) \quad (15)$$

and the probability that the buffer size exceeds a certain length, referred to as the *survivor function*, is  $\bar{F}(x) = 1 - F(x)$ .

For the special case of the source model in Fig. 8, the rates in (6) simplify (12) to

$$(iA - c) \frac{dF_i(x)}{dx} = (M - i + 1)\alpha F_{i-1}(x) + (i + 1)\beta F_{i+1}(x) - [i\beta + (M - i)\alpha]F_i(x), \quad 0 < i < M. \quad (16)$$

Similar equations apply for  $i = 0$ , and  $i = M$ . The matrix  $R$  becomes a tridiagonal matrix. For this case, Anick *et al.* [20] were able to obtain analytic expressions for the eigenvalues, eigenvectors, and constants involved in the solution (14) that only require solving quadratic equations. They showed that for a stable system ( $\rho < 1$ ) there are exactly  $M - \lfloor c/A \rfloor$  real negative eigenvalues and one eigenvalue at zero. Thus, the initial conditions in (12) provide enough equations to solve for the constants  $k_i$  in (14). We coded their procedure, and in the next section we report results that we obtained from formulation (16) with the parameters in (11).

### III. NUMERICAL RESULTS

In this section, we present a discussion of some results obtained from our models. As mentioned in the previous section, a queueing model using the discrete-time continuous-state source Model A is difficult to analyze but straightforward to simulate. We thus simulated the queue dynamics using source Model A. The queueing analysis using source Model B was carried out as mentioned in the previous section. The parameter matching discussed there was used. In both cases, we are interested in the mean and variance of the queue size at the multiplexer, and the survivor function  $\bar{F}(x)$ . The survivor function  $\bar{F}(x) = P\{q(t) > x\}$  at steady state is an

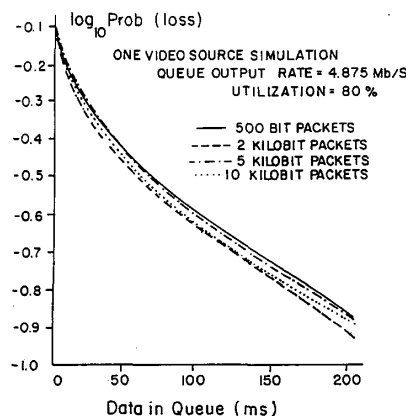


Fig. 12. Variation of loss probability with packet size Model A—simulations.

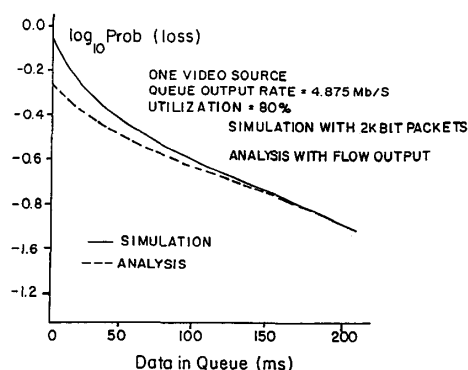


Fig. 13. Comparisons of loss probabilities—one source simulations (Model A) and analysis (Model B).

appropriate figure of merit because it represents the fraction of data that join the multiplexer when its queue size exceeds a threshold  $x$ . For real time video, there is a maximum allowable queue size  $x_0$ ; any packets that join the queue with more than  $x_0$  data ahead of them will arrive at their destination too late for resynthesizing the transmitted image sequence. For this reason, the survivor function  $\bar{F}(x)$  can be viewed as a loss probability. We have chosen to specify the queue size  $x$  in terms of time units; thus  $x = 30$  ms denotes that it would require 30 ms to empty a queue of this size at the output speed of the multiplexer. Because of the flow model used in the queueing analysis, these units seem to be the most appropriate and provide us with a normalization with respect to the output speed of the multiplexer.

Fig. 12 shows the relative insensitivity of the survivor function  $\bar{F}(x)$  to packet length for the simulation results using source Model A, when the queue size is measured in ms. In the simulation runs for Fig. 12, the nominal utilization was  $\rho = 80$  percent. However, due to the variation in the generated random sequences driving the runs, the actual utilization varied slightly. The simulation was carried out for the situation of one video source followed by a prebuffer feeding a packet queue.

Figs. 13 and 14 exhibit the close agreement between the survivor functions for the two models for the case of one and five video sources. The nominal utilization was 80 percent but the analysis with Model B was carried out for the actual utilization observed in the simulation run using Model A. The

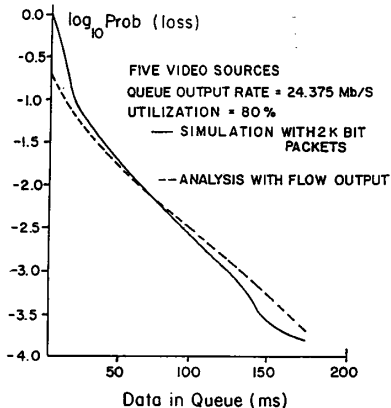


Fig. 14. Comparisons of loss probabilities—five sources simulations (Model A) and analysis (Model B).

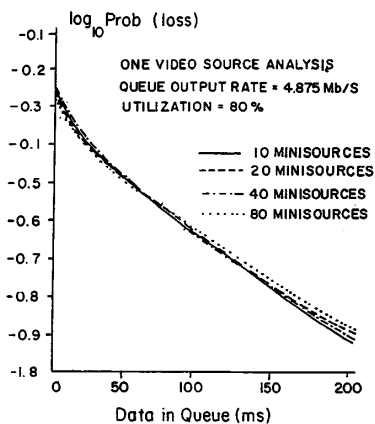


Fig. 15. Sensitivity of loss probabilities to the number of quantization levels—Model B, analysis.

agreement between the two models is very encouraging. Recall that Model A uses a Gaussian distribution for the source rates, while Model B uses a discrete binomial distribution. The agreement thus shows the insensitivity of the queueing behavior to the specific steady-state distribution used and supports our approach of matching first- and second-order statistical properties.

We note that the difference in the survivor function at and near the zero queue size level are due to the basic differences in the queueing models. In the queueing simulation using Model A, the queue is a single server queue, while the queueing analysis using Model B uses the flow approximation and behaves like an infinite-server queue. This has been noted earlier in the context of voice-source multiplexing [12].

The insensitivity of the survivor function to the number of quantization levels (or minisources) in the analysis using Model B is the subject of Fig. 15. The results presented are for one video source, but the trend carries over for more than one source. These results show that we can choose the number of quantization levels in a fairly wide range to satisfy other constraints, if necessary, without significantly perturbing the results. Our studies show that 10 levels are quite adequate, and it is really unnecessary to consider a large number of levels. We also note that increasing the number of levels increases the running time of the analysis program, but the difference between that for 20 sources and for 80 sources was hardly noticeable.

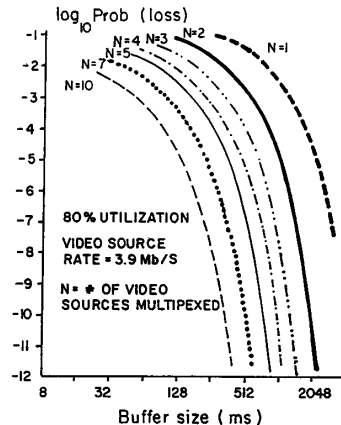


Fig. 16. Variation of loss probabilities with the queue size parametrized by the number of sources—utilization 80 percent.

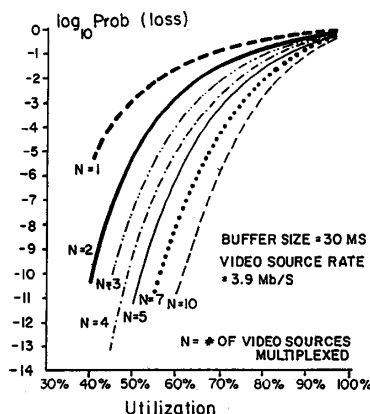


Fig. 17. Variation of loss probabilities with utilization parametrized by the number of sources—queue size 30 ms.

Source Model A is perhaps a closer approximation to reality than our Model B, from our intuitive understanding of the situation, as well as from our perusal of the data we used. However, the close agreement observed between results obtained using the two models prompted us to consider the source Model B as a more useful tool in analyzing the video multiplexer problem. This is because as with every simulation, the simulation using Model A requires long running times to yield reliable results. This is even more of a problem in our case where we are not only interested in averages and variances, but also in probability distributions for packet loss. The situation can only get aggravated when the multiplexing of multiple video sources is considered. On the other hand, the analysis using Model B is extremely fast and can be used to analyze multiple source situations with no extra effort and only a slight increase in running time.

Figs. 16–18 show the results of applying the analysis to the multiplexing of more than one video source. In the analyses presented, we chose the number of minisources  $M$  to be  $20 \times N$  (where  $N$  is the number of video sources being multiplexed). This leads to a constant quantization level for all cases, as shown in (11). We assumed that each source has an average bit rate  $E(\lambda) = 3.9$  Mbits/s; the channel speed  $c$  was adjusted to provide a specific utilization  $\rho = [E(\lambda) \times N]/c$ .

Fig. 16 shows the dramatic reduction in the survivor function (loss probability) due to the multiplexing of several

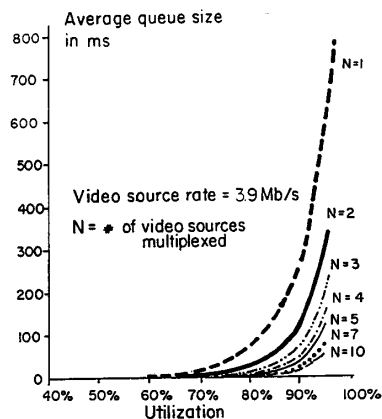


Fig. 18. Variation of average queue size with utilization parametrized by the number of sources.

video sources, for a constant utilization  $\rho = 80$  percent. Even as  $N$  increases from 1 to 2 sources, the loss probabilities drop by an order of magnitude. As  $N$  approaches 10, the probability of buffering beyond 100 ms, i.e., three frame periods at 30 frames/s becomes  $10^{-5}$ . Fig. 17 shows the same for a constant buffer size and varying  $\rho$ . The reduction in average queue size due to multiplexing is shown in Fig. 18.

The low loss probabilities shown above, demonstrate that statistical multiplexing of variable bit rate video coders will not exhibit perceptible quality variations in video reception. Thus, it is a viable networking alternative to the use of multimode video coders which maintain a synchronous transmission at the expense of variable reception quality.

#### IV. CONCLUSIONS

Variable bit rate video coding can take advantage of dynamic bandwidth sharing via statistical multiplexing and packet switching. In this paper, we have presented models and results that assess the performance of statistical multiplexing of independent video sources. Our results indicate that the probability of buffering (or delaying) video data beyond an acceptable limit drops dramatically as the number of multiplexed sources increases beyond one. This demonstrates that statistical or asynchronous time division (ATD) multiplexing can efficiently absorb temporal variations of the bit rate of individual sources without the significant variations in reception quality exhibited by multimode videocoders for synchronous TDM or circuit switched transmission.

We have presented two source models. Model *A* is an autoregressive continuous-state, discrete-time Markov process, which was used to generate source data in simulation experiments. Model *B* is a discrete-state, continuous-time Markov process that we used in deriving a fluid-flow queueing analysis. Our study showed that they both generated consistent numerical results in terms of queueing performance.

The parameters of the two models were chosen to match experimental measurements. The experiment consisted of a 10 s (300 frames) sequence of a video telephone scene, coded using a conditional replenishment interframe compression algorithm. Both models match the average bit rate and the exponential fit of the autocovariance of the experimental data. The steady-state density of both models has a shape similar to that of the measured histogram of the videocoder bit rate. The experimental results may be limited in statistical significance, especially given the short duration of the moving scene. Nevertheless, we trust the structure of our models as it agrees with previous experimental results reported in [3] for a sequence that lasted 1 h, and recent experimental evidence

communicated to us in [16]. Specific parameters depend on the particular coding scheme, the nature of the moving scene, and the confidence of the measured statistics. Our source and queueing models can be tuned to other experimental results.

As our experiments indicate, the required instantaneous bit rate for constant image quality can vary from extremely low values to peak values that can reach four times the average bit rate as in our 10 s sequence in Fig. 3. The correlation of the bit rate between frames is high for a lag of a few frames, but is negligible for lags greater than 30 frames or 1 s. With dedicated fixed bandwidth channels, e.g., synchronous TDM and circuit switching, the required image quality can be maintained if the bandwidth is equal to the peak bit rate, which could be wasteful. Thus, to avoid the waste, a constant rate coder will have to sacrifice the quality of high activity scenes, e.g., via multimode buffering. We observed such quality degradation by replaying the coded sequence using a multimode coder with the same average bit rate. The most serious degradation was observed at the peak bit rate around frame 217 in Fig. 3. On the other hand, if video is packetized and many sources are present, then buffering can be provided by the network itself. In that case, the coders need not adjust their bit rate by varying the image quality. Due to the statistical smoothing in the network common buffers, peak rates can be transmitted. Occasionally, due to buffering delays, some packets may be lost.

There are two research areas stemming from the problem of packet loss. The first is the development of robust coding schemes that tolerate loss. Some preliminary results in this area were recently presented at the First International Packet Video Workshop, e.g., [23], [24], [25]. The second area concerns network performance and design in order to assess and minimize such loss. This paper has addressed these networking issues in the simple case of a statistical multiplexer. Our contribution is two-fold:

a) We have used queueing theory tools for modeling video sources, and for analyzing and simulating queues that multiplex them. These tools were tuned to a particular coding scheme and moving image sequence, but can be adapted to model other video coding scenarios.

b) For the parameters measured in a short video sequence, our analysis demonstrated that probabilities of loss in a statistical multiplexer can be reduced by orders of magnitude as the number of multiplexed sources increases from two to ten. Thus, bit rate variations of the videocoder outputs will be smoothed out without deterioration of the image quality, except possibly during the rare instances of packet loss.

We have considered only statistics of a talking person. In a two-way video telephone conversation each participant alternates between talking and listening. Consequently, the one-way image sequence alternates between high and low activity scenes, affecting the bit rate in a way reminiscent of the conversational voice variations that motivated time assigned speech interpolation (TASI) systems. In this case, the statistical smoothing of packetized video transmission is expected to be more pronounced than that which was reported in this paper.

Given the attractive features of packetized video that we reported, its feasibility will be determined by advances in robust video coding, and by the development of flow control protocols for broad-band fast packet switched networks. A packet video network will introduce losses, both due to transmission errors, and to buffering delays. Coding schemes should tolerate some loss and flow control could try to prevent losses due to congestion. Flow control protocols are plausible mechanisms in a local area environment but tougher to implement in public broad-band networks. Their function would be to provide sources with timely feedback alerting them of potential congestion in the network. In such a case, sources may switch to a higher compression mode to alleviate

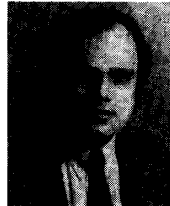
a temporary threat. Thus, variations in video reception quality may not be completely avoided with packetized transmission. It is expected, however, that they will occur in rare instances, resulting in high network utilization and reception quality much superior to fixed speed videocoders at the same average utilization.

## ACKNOWLEDGMENT

The authors wish to thank F. Dix for providing the video sequence data and for recording the encoded data using the equipment in Bell Communications Research, Navesink. Thanks are also due to Dr. C. Judice, Dr. D. LeGal, T.-C. Tsai and Prof. T. Stern for helpful conversations. This work was partially performed at Bell Communications Research within a summer assignment of D. Anastassiou and G. Karlsson.

## REFERENCES

- [1] A. N. Netravali and J. O. Limb, "Picture coding: A review," *Proc. IEEE*, vol. 68, pp. 366-406, Mar. 1980.
- [2] T. Berger, *Rate Distortion Theory, a Mathematical Basis for Data Compression*. Englewood Cliffs, NJ: Prentice Hall, 1971.
- [3] B. G. Haskell, "Buffer and channel sharing by several interframe picturephone coders," *Bell Syst. Tech. J.*, vol. 51, no. 1, pp. 261-289, Jan. 1972.
- [4] J. O. Limb, "Buffering of data generated by the coding of moving images," *Bell Syst. Tech. J.*, pp. 239-255, Jan. 1972.
- [5] W. Verbiest, "Video coding in an ATD environment," presented at Proc. Third Int. Conf. New Syst. Services Telecommun., Liege, Belgium, Nov. 1986.
- [6] A. A. Lazar, A. Patir, T. Takahashi, and M. El Zarki, "MAGNET: Columbia's integrated network testbed," *IEEE J. Select. Areas Commun.*, vol. SAC-3, pp. 859-871.
- [7] A. A. Lazar and J. S. White, "Packetized Video on MAGNET," presented at Proc. SPIE—The Int. Society Opt. Eng., vol. 707, Cambridge, MA, Sept. 1986.
- [8] J. S. Turner and L. F. Wyatt, "A packet network architecture for integrated services," in *Proc. GLOBECOM '83*, San Diego, CA, Dec. 1983, pp. 2.1.1-6.
- [9] T. Stern, "A queueing analysis for packetized voice," in *Proc. GLOBECOM '83*, San Diego, CA, Dec. 1983, pp. 2.5.1-2.5.6.
- [10] H. Hefes and D. M. Lucantoni, "A Markov characterization of packetized voice and data traffic and related statistical multiplexer performance," *IEEE J. Select. Areas Commun.*, vol. SAC-4, pp. 856-868, Sept. 1986.
- [11] K. Sriram and W. Whitt, "Characterizing superposition arrival processes in packet multiplexers for voice and data," *IEEE J. Select. Areas Commun.*, vol. SAC-4, pp. 833-846, Sept. 1986.
- [12] J. N. Daigle and J. D. Langford, "Models for analysis of packet voice communications systems," *IEEE J. Select. Areas Commun.*, vol. SAC-4, pp. 847-855, Sept. 1986.
- [13] K. Koga, Y. Iijima, I. Kazumoto, and T. Ishiguro, "Statistical Performance Analysis of an Interframe Encoder for Broadcast Television Signals," *IEEE Trans. Commun.*, vol. COM-29, pp. 1868-1875, Dec. 1981.
- [14] B. G. Haskell, F. W. Mounts, and J. C. Candy, "Interframe coding of videophone pictures," *Proc. IEEE*, vol. 60, pp. 792-800, July 1972.
- [15] L. Kleinrock, *Queueing Systems, Vol. 2*. New York: Wiley, 1976.
- [16] W. Verbiest, private communication, July 1987.
- [17] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York: McGraw Hill, 1984.
- [18] N. U. Prabhu, "Time-dependent results in storage theory," *J. Appl. Prob.*, vol. 1, pp. 1-46, 1964.
- [19] L. Kleinrock, *Queueing Systems, Vol. 1*. New York: Wiley, 1975.
- [20] D. Anick, D. Mitra and M. M. Sondi, "Stochastic theory of a data-handling system with multiple sources," *Bell Syst. Tech. J.*, vol. 61, no. 8, pp. 1871-1894, Oct. 1982.
- [21] L. Kosten, "Stochastic theory of a multi-entry buffer," *Delft Prog. Rep.*, vol. 1, pp. 10-18, 1974.
- [22] R. Bellman, *Introduction to Matrix Analysis*. New York: McGraw-Hill, 1970.
- [23] M. Vetterli and G. Karlsson, "Sub-band coding of video for packet networks," presented at the First Int. Packet Video Workshop, Columbia Univ., New York, NY, May 1987.
- [24] N. Ohta, "Variable rate video transmission in asynchronous transfer mode networks," presented at the First Int. Packet Video Workshop, Columbia Univ., New York, NY, May 1987.



**Basil Maglaris (S'74-M'79)** was born in Athens, Greece, in 1952. He received the undergraduate Diploma degree in electrical engineering from the National Technical University of Athens in 1974, the M.Sc. degree from the Polytechnic Institute of Brooklyn, Brooklyn, NY, in 1975, and the Ph.D. degree in electrical engineering and computer science from Columbia University, New York, NY, in 1979.

From 1979 to 1981 he was with the Network Analysis Corporation, Great Neck, NY, where he was involved in several projects in data and voice networks for both government and industry. In 1981 he joined the Polytechnic Institute of New York, Brooklyn, where he is currently Associate Professor of Electrical Engineering and Computer Science. His research interests focus on the analysis, performance evaluation, and optimization of data, voice and integrated networks, packet radio, and local area networks.

Dr. Maglaris has been involved in various professional activities with the IEEE and the ACM.



**Dimitris Anastassiou (S'77-M'78)** was born in Athens, Greece, in 1952. He received the Diploma degree in electrical engineering from the National Technical University of Athens in 1974, and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, in 1975 and 1979, respectively.

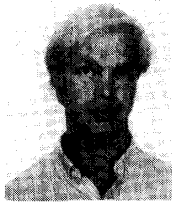
From 1978 to 1983 he worked at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, as a Research Staff Member, doing research on various aspects of image processing and coding. Since 1983 he has been with the Department of Electrical Engineering of Columbia University, New York, NY, where he is currently an Associate Professor. His research interests include digital image processing, coding and understanding, digital signal processing, and VLSI architectures for image processing.

Dr. Anastassiou has received an IBM Outstanding Innovation Award for his contributions to the development of the IBM videoconferencing system, and a 1985 NSF Presidential Young Investigator Award.



**Prodip Sen (S'75-M'81)** was born in Amritsar, India, in 1951. He received the B.Tech. degree in electrical engineering from The Indian Institute of Technology, Bombay, India, in 1973, and the Ph.D. degree in electrical engineering from the Indian Institute of Science, Bangalore, India, in 1977.

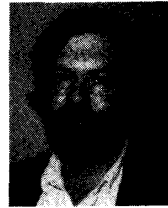
He was a Research Fellow and Visiting Assistant Professor at the System Science Department in the School of Engineering at UCLA, Los Angeles, CA, from 1978 to 1980. From 1980 to 1983 he was a Research Mathematician at Polysystems Analysis Corporation, Huntington, NY. Since 1983 he has been with Polytechnic University, Brooklyn, NY, where he is currently Assistant Professor of Electrical Engineering. His research interests are in the areas of analysis and design of packet radio networks, local area networks, and metropolitan area networks.



**Gunnar Karlsson** was born in Jönköping, Sweden. He received the M.S. degree in electrical engineering from Chalmers University of Technology, Gothenburg, Sweden, in 1983. He is currently working towards the Ph.D. degree in electrical engineering at Columbia University, New York, NY.

During the academic year 1982-1983 he was studying at the University of Massachusetts, Amherst, on a Fulbright Scholarship. He did the Master's thesis at Telefonaktiebolaget LM Ericsson, Stockholm, Sweden, during the Summer 1983.

In 1984 he was working as a hardware designer at Saab Training Systems in Huskvarna, Sweden. During the summers in 1985 and 1986 he did research at Bell Communications Research, Morristown, NJ. Since 1985 he has held positions as Teaching Assistant at the Department of Electrical Engineering and Research Assistant at the Center for Telecommunications Research at Columbia University.



**John D. Robbins** received the B.S.E.E. degree and the M.S.C.S. degree from Rutgers University, NJ, in 1976 and 1980, respectively.

He joined the research area of Bell Telephone Laboratories in 1976, where he worked on a variety of problems in image and video processing. In 1983, he joined Bell Communications Research as District Manager of Visual Communications Research, where he was responsible for projects addressing the integration of visual communication. In 1987, he joined the Technical Staff at Bell

Communications Research to further pursue problems in image and video processing. He has published 15 papers in the area of visual communication and holds six patents in video data compression and interpolation.