

A Traffic Model for MPEG-Coded VBR Streams*

Marwan Krunz[†] and Herman Hughes[‡]

[†] Department of Electrical Engineering

[‡] Department of Computer Science

Michigan State University

East Lansing, MI 48824

e-mail: [krunz,hughes]@cps.msu.edu

Abstract

Compression of digital video is the only viable means to transport real-time full-motion video over BISDN/ATM networks. Traffic streams generated by video compressors exhibit complicated patterns which vary from one compression scheme to another. In this paper we investigate the traffic characteristics of video streams which are compressed based on the MPEG standard. Our study is based on 23 minutes of video obtained from an entertainment movie. A particular significance of our data is that it contains all types of coded frames, namely: Intra-coded (I), Prediction (P), and Bidirectional (B) MPEG frames. We describe the statistical behavior of the VBR stream using histograms and autocorrelation functions. A procedure is developed to determine the instants of a scene change based on the changes in the size of successive *I* frames. It is found that the length of a scene can be modeled by a geometric distribution.

A model for an MPEG traffic source is developed in which frames are generated according to the compression pattern of the captured video stream. For each frame type, the number of cells per frame is fitted by a lognormal distribution whose parameters are determined by the frame type. The appropriateness and limitations of the model are examined by studying the multiplexing performance of MPEG streams. Simulations of an ATM multiplexer are conducted, in which traffic sources are derived from the measured VBR trace as well as the proposed model. The queueing performance in both cases is found to be relatively close.

1 Introduction

Future BISDN/ATM networks promise to provide the means to transport diverse traffic streams. These streams vary in their traffic characteristics and performance requirements. The largest proportion of the bandwidth in BISDN/ATM networks is expected to carry video traffic. This is due to the introduction of many new video/multimedia services and also to the large amount of bandwidth needed to transport

real-time digital video. The huge link capacity provided by optical fibers can be quickly saturated by few video streams transmitted in their *uncompressed* digital format. To increase the number of video streams that can be simultaneously carried over the link, compression schemes are employed in order to reduce the amount of bits contained in video frames. The size of a compressed video frame varies depending on the scene activity and the type of compression involved. Hence, the output of a video codec is a VBR stream. Unlike conventional data networks, ATM networks provide efficient support for VBR traffic.

The focus of this paper is on a particular compression algorithm which has recently gained considerable attention, namely the Motion Picture Experts Group (MPEG) standard (see [1] for an overview of MPEG). Among other compression standards, MPEG is being the focus of our study for two reasons: (1) it has been standardized by CCITT and (2) it involves several types of compression that are also used by other compression algorithms. This last reason makes MPEG a generalization of many existing compression algorithms. The main objective of this paper is to develop a traffic model for an MPEG-coded video stream.

Several traffic models have been proposed to characterize compressed video streams [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] (also see [14] for a survey). The parameters of these models were obtained by matching certain statistical characteristics of an actual video sequence and the model under consideration. Particular emphasis was on matching the correlation structure of the bits-per-frame sequence, since correlations are known to have a great impact on the queueing performance of a statistical multiplexer [15]. Correlations arise naturally in compressed video traffic and their patterns are largely determined by the type of video (e.g., videophone, motion-picture, teleconferencing) as well as by the compression algorithm used. In [2] the authors proposed two traffic models for the frame-size sequence of a video-phone stream compressed using a conditional replenishment algorithm. The first model is a first-order autoregressive process, while the second is a fluid model. A more elaborate model based on an *autoregressive-moving average* (ARMA) process was proposed in [4]. Skelly et al. [8] introduced a histogram-based traffic model using a quasi-static approximation. In [9] the authors used the Transform-Expand-Sample (TES) approach developed in [5] to model a video stream that was generated using a DCT, but with no differential or motion compensation. Heyman *et al.* [6] analyzed the VBR trace for a 30-minute long video-teleconferencing sequence and suggested that the number of cells per frame approximately follows a gamma distribution. A sophisti-

*This research was partially supported by the NSF grant # NCR 9305122.

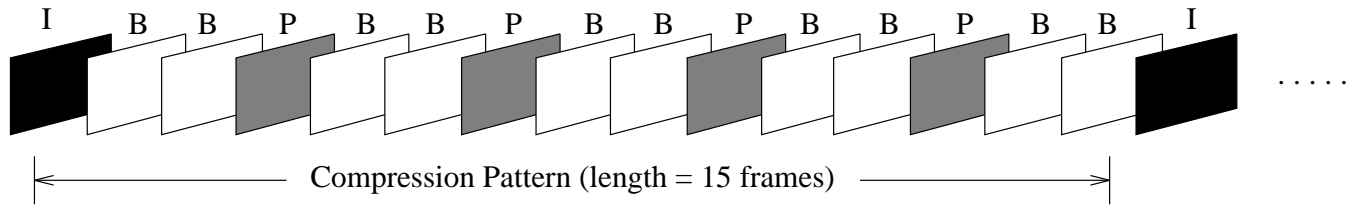


Figure 1: The compression pattern used to generate the video stream.

ated autoregressive model was proposed in [7], which consists of the sum of two AR(1) processes and a Markov chain. Pancha and El Zarki [16] described the statistical characteristics of a few minutes of full-motion video generated by an MPEG encoder with no Bidirectional (B) frames. None of these models can be used to characterize video streams generated by MPEG encoders. The main reason is that MPEG allows several modes of compression which, altogether, result in a quite different traffic pattern than the patterns observed in previous studies.

In this paper, we use a 23-minute long video sequence to study the statistical characteristics of MPEG traffic and to develop a traffic model for an MPEG-coded video stream. The sequence was taken from the movie *The Wizard of Oz*, and was digitized and encoded using a public domain software MPEG encoder developed by the Berkeley Plateau Research Group. The procedure which was used to obtain the compressed video sequence is described in the next section.

2 The Compression Setup

The experimental data (i.e., the sequence of bits per frame in the compressed video stream) was obtained via a two-stage approach. In the first stage, video frames are digitized and stored on magnetic tape. A computer program was written to capture and digitize video frames. The program runs in a workstation connected to the laser disk player. Using a *SunVideo* board installed in the workstation, the program receives analog frames and digitizes them one at a time. It then sends control signals to the laser player to position the head at the next frame and repeat the process. Frames are converted to 4:1:1 YUV format (the input format for MPEG encoders) and are written to hard disk. After accumulating some number of frames, these frames are moved from the hard disk to magnetic tape. The process continues until the entire stream is stored on tape. In the second stage, the frames are read from tape and compressed using the Berkeley MPEG encoder. The frame-size data are extracted during the encoding process.

Compression in MPEG is based on exploiting spatial as well as temporal data redundancies. Frames can be coded as *I*, *P*, or *B* frames (intra-frame, predictive, or bidirectional, respectively). Frames of type *I* are compressed using DCT only, while *B* and *P* frames involve motion compensation (prediction and interpolation) as well. Details of the MPEG standard can be found elsewhere (see [1]). The compression pattern used to encode the examined video stream is shown in Figure 1.

3 Statistical Characteristics of MPEG Streams

3.1 Frame Size Measurements

Based on the previous setup, 41760 video frames were compressed and their sizes were recorded. At 30 frames/sec, this

represents about 23 minutes of real-time full-motion video.

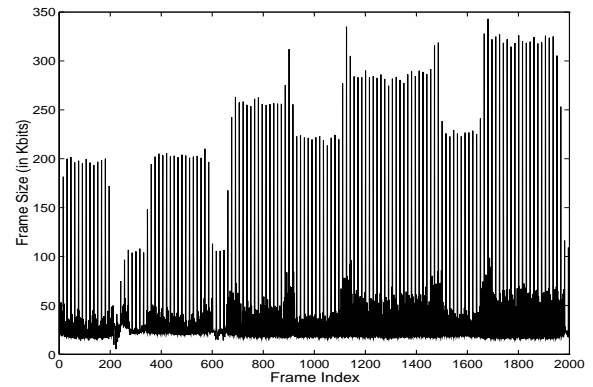


Figure 2: The frame size sequence (frames 1 to 2000).

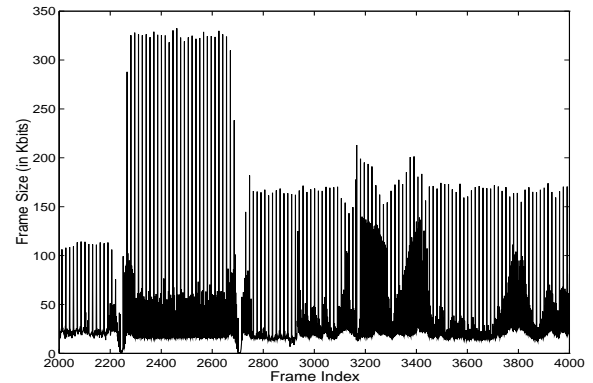


Figure 3: The frame size sequence (frames 2001 to 4000).

Portions of the frame-size sequence are plotted in Figures 2 and 3. In both figures, the larger frame sizes (which are depicted by the lightly shaded areas) correspond to *I* frames. The darker parts of the figures correspond to *P* and *B* frames which are smaller in size than *I* frames. It is difficult, however, to distinguish *P* frames from *B* frames in these figures. Sample statistics computed from these data are shown in Table 1.

As we demonstrate later in this paper, an MPEG traffic stream exhibits a complicated correlation structure. The complexity of this structure is the result of having three types of frames in one stream. A model for MPEG traffic can be facilitated by decomposing the VBR sequence into three separate subsequences, each consisting of frames of the same type. Traffic modeling can then be applied to each subsequence. Sample paths from these subsequences are shown in Figures 4-6.

Statistic	All Frames	<i>I</i> Frames	<i>P</i> Frames	<i>B</i> Frames
Number	41760	2784	11136	27840
Sample Mean (in Kbits)	41.7	197.1	58.0	19.6
Sample Standard Deviation (in Kbits)	51.7	63.8	37.3	5.7
Maximum (in Kbits)	343.1	343.1	284.6	60.0
Minimum (in Kbits)	0.56	36.2	0.56	0.57

Table 1: Summary statistic for the frame-size sequence.

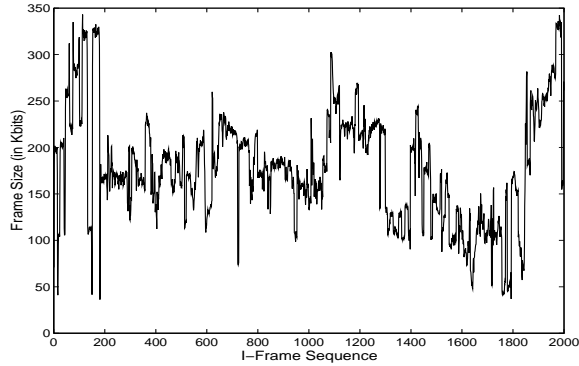


Figure 4: Sample path from the *I* frames subsequence.

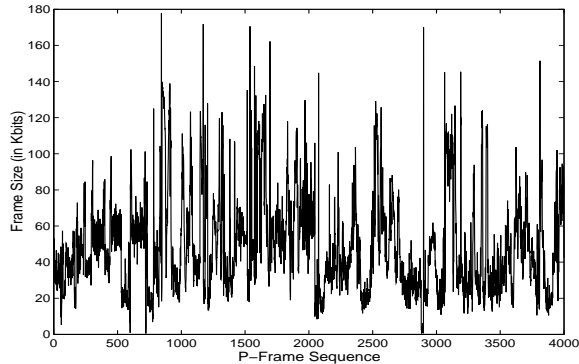


Figure 5: Sample path from the *P* frames subsequence.

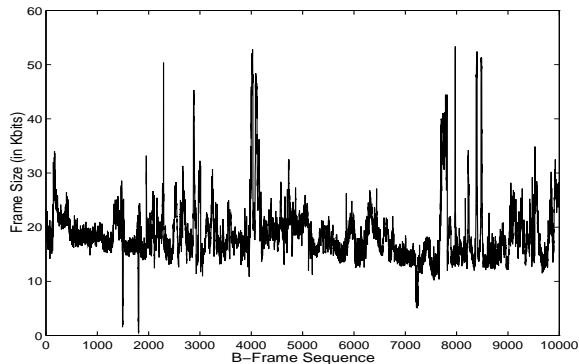


Figure 6: Sample path from the *B* frames subsequence.

3.2 Scene-Length Distribution

From Figures 2 and 3 it is observed that the video stream consists of several segments such that the sizes of *I* frames in each segment are close in value. Intuitively, one expects that each segment corresponds to a part of the movie with no *abrupt* view shifts. We verified this intuition by observing the movie. Each of these segments is referred to as a *scene*. Note that only an abrupt change of the view accounts for the beginning of a new scene. Camera panning and zooming are not considered as scene boundaries.

To model the length of a scene, we watched the movie and measured the length of all the scenes (in seconds). This approach has the disadvantage of requiring the actual movie to be displayed on screen. Another heuristic approach was also used which (compared to the visually-measured values) proved to be quite accurate. In this heuristic we use the fact that a ‘sufficient’ change in the size of consecutive *I* frames is a strong indication of the start of a new scene. Thus, one can use the frame-size subsequence of *I* frames to obtain the scene-length data. We shall assume that the minimum length of a scene is one second (i.e., two consecutive *I* frames). Let $\{Z_I(n) : n = 1, 2, \dots\}$ be the *I* frames subsequence. Suppose we are in the *i*th scene that started with the *k*th *I* frame. The $(n + k + 1)$ th *I* frame of the sequence indicates the start of the $(i + 1)$ th scene if

$$\frac{|Z_I(n + k + 1) - Z_I(n + k)|}{\left(\sum_{j=k}^{n+k} Z_I(j)\right) / n} > T_1$$

and

$$\frac{|Z_I(n + k + 2) - Z_I(n + k)|}{\left(\sum_{j=k}^{n+k} Z_I(j)\right) / n} > T_2$$

where T_1 and T_2 are two thresholds. A similar approach based on a different heuristic was used in [9] to obtain scene lengths.

The histogram for the scene length (in *I* frames) is shown in Figure 7 using $T_1 = 0.05$ and $T_2 = 0.1$. The figure also depicts a geometric *pmf* (with mean = 10.5 *I* frames) fitted to the histogram. A Q-Q plot of the experimental distribution versus a geometric distribution is shown in Figure 8. It is clear that the length of a scene can be adequately modeled using a geometric distribution. We also tested the hypothesis of independence between scene lengths, based on the runs-up test [17]. At a 95% level of confidence, the hypothesis of independence cannot be rejected.

3.3 The Autocorrelation Structure

Correlations in video data arise as a consequence of visual similarities between consecutive images (or parts of images) in a video stream. While compression of video results in a reduction in these correlations, the VBR sequence of a compressed stream still contains considerable amounts of correlations. This is due to the periodic fashion of applying a given type of compression. For example, consider two

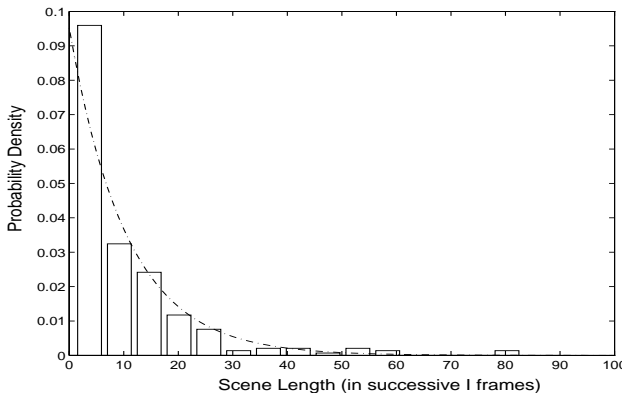


Figure 7: Probability distribution for scene length.

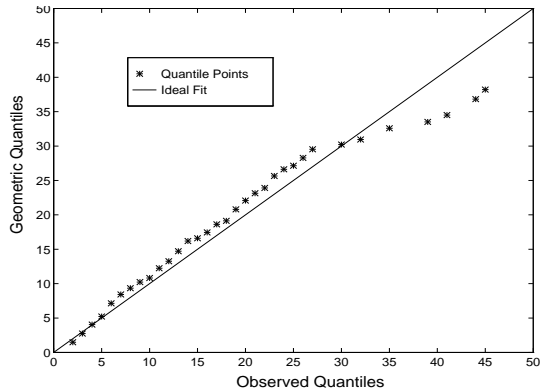


Figure 8: The Q-Q plot for the scene-length distribution.

consecutive frames that correspond to two similar images. Applying DCT to these frames (one at a time) will result in two compressed frames with similar sizes (i.e. highly correlated frame sizes).

Two types of correlations can be found in compressed video streams: intra-frame and inter-frame correlations. Intra-frame correlations appear when intra-frame compression (e.g., DCT coding) is involved and the VBR sequence is studied at a smaller level than a frame (e.g., a slice level) [9]. Inter-frame correlations are observed when the VBR stream is studied at the frame level, as in the case of our study.

In several previous studies of video traffic, researchers observed that the autocorrelation function for the frame-size sequence of the examined compressed video stream has a negative-exponential shape [2, 18, 6, 9]. A different shape of this function should be expected in MPEG-coded streams. The reason is that MPEG defines three different types of frames which are generated according to a compression pattern. The compression pattern is repeated until all the frames in the stream are compressed. This results in persistent periodicities in the inter-frame autocorrelation function. The autocorrelation function for the frame-size sequence of our video stream is shown in Figure 9.

Although the VBR sequence is being studied at a frame level, the correlation structure for the sequence is quite complicated and it depicts strong and pseudo-periodic correlations. Two apparent periodic components can be seen in Figure 9: one at lags of multiples of 15, which is due to inter-frame correlations between I frames only, while the other is at lags of multiples of 3, which is due to correlations

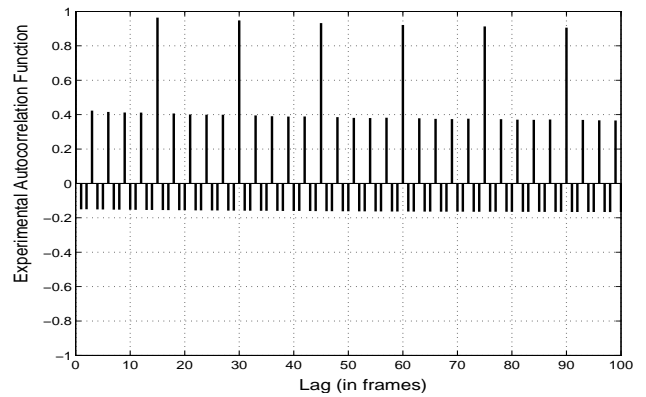


Figure 9: The autocorrelation function for the frame size of the VBR sequence.

between P frames. Both components are bounded by exponentially decaying envelopes with large time constants (i.e., slowly decaying). This behavior can be better illustrated by computing the autocorrelation function for the individual I frames subsequence (similarly, P frames and B frames subsequences) as shown in Figures 10-12.

4 Modeling the VBR Stream

4.1 Frame Size Distribution

The frame-size histogram based on the complete VBR stream is shown in Figure 13. It has the general shape of a Pearson type V density function. We will not attempt, however, to develop a model based on a distribution for all frames since the impact of the frame type would not be captured in a such model. It is more appropriate to study the frame size distribution for each frame type. The frame-size histograms for the I , P , and B subsequences are shown in Figures 14- 16.

Three probability density functions (*pdfs*) are examined to determine the best fit to the experimental histograms. These are Gamma, Weibull, and lognormal *pdfs*. The *maximum-likelihood estimators* (MLEs) for the parameters of these functions are used to obtain the fitted distributions shown in Figures 14-16. Note that the amplitudes in each histogram are normalized so that the total area under the histogram is one.



Figure 10: Correlations in the I -frames subsequence.

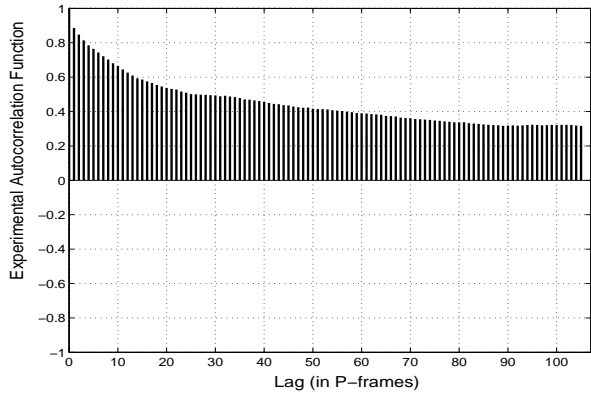


Figure 11: Correlations in the P -frames subsequence.

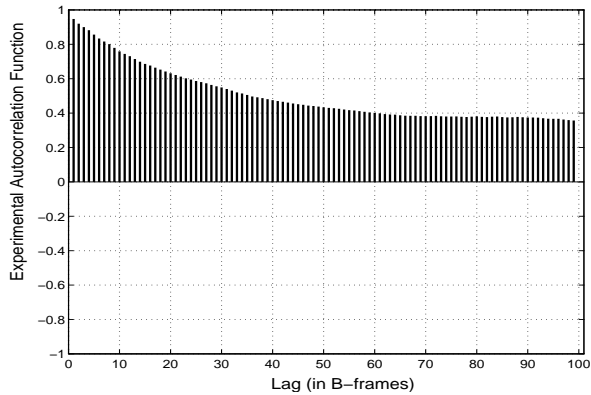


Figure 12: Correlations in the B -frames subsequence.

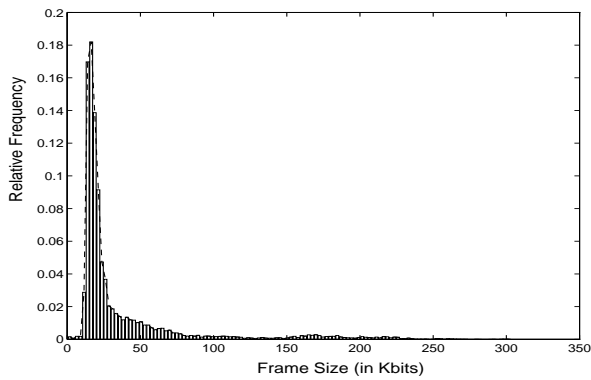


Figure 13: Histogram for the frame size of the VBR sequence.

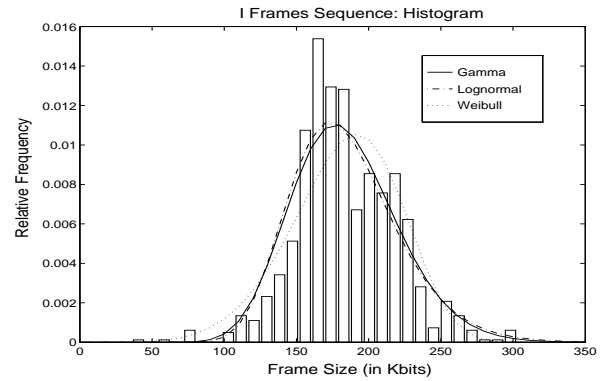


Figure 14: Histogram and fitting density functions for the I -frames subsequence.

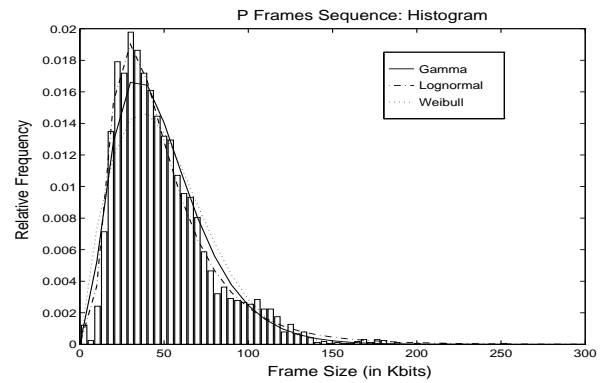


Figure 15: Histogram and fitting density functions for the P -frames subsequence.

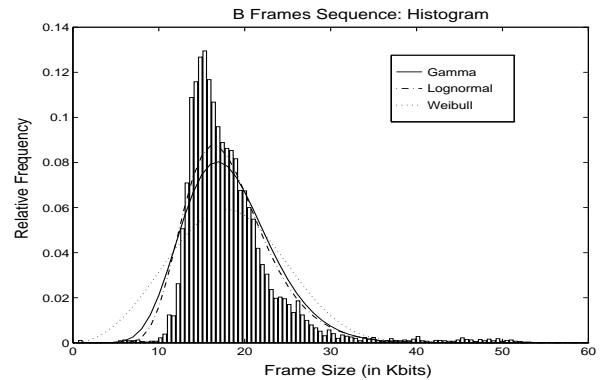


Figure 16: Histogram and fitting density functions for the B -frames subsequence.

4.2 Video Traffic Model

From Figures 14-16 it is concluded that for each of the three frame-size subsequences, a lognormal *pdf* provides the best fit for the corresponding histogram. Hence, we choose the lognormal distribution as the basis for our model. The *pdf* for this distribution is given by:

$$f(z) = \begin{cases} \frac{1}{z\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(\ln z - \mu)^2}{2\sigma^2}\right], & z > 0 \\ 0, & \text{otherwise} \end{cases}$$

This function is parameterized by a shape parameter $\sigma > 0$, and scale parameter $\mu \in (-\infty, \infty)$.

In our model, we assume that a video stream consists of three types of frames: *I*, *P*, and *B*. Frame types are generated according to the compression pattern in Figure 1. The pattern is continuously repeated until the end of the stream. The number of cells in a frame is assumed to follow a lognormal distribution with MLE parameters which are determined by the frame type. These parameters are given Table 2.

Frame Type	lognormal	
	μ	σ
<i>I</i>	5.1968	0.2016
<i>P</i>	3.7380	0.5961
<i>B</i>	2.8687	0.2675

Table 2: The MLE parameters of fitting distributions.

The effect of scene changes is incorporated in the model as follows: each stream is assumed to consist of several scenes. Scene lengths constitute a sequence of *iid* random variables with a geometric distribution (in units of *I* frames). The size of the first *I* frame in each scene is sampled from a lognormal *pdf*. Consecutive *I* frames in the same scene have *exactly* the same size of the first *I* frame. The same procedure is applied to each scene. The fitting lognormal distribution for *I* frames must be adjusted so that only one *I* frame from each scene is used in the fitting. We use the average size of *I* frames in each scene to obtain the adjusted histogram for the size of *I* frames (generated once at the start of each scene). This histogram and the corresponding candidate fits (based on MLEs) are shown in Figure 17. The MLE for the lognormal function in Figure 17 are $\mu = 6.0188$ and $\sigma = 0.4556$.

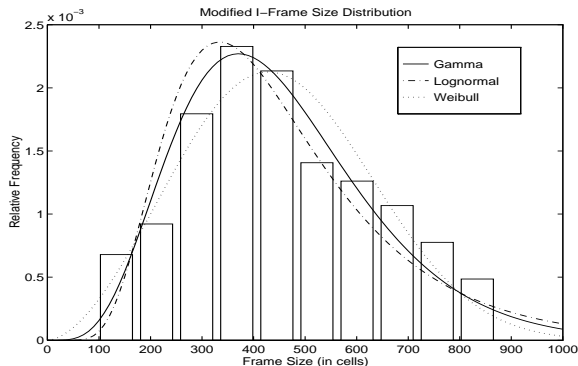


Figure 17: The adjusted histogram and fitting density functions for the *I* frames subsequence.

This model is used in some of the simulation experiments presented in the next section.

5 Multiplexing of MPEG Video Sources

In this section, simulation experiments are conducted to study the performance of an ATM multiplexer for MPEG streams. The multiplexer is modeled as a finite capacity queueing system with buffer size B (in cells) and one server with service rate C . A FIFO service discipline is assumed. The input to the multiplexer consists of MPEG-coded video streams. Two types of simulation experiments are conducted. The first type is trace-driven (i.e., using the actual experimental VBR stream). The second type of experiments is based on the proposed traffic model. In either case, a video source consists of a large number of frames arranged according to the compression pattern in Figure 1. Bits in each frame are packetized into ATM cells (with a 5-byte header added to each cell). Cells are evenly distributed over the period of a frame (see Figure 18).

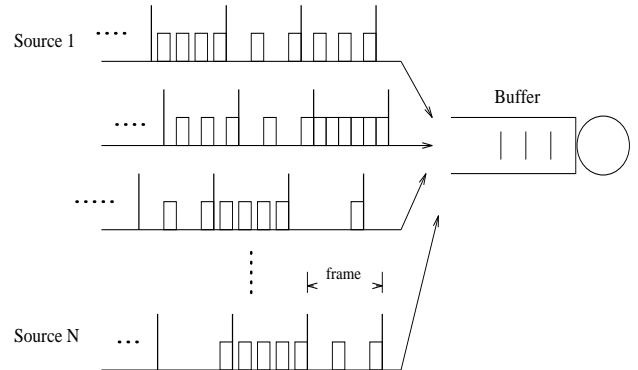


Figure 18: A multiplexer for MPEG streams.

In order to obtain multiple streams for the trace-driven simulations, we assume that the frame-size sequence of the measured VBR stream constitute a homogeneous stochastic process. Following the approach of Heyman *et al.* [6], the measured trace is arranged as a circular list by connecting its ends. To obtain each of the multiplexed streams in the trace-driven simulations, a starting frame is randomly selected from the circular list. Thereafter, the remaining frames are selected sequentially following the starting frame. This process is similar to a random shift with rotation in sequential lists. The procedure is repeated until all the streams are obtained. To eliminate synchronization effects, the starting times for the video streams are selected randomly from a uniform probability distribution (over a frame period).

5.1 Trace-Driven Simulations

In the following experiments, the performance of an ATM multiplexer for MPEG streams is studied via trace-driven simulations. The service rate for the multiplexer is adjusted to obtain a desired level of utilization, U . We define U as the ratio of the aggregate arrival rate from all multiplexed sources to the service rate. In Figure 19 the average cell loss rate, P_{avg} , is shown versus the buffer size, B , using different numbers of multiplexed sources: $N = 1$ (no multiplexing) and $N = 10$. The results are shown using two levels of utilization ($U = 60\%$ and $U = 80\%$).

Observe that P_{avg} for $N = 1$ is surprisingly large under relatively light load ($U = 60\%$) and large buffer size ($B = 600$). This is true despite the fact that each stream is smoothed by spacing cells over the frame. Another important observation is the improvement in the cell loss perfor-

Buffer Size	$U = 60\%$			$U = 80\%$		
	P_{avg}	P_{max}	P_{min}	P_{avg}	P_{max}	P_{min}
100	6.99E-4	1.05E-3	3.98E-4	2.29E-2	5.64E-2	7.04E-3
200	1.62E-4	3.14E-4	8.81E-5	7.83E-3	1.66E-2	4.04E-3
300	3.04E-5	6.30E-5	1.63E-5	3.85E-3	6.86E-3	1.24E-3
400	5.80E-6	1.30E-5	2.20E-6	2.05E-3	4.18E-3	8.59E-4
500	1.80E-6	5.05E-6	2.20E-7*	8.32E-4	2.60E-3	6.05E-4
600	0*	0*	0*	5.47E-4	9.11E-3	2.62E-4

(* loss rates are less accurate due to small or zero number of lost cells)

Table 3: The upper and lower limits of cell loss performance for multiplexed streams ($N = 10$).

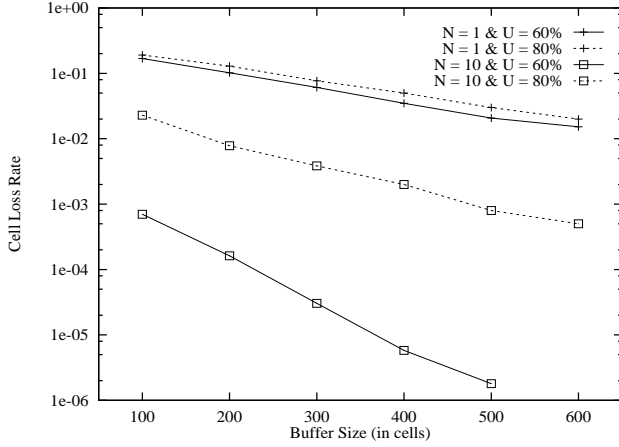


Figure 19: The average cell loss rate versus buffer size.

mance when video streams are multiplexed. When $N = 10$, P_{avg} is the average cell loss rate for the ten sources. Not all sources experience this rate. The maximum (P_{max}) and minimum (P_{min}) loss rates among the ten sources are shown in Table 3, for $U = 60\%$ and $U = 80\%$ and several values of B . These simulations were conducted using the whole VBR trace (i.e., 41760 frames). Each stream generated about $4.5E+6$ cells. The average number of cells per frame for the various frame types is given in Table 4.

Type	Cells/Frame
All	109.02
I	513.88
P	151.63
B	51.49

Table 4: Average frame size in cells.

Figure 20 depicts the average cell loss rate versus the link utilization using different N and $B = 150$. It is evident that the loss rate drops significantly when N increases at a fixed U . Note that at certain values of U and N no cell losses occurred in the simulations. The figure clearly indicates that a considerable gain can be obtained by multiplexing VBR video streams. We define the following quantity which will be referred to as the *multiplexing gain*:

$$\mu_p(n) \triangleq \frac{n \times \text{Utilization at } N = 1 \text{ and } P_{avg} = p}{\text{Utilization at } N = n \text{ and } P_{avg} = p} \quad (1)$$

For a given P_{avg} and N , the utilization can be obtained graphically from Figure 20. Accordingly, $\mu_p(n)$ is found for various n and for $P_{avg} = 10^{-2}$ and $P_{avg} = 10^{-3}$ as shown in Table 5.

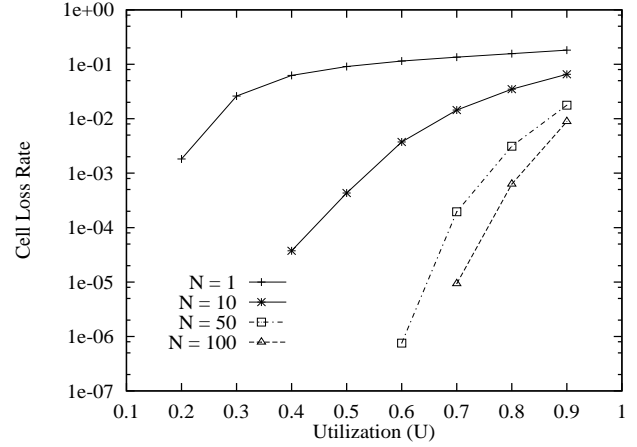


Figure 20: The average cell loss rate versus utilization for various number of multiplexed sources.

n	$\mu_p(n)$	
	$p = 10^{-2}$	$p = 10^{-3}$
1	1	1
10	4.1	3.4
50	16.3	11.7
100	31.1	21.4

Table 5: The multiplexing gain for different numbers of multiplexed sources.

5.2 Model-Based Simulations

In this section, simulation experiments are conducted using computer-generated traffic streams based on the proposed model. As before, the starting times for the traffic streams are selected randomly over a frame period. Also, in multiplexing video streams, the compression pattern is treated as a circular list of length 15 and the starting frame for each stream is chosen randomly from that list. This is necessary to avoid a situation in which all the sources are nearly synchronized (with respect to the type of the frame entering the multiplexer). Otherwise, I frames from all sources would arrive together within a short time interval, thus causing many cell losses due to buffer overflow. The number of generated frames per source is 15000 for every simulation run.

In Figure 21, the average cell loss rate is shown for different buffer sizes using both trace-driven and model-based video streams. The results for $N = 1$ and $N = 10$ at fixed utilization ($U = 60\%$) are given. In general, the loss rates for both traffic inputs are sufficiently close. This is particularly true for $N = 1$. For $N = 10$, a slight difference is observed between the model and the trace traffic inputs. The differ-

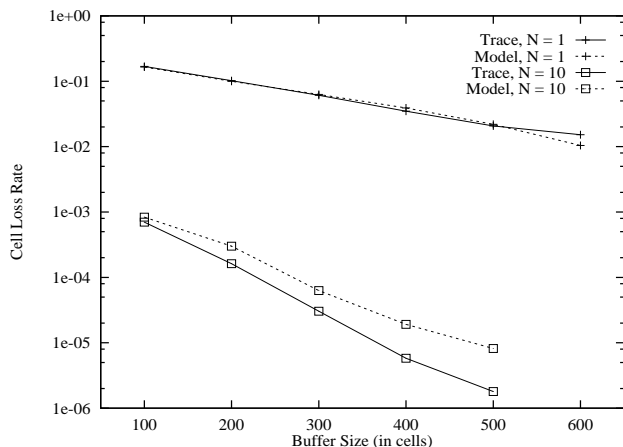


Figure 21: The average cell loss rate versus buffer size using captured VBR trace and model-based input traffic streams ($U = 60\%$, and $N = 1, 10$).

ence increases slowly with larger buffer sizes. This says that we get less accurate performance results from the model as the size of the buffer increases or the number of multiplexed sources gets large. For moderate buffer size (say, around 400 cells) and up to 50 multiplexed sources, the model is considered adequate and its results provide a tight upper bound on the actual performance. Comparisons between trace-driven and model-based simulations at $N = 50$ and $U = 80\%$ are given in Table 6.

B	P_{avg} (Trace)	P_{avg} (Model)
100	4.69E-3	6.69E-3
200	3.23E-3	5.23E-3
300	2.23E-3	4.20E-3
400	1.51E-3	3.51E-3
500	9.89E-4	2.89E-3
600	6.24E-4	9.24E-4

Table 6: Average loss rates at $N = 50$ and $U = 80\%$.

The cell loss rates for each type of frames are shown in Figure ??, using $N = 1$ and $N = 10$, and $U = 60\%$. Again, these results are based on the proposed traffic model.

6 Summary

In this paper the statistical characteristics of an MPEG-coded video stream were studied based on a long trace of actual video data taken from an entertainment movie. A statistical study of the VBR trace indicated that the inter-frame autocorrelation function for the whole frame sequence is pseudo-periodic with particularly strong correlations between I frames. This correlation structure is more complicated than what has been previously reported with regard to frame-level autocorrelations. To simplify the modeling task of MPEG sources, correlations were examined between frames of the same type. The number of bits per frame for each type was fitted to a lognormal distribution. Accordingly, a model for an MPEG source was developed which is based on knowledge of the compression pattern of the actual data, as well as the first two moments.

Simulation experiments were conducted to study the performance of an ATM multiplexer for video sources. The

cell loss performance was reported using two types of input traffic: one from the actual MPEG VBR trace and the other was based on the proposed source model. Results from these simulations indicate that a significant gain in cell loss performance is obtained when VBR video streams are statistically multiplexed. Moreover, the performance obtained using the proposed model was sufficiently close to the one obtained in trace-driven simulations.

References

- [1] D. Le Gall, "MPEG: A video compression standard for multimedia applications," *Communications of the ACM*, vol. 34, pp. 47-58, Apr. 1991.
- [2] B. Maglaris *et al.*, "Performance models of statistical multiplexing in packet video communications," *IEEE Journal on Selected Areas in Communications*, vol. 36, pp. 834-844, July 1988.
- [3] P. Sen, B. Maglaris, N.-E. Rikli, and D. Anastassiou, "Models for packet switching of variable-bit-rate video sources," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 865-869, June 1989.
- [4] R. Grunenfelder, J. P. Cosmos, S. Manthorpe, and A. Odinma-Okafor, "Characterization of video codecs as autoregressive moving average processes and related queueing system performance," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 284-293, Apr. 1991.
- [5] B. Melamed, "TES: A class of methods for generating autocorrelated uniform variates," *ORSA Journal on Computing*, vol. 3, no. 4, pp. 317-329, 1991.
- [6] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconferencing traffic in ATM networks," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 2, pp. 49-59, Mar. 1992.
- [7] G. Ramamurthy and B. Sengupta, "Modeling and analysis of a variable bit rate video multiplexer," in *Proc. of IEEE INFOCOM '92*, vol. 2, pp. 817-827, 1992.
- [8] P. Skelly, M. Schwartz, and S. Dixit, "A histogram-based model for video traffic behavior in an ATM multiplexer," *IEEE/ACM Transactions on Networking*, vol. 1, pp. 446-459, Aug. 1993.
- [9] A. A. Lazar, G. Pacifici, and D. E. Pendarakis, "Modeling video sources for real-time scheduling," Tech. Rep. 324-93-03, Columbia University, Department of Electrical Engineering and Center for Telecommunications Research, Apr. 1993.
- [10] F. Yegengolu, B. Jabbari, and Y.-Q. Zhang, "Motion-classified autoregressive modeling of variable bit rate video," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 3, pp. 42-53, Feb. 1993.
- [11] N. M. Marafih, Y.-Q. Zhang, and R. L. Pickholtz, "Modeling and queueing analysis of variable-bit-rate coded video sources in ATM networks," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 4, pp. 121-128, Apr. 1994.

- [12] A. W. Bragg and W. Chou, "Analytic models and characteristics of video traffic in high speed networks," in *Proceedings of International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS '94)*, (Durham, North Carolina), pp. 68–73, Jan. 31 – Feb. 2 1994.
- [13] S. K. Chan and A. L. Garcia, "Analysis of cell inter-arrival from VBR video codecs," in *Proc. of IEEE INFOCOM '94*, vol. 1, pp. 350–357, 1994.
- [14] V. S. Frost and B. Melamed, "Traffic modeling for telecommunications networks," *IEEE Communications Magazine*, vol. 32, pp. 70–81, Mar. 1994.
- [15] K. Sriram and W. Whitt, "Characterizing superposition arrival processes in packet multiplexers for voice and data," *IEEE Journal on Selected Areas in Communications*, vol. SAC-4, pp. 833–846, Sept. 1986.
- [16] P. Pancha and M. El Zarki, "MPEG coding for variable bit rate video transmission," *IEEE Communications Magazine*, vol. 32, pp. 54–66, May 1994.
- [17] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*. McGraw-Hill, Inc., second ed., 1991.
- [18] D. P. Heyman and T. V. Lakshman, "Source models for VBR broadcast-video traffic," in *Proc. of IEEE INFOCOM '94*, vol. 2, pp. 664–671, 1994.
- [19] D. Reininger *et al.*, "Statistical multiplexing of VBR MPEG compressed video on ATM networks," in *Proc. of IEEE INFOCOM '93*, vol. 3, pp. 919–926, 1993.