INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

MPEG-2 VIDEO TRAFFIC MODELS AND THEIR IMPACT ON NETWORK PERFORMANCE

by

CHRISTOPHER AMO-QUARM

A dissertation submitted to the Graduate Faculty in Engineering in partial fulfillment of the requirements for the degree of Doctor of Philosophy, The City University of New York.

2002

This manuscript has been read and accepted for the Graduate Faculty in Engineering in satisfaction of dissertation requirements for the degree of Doctor of Philosophy.

06/03/02

Date

Chairman of Examination Committee:

Dr. Kaliappa Ravindran

Professor of Computer Science, The City College of the City University of New York

06/03/2002

Date

Executive Officer:

Dr. Mumtaz K. Kassir

Dean of the Graduate Studies at the Engineering School, The City College of The City University of New York.

Dr. Myung J. Lee

Dr. Tarek N. Saadawi

Dr. Kazem Sohraby

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

ABSTRACT

MPEG-2 VIDEO TRAFFIC MODELS AND THEIR IMPACT ON NETWORK PERFORMANCE

by

CHRISTOPHER AMO-QUARM

Adviser: Professor Kaliappa Ravindran

Variable Bit Rate (VBR) video traffic, arising from scene changes, demonstrates long and short term correlation among consecutive frames. Proper traffic models able to capture such traffic characteristics have attracted the attention of researchers in the area of traffic engineering and their impact on network performance. Some of the models have captured scene changes which lead to bursty traffic. Others have captured correlation not only between frames but between Group of Picture (GOP). Proper traffic models can be used to synthetically generate traffic, which can be used as inputs to networks for performance studies on parameters such as cell loss and cell delay, etc.

Our first contribution is an improved Autoregressive Model which takes into account the periodicity in the GOP structure. A GOP starts with an intracoded frame (I-frame). After every 15 frames (in our case), an I –frame is generated. This is then a period of 15, and it repeats itself throughout the video sequence. The Autoregressive process has been used in existing works to model the Mpeg video traffic that attempts to capture the frame correlation as well as the gaussian shape of the bit rate variation. However, the autoregressive process alone does not capture scene changes. We propose an Autoregressive model of order P, AR(P) + IAP (Interrupted

Autoregressive Process), to capture scene changes. We compare the model performance to that of the actual video trace, as well as the autoregressive process without scene changes. We have carried out a performance analysis of a Network Multiplexer with this new input stream.

Our second contribution, is in characterizing a relation between Correlation and Effective Bandwidth. Effective capacity as a function of correlation in several moments of video clips demonstrates interesting behavior. In several cases of analysis and measurements, it is observed that depending on the parameters of the source, effective capacity may be below or above the steady state value. Knowledge of the source parameters will aid network designers to better estimate bandwidth needs, particularly, in the transient time interval. Hence, attention has been given to such behavior and we have tried to explain the implications.

ACKNOWLEDGEMENTS

Many individuals have touched my life on the road to finishing up the most painful yet fun experience of my academic life.

First, I would like to thank Dr. Ron Brown of the Chemistry department for giving me good directions to secure the Magnet Scholarship and for his sustained effort throughout my graduate program. Many thanks to Dr. Adebimpe for giving me the boost at a time when I was fallen apart. The financial assistance and encouragement really revived me. My sincere thanks go to Ms. Sandra Smith for helping me process my paper work for financial assistance.

I like to express great thanks to my mentor, Professor Kaliappa Ravindran, for countless hours in correcting my work and helpful suggestions and insight. I like to thank both Professors Mjung Lee and Tarek Saadawi for reading through my work and making invaluable suggestions. My external advisor, Dr. Kazem Sohraby, spent countless hours with me through some difficult periods, times when things got out of hand. I thank him for his patience and for making difficult periods fun time. I want to thank Mohcene Mehzoudi and Tewfik Doumi at Bell Labs, for hours spent with me in lengthy discussions.. and bouncing ideas. I want to thank Paul Heyford, Kim Matthews and Chris Yu for setting up the lab at Murray Hills and coaching me to collect Mpeg-2 Video streams. My sincere thanks to Dr. Bharat Doshi for getting me started with Q+-the performance analysis tool. Carol Janczewski and Cindy Funka-Lee coached me to effectively use this tool.

There are two individuals that have really supported me in this endeavor. I thank God, the greatest supporter, for making these individuals cross my academic path. These are Dr Victor Bernard Lawrence and his family, and Dr. Edwin Lambert, all of Bell Labs. Mere words cannot describe the unprecedented and tireless support they have given me over the years. It is a story of my life that I have to devote a whole chapter to.

My special thanks go to my wife Esther, and my two daughters Amy and Crystal for their patience over the years. They encouraged me during difficult and emotional times. And I certainly want to express deep appreciation for their loving support. Additionally, I like to thank Maryann, Cheryl, Joyce, and Debbie, all of the reproduction unit of Lucent Technologies. My sincere thanks go to Dr. Constantine Osiakwan for valuable editing tricks which saved me much time.

Most of all, I like to thank ALMIGHTY GOD for preserving our lives over the years and helping us to see good days.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

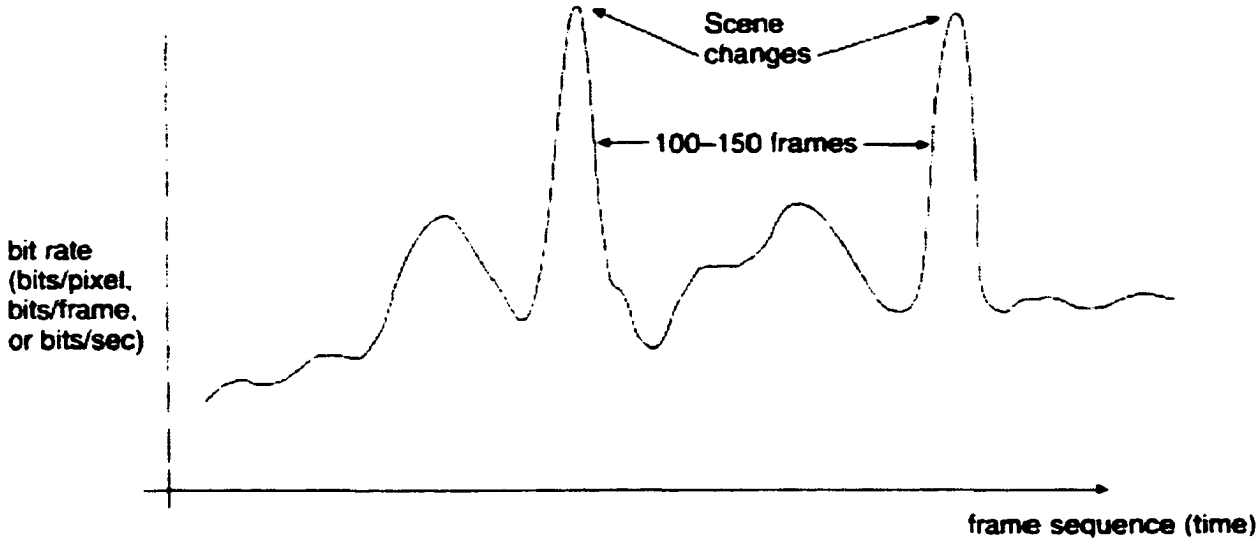
Today, as never before, communication networks are being designed to support bandwidth-hungry applications, such as video, image, voice, data, and high definition television (HDTV). There is, therefore, the need for proper traffic models to realize, and to transport such data over networks such as Asynchronous Transfer Mode (ATM) with the desired performance. In the video arena, the Motion Picture Expert Group-2 (Mpeg-2) is the international standard of video compression. These compression techniques lead to minimizing storage capacity or bandwidth, measured in bits per second. What determines the bit rate of compressed video is scene complexity, picture format, and constraints such as Quality, Delay, and Encoder complexity and algorithm, as well as noise.

Mpeg-2 Variable Bit Rate (VBR) video traffic consumes large bandwidth. Proper traffic models which are able to capture such traffic characteristics have attracted the attention of researchers in the area of traffic engineering and network performance. Some of the models have captured scene changes that lead to bursty traffic. Others have dealt with correlation not only between frames, but also between Group of Pictures (GOP). Other traffic models have captured self-similarity (long-range dependency) inherent in video traffic. As of yet, no particular traffic model has been obtained that captures all the traffic characteristics.

Since correlation, both intra-stream and inter-stream, impact system performance, an analytical model that captures a relation between correlation and effective capacity

should help network engineers to properly allocate bandwidth. In the past, effective capacity calculations have assumed independent sources. These models did not deal with correlated sources. The effective capacity has therefore been the sum of the effective capacities from independent heterogeneous sources. The impact of the independent traffic model without considering correlation leads to losses among other performance measures. Incorporating the effect of correlation will reduce losses, and allow capturing its impact on the quality of service. Our work on Mpeg-2 variable bit rate has factored in the effect of correlation both autocorrelation and cross-correlation.

Variable-bit-rate (VBR) video traffic also exhibits burstiness. Figure 1.1 shows a typical bit rate as a function of time. Different compression algorithms exhibit different bit rate characteristics, but all are of this general form[1]. Several characteristics are worthy of notice. First is the relatively slow variation of the bit rate of transmission. A second characteristic is that the bit transmission rate is found to increase dramatically for a brief interval of time (1-2 frames) and then drop down to a more normal range. This is attributed to scene changes, with correspondingly large changes in the picture content. The coder then takes a brief interval of time to adapt to the change and resume its normal compression process. Such scene changes may occur anywhere from 100-150 frames, and even more apart. Video signals, despite the heavy compression, typically manifest high correlation from frame to frame (most often the frame contents do not change very much from frame to frame). A typical correlation (similarity) measure is 10-20 frames corresponding to 300-600 msec.



Reproduced from [1]

Figure 1.1: Typical compressed video bit rate

Mpeg VBR traffic leads to constant image quality, since any possible degradation in quality due to scene changes are correspondingly compensated for with the encoder's generation of more bits. Additionally, because of its bursty nature, output channel could be under utilized during the burst period, leading to waste of bandwidth. Proper scheduling can lead to maximum use of bandwidth leading to statistical multiplexing gain. This though, can make network design and management difficult to perform. Effective design and performance analysis depend on accurate modeling of the traffic. VBR video sources are important from a modeling perspective, because they can represent a variety of data source behaviors. The modeling exercises are demanding because of the randomness in the rate fluctuations, and the statistical dependencies of rates over time, as well as their complex generation scheme (coding algorithm).

Existing models based on Markovian structures have been widely used to statistically approximate VBR video traffic. All these models have in common an asymptotically exponential decay of the autocorrelation function and a rapidly decaying marginal distribution tail. Further, they lack a systematic way of simultaneously fitting both the empirical marginal distribution and the autocorrelation function. Recent extensive measurements of real traffic data, have led to the conclusion that VBR video traffic cannot be sufficiently represented by the traditional models, but instead can be more accurately matched by self-similar (fractal) models [2]. The crucial feature of self-similar processes is that they exhibit long-range dependency (LRD), that is their autocorrelation function decays less than exponential and is non-summable. This is in contrast to the traditional stochastic models, all of which captures short range dependency (SRD), i.e. they have an autocorrelation function that decays exponentially (or faster). The serious implication for network design is that, results based on traditional models may not be applicable under self-similar traffic. Recent studies on self-similar traffic have shown that LRD structure may have a significant impact on queueing performance in networks [2]. Because of possible network performance issues, it is important to study the characteristics of multimedia traffic.

The main characteristics of multimedia are related to "synchronization" which assures a temporal order of events. A multimedia stream is therefore characterized by multiple data streams related to each other by means of proper time relationships. Although several models have been proposed to define synchronization properties, analytical paradigms for multimedia synchronized traffic have not been adequately developed. The main difficulty in defining an arrival process for modeling multimedia

traffic is concerned with the mutual dependence of the monomedia components model. For this reason, in fact, the independence hypothesis which several models for mixed traffic [3-7] are based on, is not valid for multimedia sources. In particular, the auto-and cross-correlations have to be taken into account.

Analytical paradigms have been proposed for modeling multimedia source and modeling of the arrival processes as a superposition of a number of heterogeneous correlated arrival processes. There still remains an investigation of long-range dependency on resource allocation, particularly bandwidth. Of more interest will be an analytical model that captures overall correlation among the traffic sources and Effective bandwidth in networks.

Current works have investigated different types of models in the context of ATM networks. Among those models, a Markov-modulated Poisson process (MMPP) proposed by Heffes and Lucantoni [8] is well known. The MMPP captures stochastic characteristics based on Poisson processes. Hence, it allows the exact queueing theoretic analysis for the calculation of performance measures such as cell loss probability, cell transmission delay, and so on. However, the MMPP is not capable of representing the autocorrelation of a traffic, which is a measure of burstiness.

In the area of traffic engineering, it is important to specify the specific control point in a video transport system at which we apply our model. Figure 1.2 represents a video transport system with two main phases: 1) encoder phase and 2) the scheduler phase.

In figure 1.2, a periodic sampling of the output of a video encoder generates a VBR stream at fixed durations. These are then buffered, and transmitted over the output link at a constant rate, Rp, but different transmission times by the network scheduler.

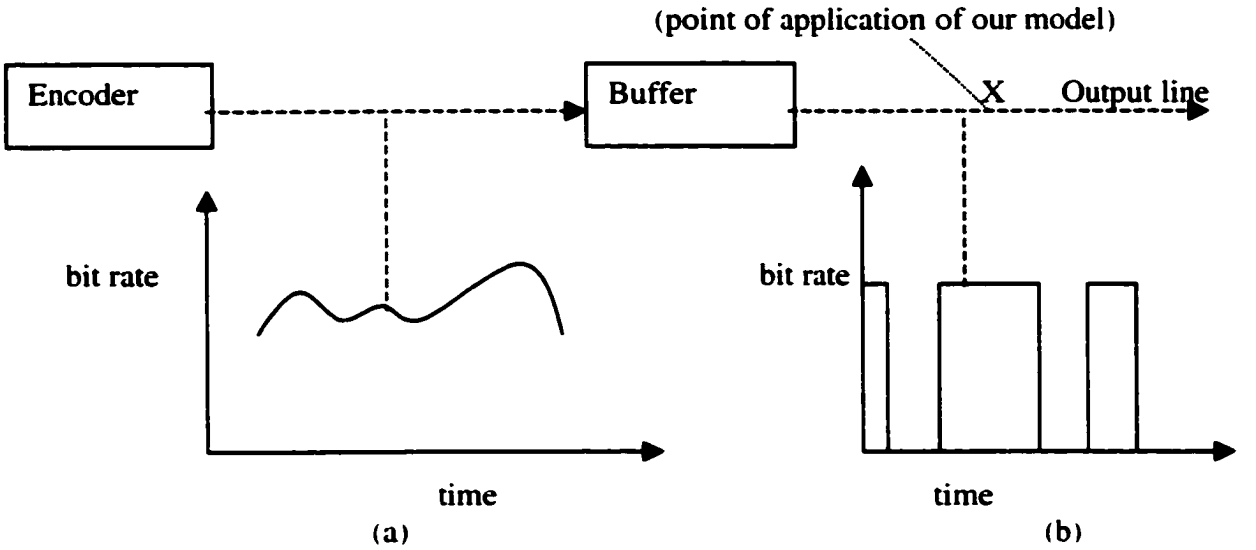


Figure 1.2: System transport Architecture a) Encoder and b) Scheduler phase (Fixed Rate)

Traffic engineers have modeled the variable bit rate (VBR) at the encoder phase by various traffic models such as the Autoregressive model (AR), self-similar model, etc. Some of these models capture the autocorrelation between successive frames, scene changes, and the long-range dependency. We have modeled the video source at the scheduler phase of the transport system. Figure 1.3 shows the scheduler phase. The buffered VBR traffic is transmitted over a constant rate transmission line with variable on/off times. The 'on' periods are the transmission times and the 'off' periods are the off times before the next transmission. Notice that when a buffer is added, an inevitable

delay is introduced as a consequence. Figure 1.3b shows the delay. This delay adds to the delay that is encountered at the decoder because of rearrangement of frames for playback. To combat this additional delay, the scheduler at the decoding end may need to accommodate this additional delay when playing out the video frames.

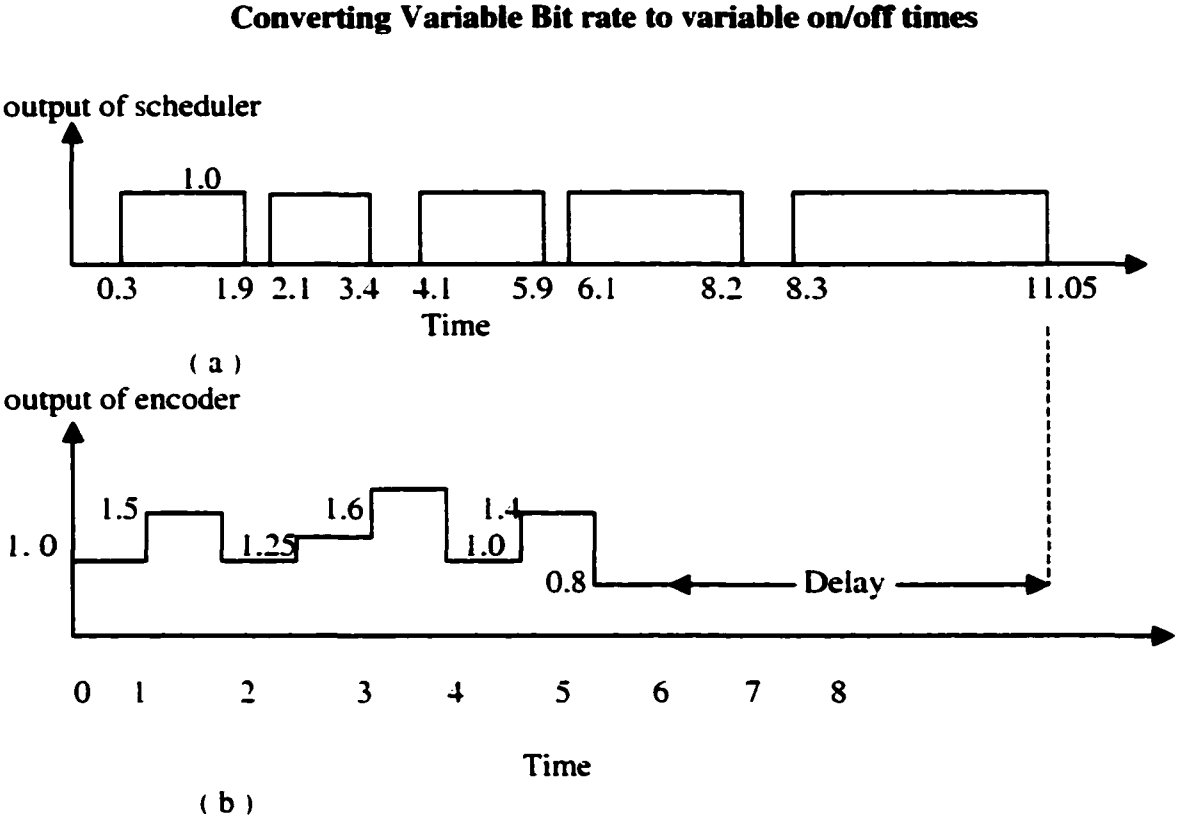


Figure 1.3: a) variable on/off times in scheduler b) output of encoder (discretized view)

Notice that the area covered in both graphs are the same, and there is a slight shift in time in the scheduler phase for framing packets. This video transport architecture will be referred to in chapter 6 of this thesis.

This thesis has been organized as follows: chapter 2 describes the sources of variability in Mpeg-2 video traffic. Variability in the bit rate has an effect on the auto regressive model, and does affect bandwidth allocation. Chapter 3 describes the statistical analysis of Mpeg-2 video traffic. The statistical analysis also covers the distributions of the three picture formats, namely the I, P, and B frames, the correlation, and long-range dependency. The distributions of these picture types play a key role in the formulation of the 'Improved Autoregressive Model.' Related works in the area video traffic modeling are discussed in chapter 4. Chapter 5 provides an improved AR model, along with analytical results that show the superior performance of our model in terms of loss rate. Chapter 6 provides a model that takes into account the effect of correlation when calculating bandwidth. It also discusses the effect of short and long-range dependency, as well as burstiness, on bandwidth allocation. In chapter 7, the observation that effective capacity can be below or above the steady state capacity, is discussed and the notion of effective capacity switchover is introduced. Chapter 8 provides a model that takes into account the effects of correlation from various sources. In chapter 9, we draw various conclusions, and further works. This chapter also restates our contributions.

We have briefly mentioned, at a high level, the various traffic models that researchers have used to model variable bit rate video traffic. Now we move to treat in detail the sources of variability in the Mpeg-2 traffic in chapter 2.

2. SOURCES OF VARIABILITY IN MPEG-2 VIDEO TRAFFIC

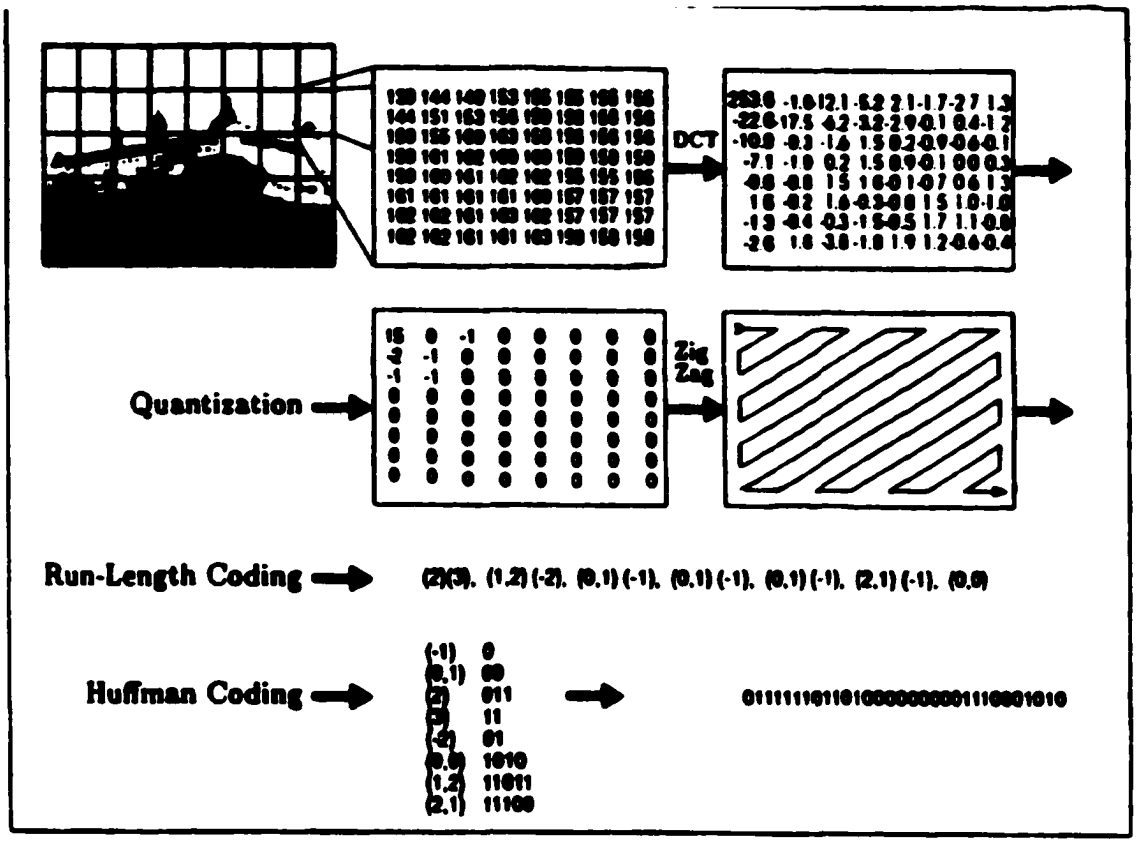
To properly model this class of traffic with its variability, it is important to understand what gives rise to such variability. This will aid traffic engineers to better understand the complexity involved in the modeling process.

2.1. Mpeg video compression and coding

The compression algorithm of Mpeg-2 leads to variability in the bit rate behavior rather than a constant bit rate [1, 9]. The variable bit rate behavior compounded by the randomness in the network behavior makes the problem of managing video traffic in the network intractable. Since video is a sequence of still images, it is possible to compress or encode a video signal using techniques similar to the Joint Picture Expert Group (JPEG). Such methods of compression are called intra-frame coding techniques. Each frame of video is individually and independently compressed or encoded. Intra-frame coding exploits the spatial redundancy that exists between adjacent pixels of a frame.

As in JPEG, the Mpeg-2 intra-frame video coding algorithms employ a Block-based two-dimensional Discrete Cosine Transform (DCT). A frame is first divided into 8x8 blocks of pixels, and the two-dimensional DCT is then applied independently to each block. This operation results in an 8x8 block of DCT coefficients in which most of the energy in the original block is typically concentrated in a few low-frequency coefficients. A quantizer is applied to each DCT coefficient that sets many of them to zero. This quantization is responsible for the lossy nature of the compression algorithm. Compression is achieved by transmitting only the coefficients

that survive the quantization operation. and by entropy coding (Huffman coding) their runs and amplitudes [9]. Figure 2.1 shows how the bit stream for one block of a still image is generated. The degree of motion and complexity of picture dictates the number of bits in a block.



Reproduced from [1]

Figure 2.1: Generation of variable # of bits with Inter-frame Coding

The quality achieved by intra-frame coding alone is not sufficient for typical video signals at bit rates around 1.5 Mbps. Therefore, inter-frame coding techniques are used to reduce the temporal redundancy which results from a high degree of correlation

between adjacent frames. Mpeg uses a combination of JPEG (Joint Picture Expert Group), (ISO(1991)) and H.261 video conferencing standard(CCITT(1990)). The H.261 algorithm exploits this redundancy by computing a frame-to-frame difference signal called the prediction error. In computing the prediction error, the technique of motion compensation is employed to correct the prediction for motion.. A block-based approach is adopted for motion compensation, where a block of pixels, called the target block, in the frame to be encoded, is matched with a set of blocks of the same size in the previous frame, called the reference frame. The block in the reference frame that best matches the target block is used as the prediction for the latter, i.e., the prediction error is computed as the difference between the target block and the best matching block. This best-matching block is associated with a motion vector that describes the displacement between it and the target block. The motion vector information is also encoded and transmitted along with the prediction error. The prediction error itself is transmitted using the DCT-based intra-frame encoding technique summarized above. In Mpeg video, the block size for motion compensation is chosen to be 16x16 pixels, representing a reasonable tradeoff between the compression provided by motion compensation, and the cost associated with transmitting the motion vectors. Pictures coded using forward prediction are called P-pictures. Thus, Motion Compensation vectors and prediction errors encoded and transmitted instead of the original pixel information, imply less information transmitted. For high activity movies with lots of scene changes, the prediction error can be high leading to more bits to be encoded. For low activity movies with few scene changes, the prediction error will be relatively low leading to fewer bits to be encoded.

Bidirectional prediction provides a number of advantages. The primary one is that the compression obtained is typically higher than that obtained from forward prediction. To obtain the same picture quality, bidirectionally predicted frames can be encoded with fewer bits than frames using only forward prediction. However, bidirectional prediction introduces extra delay in the encoding process since frames must be encoded out of sequence.

The Mpeg encoder encodes scene changes with more bits leading to variable bit rate. What else determines the variability in bit rate? Several factors such as scene complexity, picture format, constraints such as quality, delay, encoder complexity and algorithm, and noise.

2.2. Summary

In this chapter, the Mpeg video compression and coding techniques that lead to variability in the bit rate were discussed. The effect of scene changes on the encoder was mentioned. The effect of scene complexity, picture format, algorithm, and noise on the bit rate were described. To help with an appropriate model to use, the statistical nature of the various picture formats, such as the I, P, and B frames must be studied. This leads us to chapter 3 about the statistical analysis of Mpeg video traffic.

3. STATISTICAL ANALYSIS OF MPEG-2 VIDEO TRAFFIC

Traffic models obtained by several researchers, as well as our proposed improvements, depend upon the understanding of the statistical nature and characteristics of this kind of traffic. Hence, a fair amount of time will be spent on understanding the statistical characteristics, such as distribution of the individual frames, and the distribution of the GOP sizes. Correlation effect and long-range dependency will also be discussed.

3.1. Frames and GOP statistics

The various traffic models addressed in chapters 4 and 5 of the thesis, depend upon an understanding of the statistical nature of the three frame types, namely: Intracoded (I-frames), Predictional (P-frames) and Bi-directional (B-frames).

First, we introduce the MPEG-2 encoded sequences used for our statistical analysis in table 3.1. Simple statistics are provided such as moments and peak-to-mean ratios for both the frame sizes and the GOP sizes, where the GOP size is the sum of frame sizes of one GOP (see table 3.2). We fit model distributions to the frame and GOP size histograms and analyze the correlations of both the frames and GOPs. Table 3.2 shows the mean values, coefficient of variation (COV), and the peak-to-mean ratios of the frame and GOP sizes. In section 3.2, we discuss the distributions of the three frame types and the GOP sizes.

Table 3.1: Overview of 11 encoded sequences

1) Batman---------- batman 2) Tomorrow never dies ----------Indies

3) Eric Clapton----------clapton 4) Phenomenon, by John Travolta-------travolta

5) West 54^{th} ----------w54 6) Silence of the Lambs----------lambs

7) Wizard of Oz ----------oz 8) Gone with the wind----------wind

9) The twister---------- --twister 10) The Fugitive----------fugitive

11) The mask of zorro----------zorro

3.2. Distributions

Figures 3.1 and 3.2 show the frame size histograms of the I -, P -, and B – frames of Zorro and Clapton sequences respectively. The shapes of the curves indicate that the I-frames may be approximated by a normal probability density function, whereas the P – and B – frames have a histogram resembling a lognormal probability density function. Gamma or lognormal distributions are commonly suggested to model the frame and GOP sizes of MPEG video sequence [10].

For the probability density function fitting, we used QQ plots. QQ plots is a statistical method to assess whether data have a particular distribution, or whether two data sets have identical distributions. If the distributions are identical, then the plot will be linear (see appendix). The frame and the GOP size histograms of almost all traces can be well approximated with lognormal probability density functions [10]. Only for a small number of I frame histograms, both normal and lognormal probability density

functions lead to a reasonable approximation accuracy. For illustration, we show the QQ plots for the Zorro I – frames and the Zorro GOPs in the appendix .

To sum up, lognormal probability density functions are adequate model probability density functions for both frame and GOP sizes of VBR MPEG video sequences. This observation is one of the motivations for our traffic models.

Quantization = 8

Table 3.2: Simple Statistics of the encoded sequences

Sequence	Frames			GOPs		
	Mean [bytes]	CoV	Peak/Mean	Mean [bytes]	CoV	Peak/Mean
Batman	2391	0.5756	7.6689	25466	.2382	3.2825
Indies	1757	0.6134	5.6515	26376	0.3191	2.7636
Clapton	3287	0.4980	3.7497	49288	0.3020	2.0574
Trovolta	1966	0.4987	5.4047	29645	0.2599	2.2826
W54	3582	0.4387	3.3569	53550	0.2006	1.4826
Lambs	1893	0.6743	6.0932	28674	0.4797	2.7806
Oz	4499	.5348	3.4750	67700	0.2960	1.7957
Wind	3640	.6105	16.4815	55027	0.4152	2.4594
Twister	2198	.3762	3.7889	33000	0.2190	1.8309
Fugitive	2307	.4715	5.1579	34673	0.2797	2.4753
Zorro	4047	.4267	4.2791	26376	0.3191	2.7636

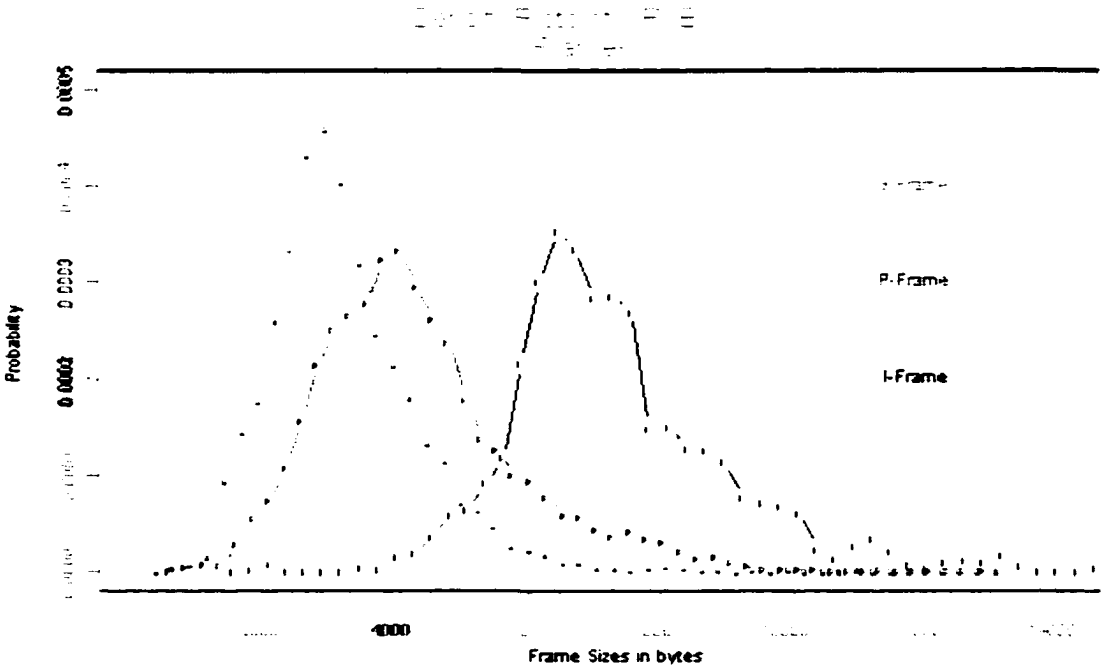


Figure 3.1: Frame size histograms of the I-, P-, and B- frames of zorro

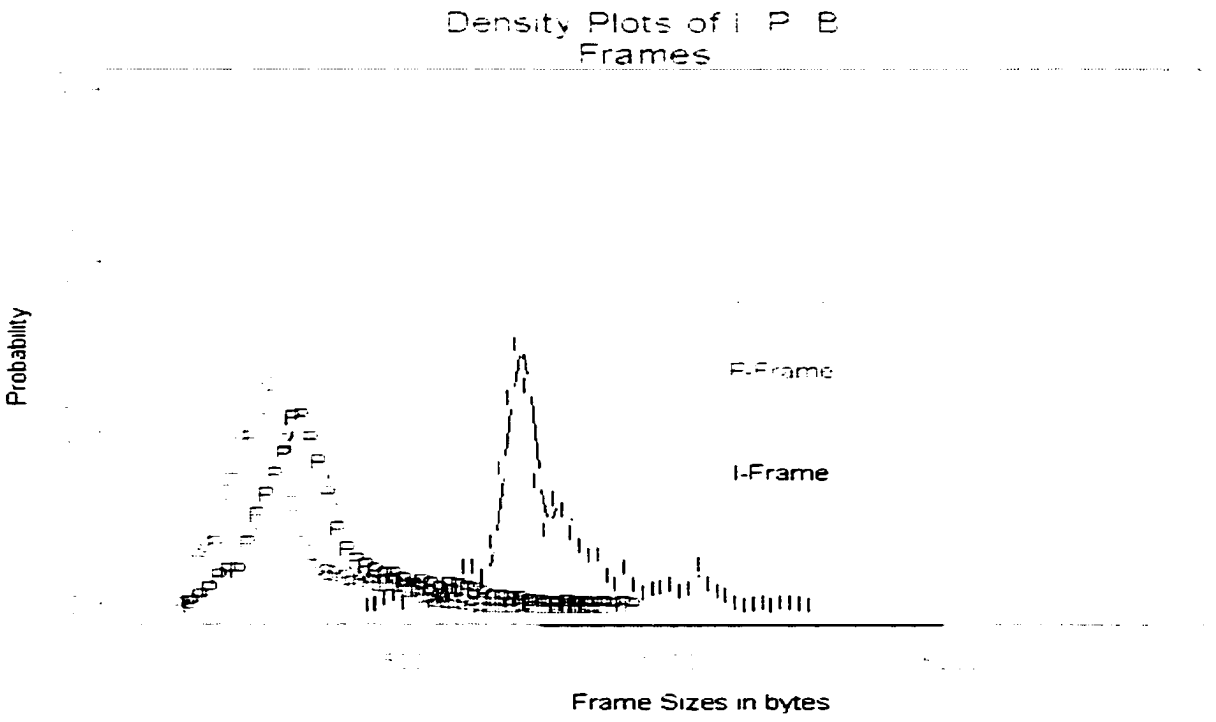


Figure 3.2: Frame size histograms of the I-, P-, and B- frames of Clapton

3.3. Correlations

Time-dependent statistics are important since correlation in data streams can affect the performance of networks carrying this traffic. All studies [11], report that packet losses and/or delays of the queueing systems are considerably higher for positively correlated input traffic than for uncorrelated input traffic. Thus, it is important to understand the correlation properties of video traffic.

There are a number of measures for second-order properties of empirical time series, such as auto correlation functions (ACFs), periodograms, indices of dispersion, and self-similar properties [11]. We will focus on ACFs and the estimation of self-similar properties since we use this sequel to determine our video traffic models. Auto correlation of frame sizes depend on the pattern of the GOP and, in principle, always look like figure 3.3, assuming the same GOP is used for the whole sequence. The correlation between frames at the current observation point, and the future observation point (every 16^{th} frame), indicates that I -- frames slowly decay in amplitude. Mpeg-2 has a unique structure determined by (N), the number of frames before the next I frame, and (M), the number of frames before the next P frame in one Group of Picture (GOP). A group of picture is defined as :

I B B P B B P B B P B B P B B I ... N=15, and M=3.

In [10], the author shows a strong frame-by-frame correlation. If a model is needed which captures the frame-by-frame correlations of an MPEG video traffic stream, the GOP – pattern based shape of the auto correlation function has to be considered. Frame-

by-frame correlation does not capture long-range dependency. The next section discusses the long-range dependency in video traffic.

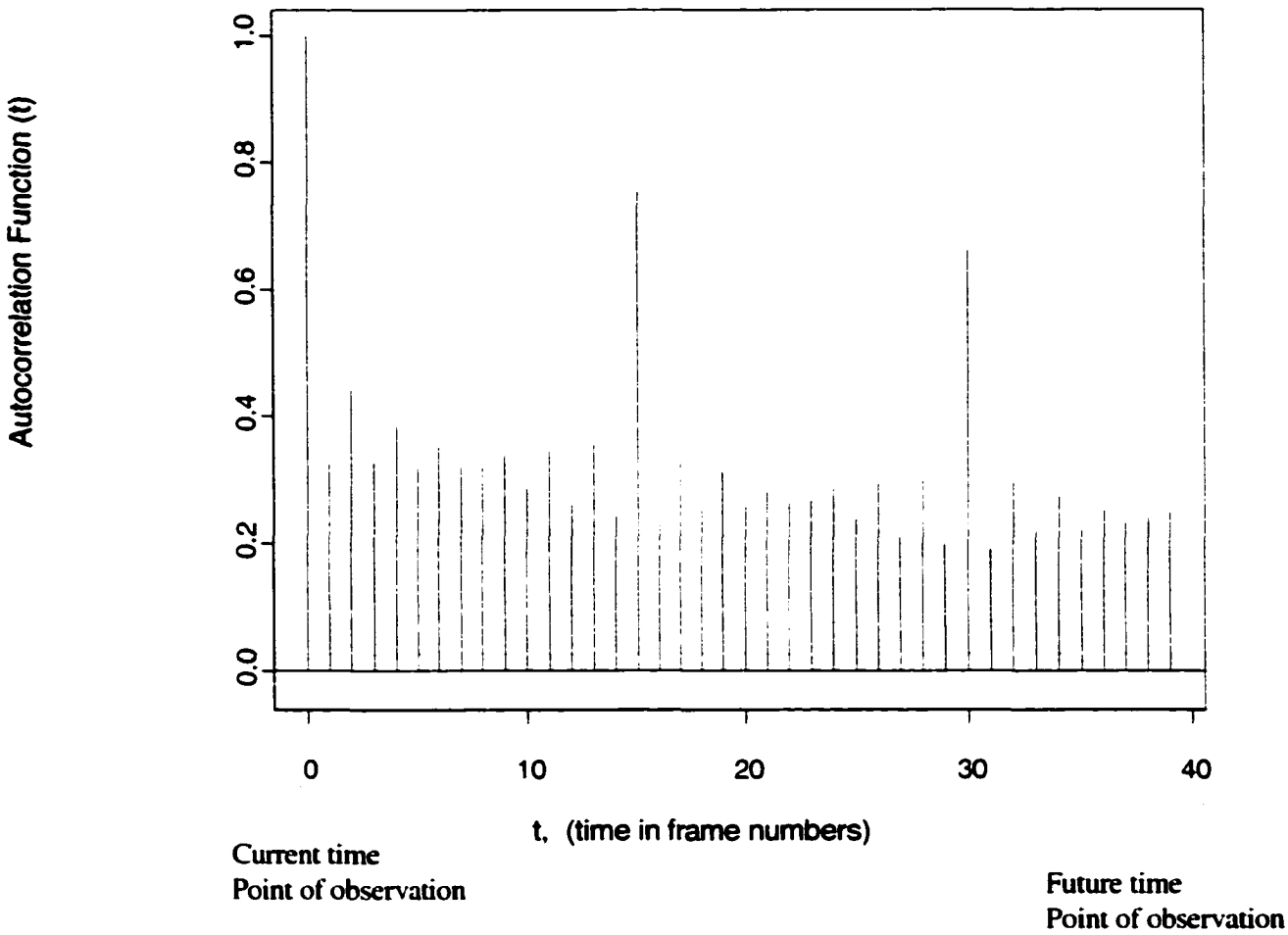


Figure 3.3: Autocorrelation studies of frames (video clip of zorro)

3.4 Long-range dependencies

More recently [12], variable–bit-rate (VBR) video traffic was found to exhibit self-similar characteristics, similar to LAN traffic. The crucial feature of self-similar processes is that they exhibit **long-range dependency (LRD),** that is their

autocorrelation function r(k) decays slower than exponential, and is non-summable, i.e., $r(k) \sim k^{-\zeta}$, as $k \to \infty$, for $0 < \zeta < 1$ (the quantity $H = 1 - \frac{\zeta}{2}$ is called the Hurst parameter). A Hurst parameter greater than 0.5 indicates long-range dependency. This is in contrast to traditional stochastic models, all of which exhibit short-range dependency (SRD), i.e., have an autocorrelation function that decays exponentially or faster [13].

To obtain an improved support for the decision about the appropriateness of a certain model, we need to examine the long-range correlation behavior of the video sequences. The importance of detecting the presence of this behavior is two fold. First, 'queueing systems, and generally statistical estimators' reaction on long-range dependent input streams differs from that of uncorrelated input. Second, most of the models, such as finite Markov chains and finite order autoregressive processes, are not capable of modeling long-range dependency. Hence, to compare model-based and trace-based traffic, we need to know whether long-range correlation has been factored in. Otherwise, performance measures could differ significantly.

It is difficult, however, to evaluate long-range dependency of the video sequences by means of the autocorrelation function alone. A succinct way of measuring long-range dependency is through the hurst parameter, H. The H parameter measures the degree of similarity between GOP sizes. An algorithm for measuring the hurst parameter, H, rescales the range R, of the video sequence, and S is the standard deviation of the new range. R/S is the ratio of the rescaled range over the standard deviation. The slope of the plot is the index of long-range dependency. Given an empirical series $\{ x_t : t = 1, \ldots, N \}$, the whole series is subdivided into K non-

overlapping blocks [14]. Now the rescaled adjusted range R (t_r,d) over S (t_r,d) for a number of ranges r, where $t_i = \lfloor N/K \rfloor (i-1) + 1$ are the starting points of the blocks which satisfy $(t_i - 1) + r < N$, is computed.

$$R(t_r, r) = \max \{ 0, W(t_r, 1), \ldots, W(t_r, r) \} - \min \{ 0, W(t_r, 1), \ldots, W(t_r, r) \}$$

where

$$W(t_r, k) = \sum_{j=1}^{k} X_{ti+j-1} - k \cdot \left(\frac{1}{r} \sum_{j=1}^{r} X_{ti+j-1} - 1 \right).$$

$K = 1, \ldots, r.$

The formula [14] divides the sequence into a number of ranges, r. The starting points of each range is t_i that is the floor of N/K times $(i-1) + 1$. R is the rescaled adjusted range and S is the standard deviation.

Let $S^2(t_r, r)$ denote the sample variance of $x_n, \ldots, x_{n+r-1}$. For each value r we obtain a number of R/S samples. For small values of r there are k samples. The number decreases for larger ranges r due to the limiting condition on the t_i values mentioned above. These samples are computed for logarithmically spaced values r, i.e. $r_{i+1} = m \bullet r_i$ with m>1 starting with a value r_0 of about 10. Plotting log [R (t_i, r)/S (t_i, d)] versus log r results in the R/S plot.

Next, a least squares line is fitted to the points of the R/S plot, where R/S samples of the external ranges are not considered. The R/S samples of the smallest ranges are dominated by short-range correlations and samples of large ranges are statistically insignificant if the number of samples per range is less than say 5. The

slope of the regression line for these R/S samples is an estimate of the Hurst parameter H. Both the number of blocks K and the number of values r should not be chosen too small. In addition, some care has to be taken into consideration for the regression line. In practice, it has to be checked whether different parameter settings lead to consistent H estimates for $\{X_t^{(m)}\}$ with different aggregation levels m.

Figures 3.4, 3.5 show the R/S plot for the Zorrogop and Claptongop sequence with K = 10. The regression lines have slopes of .9 and .98 respectively for Zorrogop and Clapton indicating a Hurst parameter of H = .9 and .98

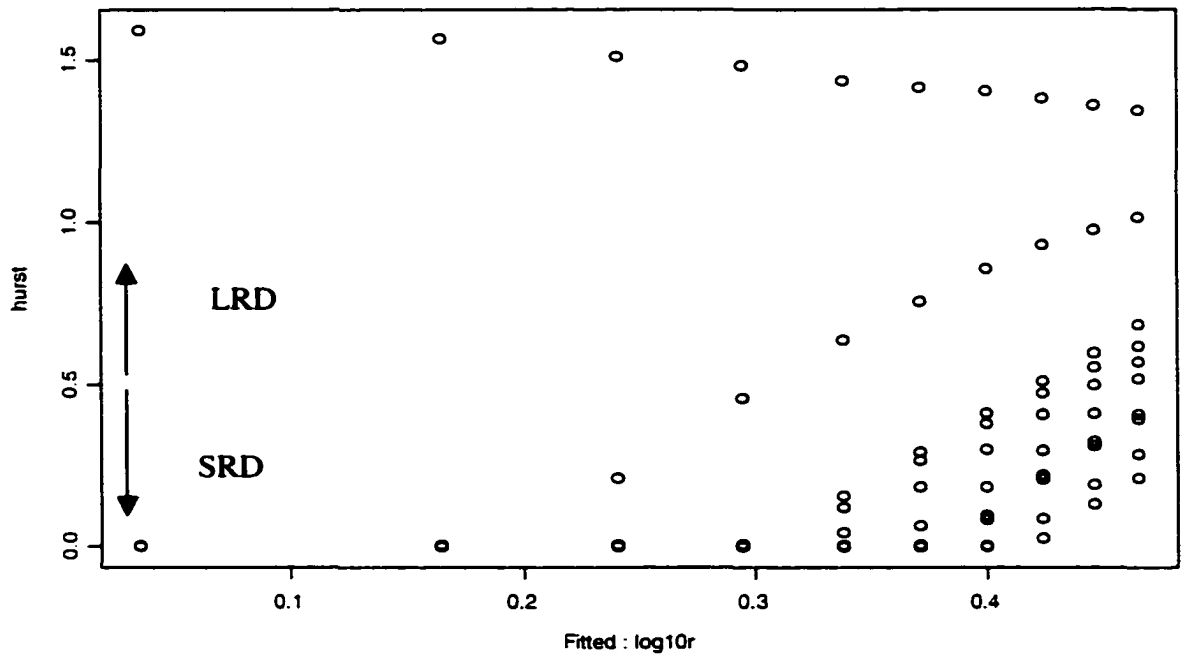


Figure 3.4: Hurst coefficient vs. log10r for zorrogop

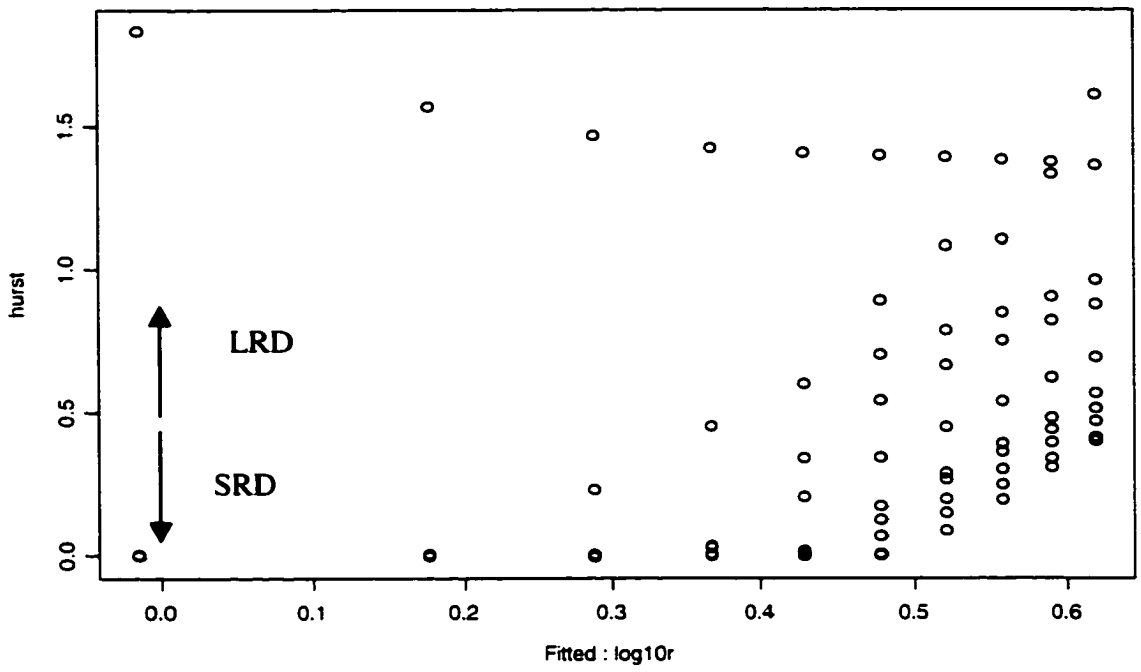


Figure 3.5: Hurst coefficient vs. log10r for claptongop

3.5. Summary

In chapter 3, we discussed the statistical distributions of the Mpeg frames. The video clips were collected at Bell Labs, at Murray Hill, the research unit of Lucent Technologies. The frames statistics were computed using MATLAB. For the most part, the I frames can be thought of following a normal or lognormal distribution. The P and B frames follow a lognormal distribution. The chapter also discussed the correlation between Mpeg frames, as well as the group of pictures (GOP).

The long-range dependency, measured by means of the hurst parameter was presented. Based on the statistical measures described so far, we now examine the related works as to how these measures are used in these works.

4. RELATED WORKS ON MPEG-2 TRAFFIC MODELING

In lieu of actual traces, synthetically generated traffic traces can be used to study network performance. Appropriate models that capture specific traffic characteristics have caught the attention of researchers in traffic engineering. In this chapter, we will review a few of the major traffic models in the literature, such as the histogram, the scene-oriented, the self-similar and autoregressive models. We will look at how the various parameters are computed, and the model traces generated. Then in chapter 5, we will indicate the first of our contributions by expanding on the Autoregressive Model.

4.1 Histogram-based studies

Histograms are among the simplest modeling approaches, and they do not take account the correlation structure of the video data set. Therefore, they provide no in-depth statistical inference for the computation of their parameters. Krunz et al. (1995)[10] provides a model for Mpeg video traffic with all three frame types although the authors used a fitted lognormal distribution instead of a histogram. They used a simple approach of cycling through a set of three different lognormal random variables, i.e., one for each frame type, to generate a sequence which reflects the GOP pattern of Mpeg coding. Skelly et al. (1993)[15] report that a histogram model provided good results for estimating the buffer occupancy distribution of an Atm Multiplexer with

video input sources. Schroff and Schwartz (1994)[16] use these results to estimate end-to-end cell loss probabilities for video traffic. Both papers are based on data sets which consists of only one frame type.

The histogram model is equivalent to modeling a time series by independent and identically distributed (i.i.d) random variables. "Identically distributed" means that the intervals have the same probability distribution. This implies that a number of algorithms for the analysis of queueing systems with i.i.d input traffic can be applied. The main disadvantage of this model is that any GOP-by-GOP correlation remains uncaptured.

4.1.1 Parameter Estimation

Let $\{x_i : i = 1, \ldots, N\}$ denote the GOP size trace. The only user-defined parameter of the histogram model is the number k of the histogram intervals. The relative frequency h_j of the samples in the GOP size intervals and the GOP size s_j related to interval j (j =1,....,k) are computed using the formula in [17], as described below.

4.1.2 Generation of A Model Trace

Let $\{\varepsilon_i\}$ be a U(0,1) distributed white noise. Given the frequencies h_j, and GOP trace $\{t_i\}$ is generated

by $t_i = S_j$ with $j = \min\{ l : \sum_{j=1}^{l} h_j > \epsilon_i \}$

For each of the models of this chapter we present one trace with 1180 samples of Zorrogop and discuss the shape and properties. The model parameters are estimated on the basis of the Zorrogop size sequence.

Figure 4.1 shows the first 1180 samples of this data set. We point out that trace diagrams are no proof of the quality of a model. The quality of a model is measured by more appropriate statistical methods, other than observing trace diagrams.

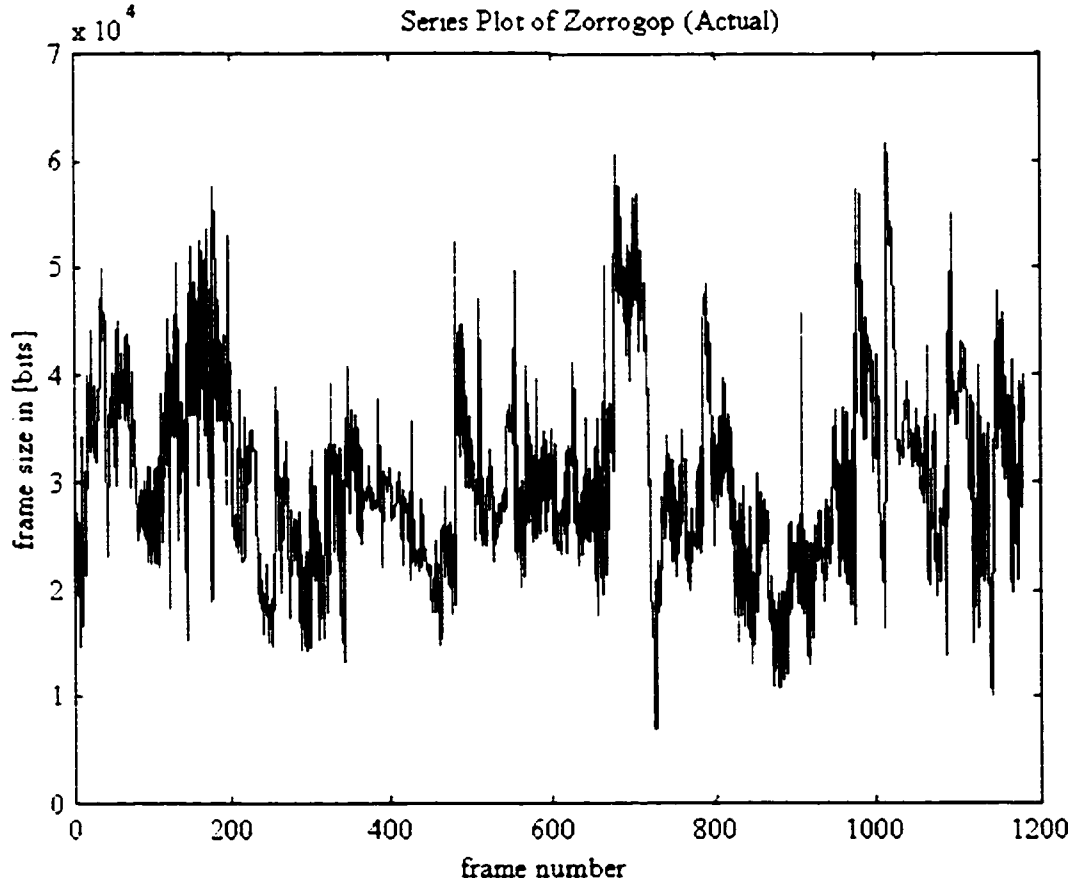


Figure 4.1: Trace from the zorrogop size sequence

Fig. 4.2 shows a trace generated by the histogram model with a number k = 11 intervals. The histogram model is affected by the bin and sample sizes. Appropriate bin sizes may be chosen using [17] to bring out needed statistical.

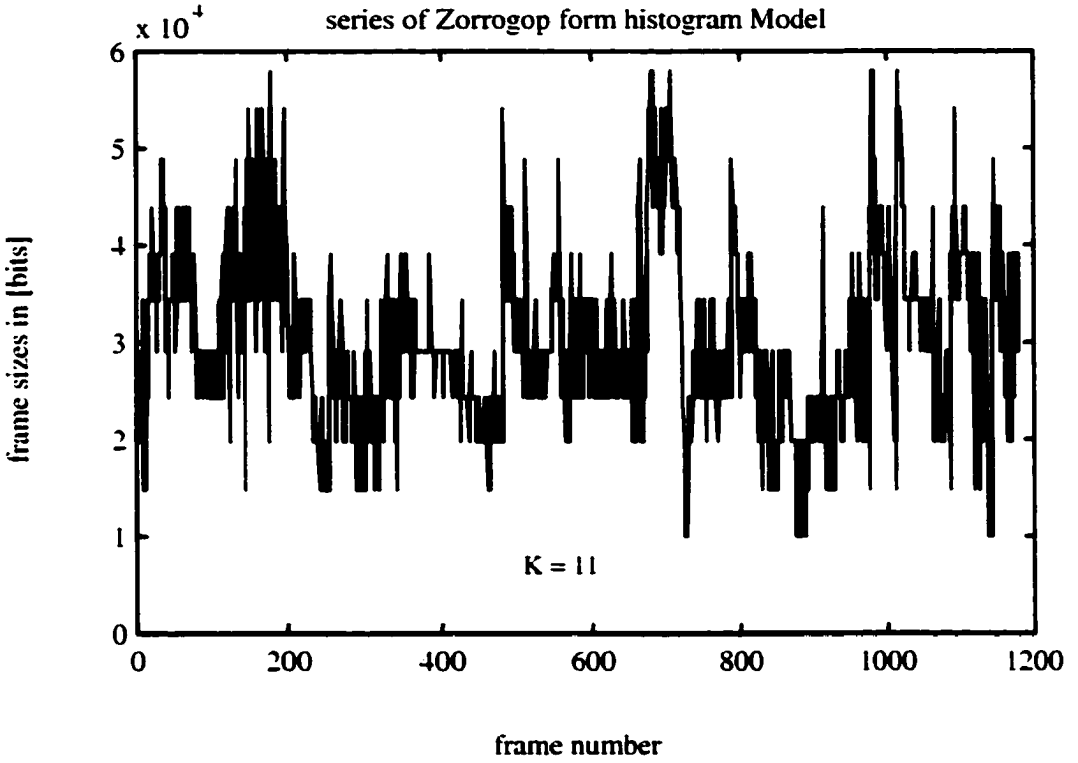


Figure 4.2: histogram model (zorrogop)

The histogram model has been used to model video traffic by putting sample sizes into bins. The model does not capture frame to frame correlation, but gives a simple method of modeling video traffic, that is putting samples into bin sizes without capturing any time dependencies.

4.2 Scene-level analysis.

The scene-oriented model is a first-order Markov chain with a redefined set of states. The intention of this redefinition is to facilitate the modeling of scene changes and to achieve an improvement of the autocorrelation modeling properties of the Markov chain, with a moderate increase in the number of states. Before we are able to classify the GOPs into different scenes we have to determine the scene boundaries of the video sequence. It is possible for offline observation of scenes by humans, which allows affixing markers along a video stream to depict boundaries of scene changes during subsequent play-outs of the stream. Instead, we use a method to find these boundaries which only depends on a few statistical properties of a group of consecutive GOP sizes. These parameters should be available by simply scanning the GOP size sequence without any knowledge about the content of the video sequence.

Heyman and Lakshman (1994)[18] present an algorithm to split a video trace into scenes. It is based on the fact that normally a new scene starts with a frame which is considerably larger than the preceding frames. This behavior results from the predictive encoding which is used in most video encoding schemes. The first frame of a new scene cannot be predicted from former frames and therefore contains more information. In this case, this algorithm is not applicable since our aim is to find out scene changes among GOP sizes. Due to the summing of frames, the sudden increase of frame size at the beginning of a scene is averaged out and is not detectable anymore. Thus, a new algorithm to determine scene changes for GOP sizes is in order [11].

4.2.1: Parameter Estimation

Let $\{x_i : i=1, \ldots, N\}$ denote the considered GOP size trace. In addition to the number of states M_G used to model the GOP sizes while being in a particular scene, we have to specify the number of scene classes M_S. To determine the scene class of each of X_i, the algorithm below is proposed. This algorithm groups together GOPs into scenes which have approximately the same GOP size.

4.2.2: Variation-Based Algorithm

Using the variation-based algorithm in [11] and described below, scene boundaries based on the coefficient of variation for a sequence of consecutive GOP sizes, were determined. GOP sizes were added to the sequence under consideration until the weighted coefficient of variation changed more than a preset value. The last GOP added is defined as the beginning of a new scene.

In the following, a formal description of the algorithm is presented. The scene change threshold ε_s is specified. Let N_G denote the current GOP number and N_s the current scene number.

Set $N_G=1$ and $N_s=1$.

Set the current left scene boundary $B_{left}(Ns) = 1$.

Increment N_G by 1. Compute the coefficient of variation cvnew of $\{X_{bleft(Ns)}, \ldots, X_{NG}\}$.

Increment N_G by 1. Set $c_v(\text{old}) = c_v(\text{new})$. Compute the coefficient of variation $c_v(\text{new})$ of $\{X_{bleft}(Ns), \ldots, X_{NG}\}$.

The formula below determines if the coefficient of variation is greater than threshold. If it is, there is a scene change, otherwise there is no scene change.

If $|c_v(\text{new}) - c_v(\text{old})|(n_G - B_{left} + 1) > \text{thresh}$; then set the right scene boundary.

$B_{right}(Ns) = N_G - 1$ and the left scene boundary of the new scene $B_{left}(Ns + 1) = N_G$.

Increment Ns by 1 and go to step (2).

If (i) does not hold go to step (3). Iterating the algorithm over the whole GOP size sequence provides a series of N_s scene boundary pairs. Now the average GOP size $x_v(i)$ for each scene is computed as below :

$$\bar{x}_v(i) = \frac{1}{b_{right}(i) - b_{left}(i) + 1} \sum_{k = b_{left}(i)}^{b_{right}(i)} x(k) \quad \text{with } i = 1, \ldots, N(s)$$

and extend the given GOP size trace $\{x_i\}$ to a series of pairs $\{(x_i, \bar{x}_i)\}$ with $x_i = x_v(s)$ for $b_{left}(s) \le i \le b_{right}(s)$, i.e. pairs formed by the GOP size and mean GOP size of the scene where this particular GOP is located. Experiments indicate that ε_s should be chosen such that the resulting average-scene length is at least ten GOPs to obtain a reasonable approximation quality of the auto correlation function [11]. The range of frame numbers which is well approximated has to be determined heuristically varying ε_s and M_s.

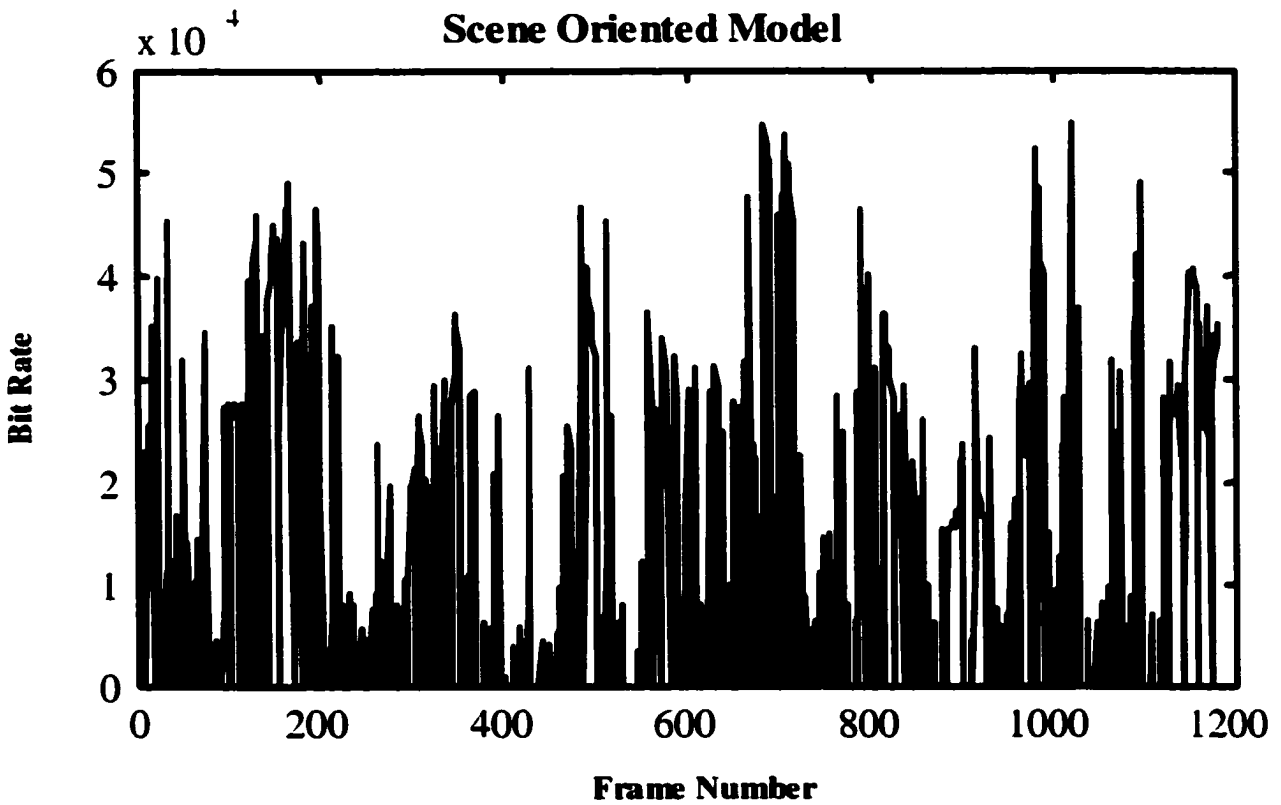


Figure 4.3: GOP size trace generated by the scene-oriented model

The scene-oriented model, based on the variation-based algorithm has been employed to mark a video sequence into separate scenes. The coefficient of variation greater than a preset value constituted a scene change.

4.3. Autoregression studies

A class of models which also has attracted researchers' attention is the class of autoregressive (AR) processes. This class is particularly interesting since a wide range of methods known from time series analysis can be applied for parameter estimation and model characterization.

Besides the Markov models already mentioned above, Maglaris et al, (1988)[19] present a first order AR model. Nomura et al. (1989)[20] also use a first order autoregressive process to model the frame size process of their experimental encoder. In addition, they suggest the use of Markov modulated AR for video sequences consisting of several scenes. Roberts et al. used the Maglaris model for their performance considerations of a multiplexer with VBR video input traffic. A superposition of two first-order AR processes and a Markov chain is considered by Ramamurthy and Sengupta [21]. The first AR process models the short-range behavior, the second AR process the mid-range behavior, and the Markov chain models the single peaks of the frame size process. Chowdhury and Sohraby [22] compare bandwidth allocation algorithms for packet video which is modeled using the Maglaris et al. approach. One of the few models dedicated to Mpeg video traffic is presented by Enssle [23]. He uses three first-order AR processes to model the Mpeg frame types separately. A similar approach is used by Adas [24] to predict the frame sizes of an Mpeg trace. He used high

order AR processes. In contrast to these aproaches, Stokes [25] uses one AR (PG) model for complete sequence, where PG is the number of frames in one GOP.

Autoregressive processes are widely used in the time series analysis literature. We obtain a lognormal marginal distribution which is typical for MPEG video traffic by simply transforming the normal distribution with mean, μ_n and standard deviation, σ_n $N(\mu_n, \sigma_n)$ marginal distribution of a standard AR(P) process by an exponential function. By means of a model order P, we are able to determine the number of lags of the empirical auto correlation curve that are modeled correctly.

4.3.1: Parameter Estimation

Let $\{x_i: i=1, \ldots, N\}$ denote the considered GOP size[15 frames] trace. We assume lognormally distributed GOP sizes. The Q-Q plots indicate that GOPs follow a lognormal distribution. See experimental results in the appendix . We have to transform $\{x_i\}$ to a time series with normal marginals, $y_i = \log x_i$. The parameters of the normal marginal of the transformed process are given by the sample mean μ_n and the sample variance σ_n^2. To determine the model order P, a set of simultaneous equations has to be solved [See Appendix].

4.3.2: Generation of a Model Trace.

Let ε be a $N(0, 1)$ distributed white noise process. The values μ_n and σ_n are set to the given estimates $\hat{\mu}_n$ and $\hat{\sigma}_n$ or in dependence of the expected mean μ_r and the variance σ_r^2 of the model trace.

$\mu_n = \log\frac{\mu_t^2}{\sqrt{\sigma_t^2 + \mu_t^2}}$; $\sigma_n^2 = \log\frac{\sigma_t^2 + \mu_t^2}{\mu_t^2}$. The mean and standard deviation of the transformed trace are related to the mean and standard deviation of the model trace as above(see [17] for general discussion on computing the statistical parameters).

For a model trace of length N, a trace is generated by the Gaussian process $\{t_i^m : i = 1, \ldots, L + N\}$ applying the recurrence relation $t_i^m = \alpha_1 t_{i-1}^m + \ldots + \alpha_p t_{i-p}^m + \in_i$ with $t_i^m = 0$ for $i < 1$ and given the parameters $\alpha_1, \ldots, \alpha_p$. The first L samples are neglected to avoid start-up errors. The value L is determined by comparing the autocorrelation curve of the trace $\{t_i^m : i = L + 1, \ldots, L + N\}$ and the theoretical autocorrelation curve of the AR(P) process. If both curves match well the value L is large enough. The marginal distribution of trace $\{t_i^m\}$ will be gaussian but not with the expected parameters. Following the algorithm, we therefore transform the sample t_i^m to the N (μ_n, σ_n) distributed samples t_i^n as follows:

$$t_i^n = \frac{(t_{L+1}^m - \mu_m).\sigma_n}{\sigma_m} + \mu_n \text{ for } i = 1,\ldots,N$$

with μ_m denoting the mean and σ^2_m the variance $\{t_i^m\}$. Finally the trace $\{ti\}$ = exp t_i^n is generated.

Figure 4.4 shows a trace generated by an AR (10) model with lognormal marginal distribution. The sample sizes follow a continuous distribution.

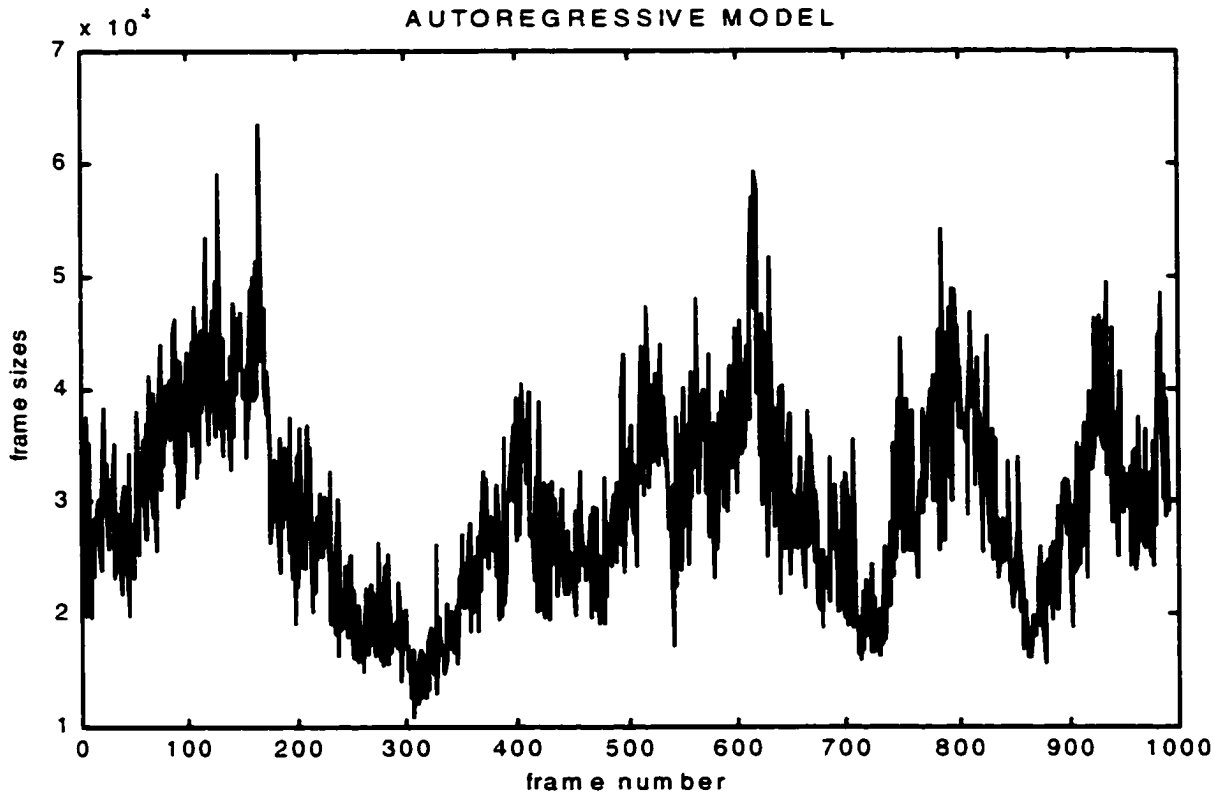


Figure 4.4: Trace generated by AR(10) model with lognormal marginal distribution

The autoregressive model of order P captures frame to frame correlation and is suitable for modeling activities which do not involve much scene changes. The model trace was generated by first transforming the lognormally distributed GOP sizes to follow a normal distribution. An inverse transform regenerated the model trace. We extend this model to capture scene changes in chapter 5.

4.4 Capturing self-similarity

Recently, attention has been paid to self-similar or long-range dependent modeling of traffic in communication networks. Up to now, most papers mainly dealt with the statistical analysis of data sets, i.e., in most cases with the estimation of the Hurst parameter of an empirical sequence. In Norros [26] and Likhanov et al. [27], the authors found that in G/D/1 systems with self-similar input, the queue length distribution does not decay exponentially as in the case of non-short correlated input traffic, but hyperbolically or Weibullian.

Garret and Willinger [13] present a detailed statistical analysis of a two hour VBR video trace and present a FARIMA (0, D, 0) model for video traffic. In Adas and Mukherjee [24], the authors use a FARIMA (1, D, 0) model for their experiments. Enssle [23] suggests to use a Fractional Gaussian Noise (FGN) model for Mpeg video traffic and compares its performance to a white noise process with the same marginals. In contrast to these papers, Huang et al. [28] generate self-similar model traces directly from the autocorrelation function of the Mpeg-I-frame sizes using Hosking's method[14].

Self-similar processes form a class which facilitates the modeling of long-range dependency. All other models are only capable of modeling short-term correlations. FARIMA (0, D, 0) also facilitate the modeling of long-range dependency but offer no possibility to match the models to the low-lag correlations of the data sets. Standard FARIMA processes have a Gaussian marginal distribution. Similar to the autoregressive

models, we obtain the adequate marginal distribution by transforming the N (μn, σn) marginal distribution of the FARIMA process to a lognormal marginal.

4.4.1. Parameter Estimation

Let $\{x_i : i = 1, \ldots, N\}$ denote the GOP size trace. Assuming lognormally distributed GOP sizes, we first have to transform $\{x_i : i = 1, \ldots, N\}$ to a time series with normal marginals $\{ x_i^n : i = 1, \ldots, N \}$ by $x_i^n = \log x_i$. The parameters of the normal marginal and the transformed process are given by mean μ_n and variance σ_{n^2}.

4.4.2 Generation Of A Model Trace

For the generation of Gaussian FARIMA (P, D, 0) trace of length N the two-step algorithm suggested by Hosking [14] was used. The algorithm first generates a FARIMA (0, D, 0) trace length L + N. then, AR(P) is added by applying the appropriate recurrence relation and cut off the first L samples. The whole generation process is equivalent to that of an AR(P) trace besides the fact that ε_i is white noise in the AR(P) case and long-range dependent in the FARIMA (P, D, 0) care for d > o. If D = 0, i.e., H = 0.5, the statistical properties of AR(P) and FARIMA (P, D, 0) traces are the same.

Generation of the FARIMA (0, D, 0) trace: Given d = H – 0.5, let $\{ \varepsilon_i : i = 1, \ldots, L + N\}$ denote the FARIMA (0, D, 0) trace. Set Vo = 1. Choose ε_0 from N (0, Vo). Then generate L + N Points by iterating the following algorithm for K = 1, …., L + N:

$$\phi_{kk} = d/(d-k)$$

$$\phi_{kj} = \phi_{k-1,j} - \phi_{kk}\phi_{k-1,k-j} \quad j = 1, \ldots, k-1$$

$$m_k = \sum_{j=1}^{k} \phi_{kj}\varepsilon_{k-j}$$

$$v_k = (1-\phi_{kk}^2)v_{k-1}$$

Choose each ε_k from N (M_k, v_k).

Generation of the FARIMA (P, D, 0) trace: In the following, the algorithm partly repeats formulae from previous section to obtain a selfcontained model description. Given $\alpha_1, \ldots, \alpha_p$ of the AR(P) part and a Gaussian FARIMA (0, D, 0) trace. $\{\varepsilon_i : i = 1, \ldots, L + N\}$. Set μ_n and σ_n to the given estimates $\hat{\mu}_n$ and $\hat{\sigma}_n$ or independence of the expected mean and variance

$$\mu_n = \log\frac{\mu_i^2}{\sqrt{\sigma_i^2 + \mu_i^2}}; \sigma_{n^2} = \log\frac{\sigma_i^2 + \mu_i^2}{\mu_i^2}$$

The algorithm obtains a trace of the Gaussian FARIMA (P, D, 0) process $\{t_i^m : i = 1, \ldots, L + N\}$ applying the recurrence relation $t_i^m = \alpha_1 t_{i-1}^m + \ldots + \alpha_p t_{i-p}^m + \varepsilon_i$, with $t_i^m = 0$ for $i < 1$. The first L samples are thrown away to avoid start-up errors. The value L is determined by comparing the autocorrelation curve of the trace $\{ t_i^m : i = L, + 1, \ldots, L + N\}$ and the theoretical autocorrelation curve of the FARIMA (P, D, 0) process. If both curves match well, the value L is large enough.The marginal distribution of $\{ t_i^m \}$ is Gaussian but not with the expected parameters.

The algorithm therefore transforms t_i^m to the N (μ_n, σ_n^2) distributed t_i^n by

$$t_i^n = \frac{(t_{L+1}^m - \mu_m).\sigma_n}{\sigma_m} + \mu_n \text{ for i = 1,....,N}$$

with μ_m denoting the mean and σ_m^2 the variance of $\{t_i^m\}$. Finally, the L (μ_n, σ_n^2) trace {ti} is generated from $\{t_i^n\}$ by $t_i = \exp t_i^n$.

Fig 4.5 shows a trace generated by an FARIMA (l, D, 0) model with lognormal marginal distribution. The trace has appealing similarities to the empirical trace. As for the original, its marginal distribution is continuous and the average behavior changes slowly over time.

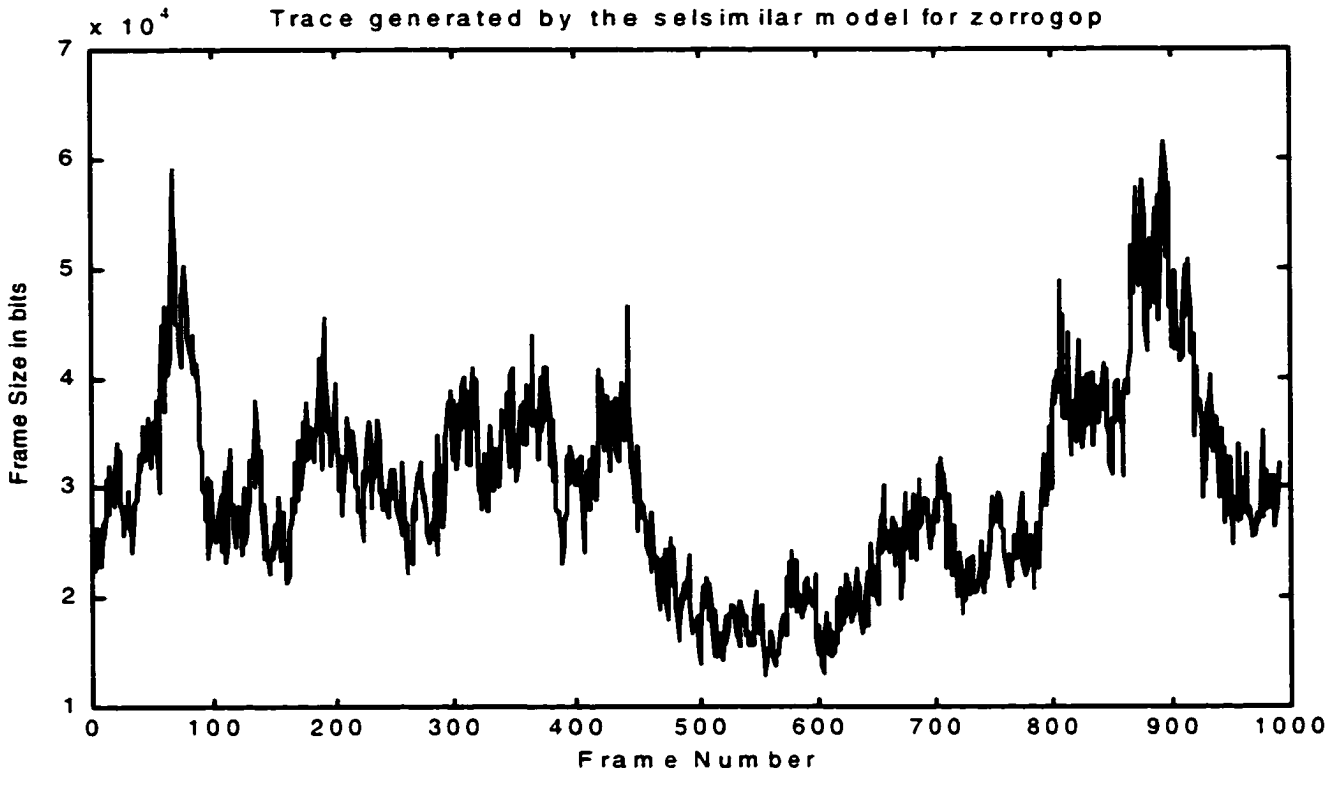


Figure 4.5: GOP size trace generated by the selfsimilar model

Whereas the Autoregressive model captures short-range inter-frame correlation, the self-similar method has been applied to capture long-range dependency in video traffic. By applying the algorithm [14], a model trace was generated. Review of previous works gave us the insight into potential problems hidden in the generation of the model traces. It is also important to review some experimental work in the area of traffic modeling, and bandwidth allocation strategies.

4.5 Measurement-based Analysis

Feng and Rexford[29] presented a comprehensive comparison of bandwidth smoothing algorithms for the delivery of compressed video streams in video-on-demand systems. It capitalized on the apriori knowledge of frame lengths to reduce the burstiness of resource requirements for the playback of prerecorded video by prefetching data as a series of fixed rates at the client buffer. The metrics used included the peak rate requirements, the number of bandwidth changes, the variability of the bandwidth allocations, and the variability of the time between bandwidth changes. They reported that for small buffer sizes, the piecewise constant rate transmission and transport (PCRTT) algorithm was useful in creating plans that have near optimal peak bandwidth requirements while requiring very little computation time to calculate. For larger buffer sizes, however, the PCRTT algorithm limited the ability of the server to prefetch frames across interval boundaries. The critical bandwidth allocation (CBA), minimum changes bandwidth allocation (MCBA), and minimum variability bandwidth allocation (MVBA) algorithms exhibit similar performance for the peak rate requirement and the variability of bandwidth rates. The MCBA algorithm, however, is more effective at reducing the total number of rate changes.

In contrast to stored video, live applications typically have limited knowledge of frame sizes and often require bounds on the delay between the source and the client(s). As a result, smoothing for interactive video typically requires dynamic techniques that can react quickly to changes in frame sizes and the available buffer and bandwidth resources. Rexford et al[30] developed online, window-based

smoothing algorithm for delay-tolerant applications such as videocasts of courses or television news where many clients may be willing to tolerate a playback delay of several seconds or minutes in exchange for a smaller throughput requirement. They developed smoothing constraints to avoid underflow and overflow of the B-byte playback buffer. To lie within the upper and lower constraints, appropriate transmission schedules were computed. They used the Hopping-Window and the Sliding-Window smoothing techniques. The hopping-window algorithms operated on consecutive non-overlapping smoothing windows, each of length W. These algorithms execute every W time units, smoothing the W most recently generated frames. The sliding-window algorithm smoothing executes on overlapping windows of size W. They concluded that the new algorithms significantly reduced the peak rate, coefficient of variation, and effective bandwidth of variable bit rate video streams using window sizes (1-10 seconds). Our work requires prior knowledge of the information about the traffic. Application of real time traffic techniques can utilize our model in real time traffic analysis.

Sohraby et al[31], demonstrated for the first time in an actual local area network, the existence of correlations among monomedia traffic streams in a multimedia session. They measured the correlation between audio and video traffic and reported that such monomedia sources cannot be assumed to be independent, but rather correlated. In Hussain et al [32], A 2-D general independent (GI) arrival process model was developed in which they assumed multiple correlated batches of information units, arriving in the same time slot and entering multiple queues with multiple servers. The result was a multi-dimensional queueing model analysis.

Heyman et al[33] did a statistical analysis and simulation studies of video teleconferencing traffic over ATM networks. They found that periodicity (Constant Video Frame Rate) can cause different sources with identical statistical characteristics to differ in cell loss rates by several orders of magnitude as a result of multiplexing several sources. They found that the number of cells per frame was not normally distributed but followed a gamma (or negative binomial) distribution. In terms of traffic models, Heyman et al[33] found that the Autoregressive Model of order 2, AR(2) and Markov Chain Model were inappropriate because of either underestimating and overestimating the occurrence of frames with a large number of cells. Nevertheless, they realized that a multistate markov chain model that can be derived from three traffic parameters(mean, correlation, and variance) was sufficiently accurate for use in traffic studies.

4.6. Pitfalls and foundation for current work

In Krunz et al [10], scene boundaries were based on the fact that significant changes in the sizes of consecutive I-frames amounted to a scene changes. What is 'significant' was not defined. The paper did not consider how network performance was affected depending on the choice of 'significant' difference that constituted a scene change. The only performance measure used was the cell loss rate, in the case of ATM network. That alone cannot predict how fitting the model is. Frequency analysis of the traffic both of the empirical trace and model trace could be done to unlock the hidden signature of energy distribution of this video traffic, not easily discernible in the time domain.

Feng and Rexford [29] analyzed smoothing algorithms by comparing performance metrics such as peak rate requirements, number of bandwidth changes, variability of bandwidth allocations. The analysis was done in the time domain, and the trace compressed and stored video. Apriori knowledge of frame sizes was important for the algorithm to work. Rexford overcame that limitation by using bandwidth allocation strategies, not on stored video, but on live movie such as videocasts of courses. This method sought to reduce the burstiness in the traffic. This method was for delay-tolerant applications only. The smoothing was done in the time domain to reduce burstiness. No traffic models were used here except the actual traces. Moreover, there was no mathematical characterization of bandwidth allocation versus autocorrelation.

Although Heyman and Laksman[18] did a statistical analysis, their traffic was the class of videoconferencing traffic only. No frequency analysis was done. The AR(2) model did not produce large values for good traffic studies. No earlier methods using the AR(P) model have captured the periodicity exhibited at the frame level traffic sequence.

Summary

In chapter 4, we considered various traffic models such as the histogram model, the scene-oriented model, the self-similar model and the autoregressive model. The histogram-based method gives a simple approach in modeling video traffic. The scene-oriented approach is based on the coefficient of variation of GOP sizes greater

than a preset value. The autoregressive analysis captured the frame-to-frame correlation, whereas the self-similarity captured the long-range dependency. The review of the above traffic models gave us the insight into potential difficulties inherent in modeling Mpeg traffic. The generation of the model trace in self-similar traffic shares some commonalities in the algorithm, and helped us further in our extended model. Our contribution has been an improvement of the autoregressive model. In the next chapter, we describe one improved autoregressive model, and highlight the specific improvements.

5. AN IMPROVED AUTOREGRESSIVE MODEL

In chapter 4, under 'Related Works', the autoregressive model, and how the parameters were estimated was presented. This section also presented how the traces were generated. A trace was generated by the autoregressive model of order 10 AR(10). Below, we present an "Improved Autoregressive Model" that takes into consideration scene changes.

5.1 Overview of AR model

The Autoregressive process has been used by several authors [8, 11, 23] to model the Mpeg video traffic that attempts to capture the frame correlation as well as the gaussian shape of the bit rate variation. However, the autoregressive process alone does not capture scene changes. (See page 61 of [1]). In this chapter, we propose an Autoregressive model of order P, AR(P) + IAP (Interrupted Autoregressive Process), to capture scene changes. We have done so by introducing an I frame according to the mean scene length distribution, as well as the distribution of the bit rate of the I frame. We compared the model performance to that of the actual video trace, as well as the autoregressive process without scene changes. Our model has a mean and standard deviation of I-frames approximately the same as the original trace. **This means that our model captures scene changes, and can be used to study network performance in lieu of an actual trace.**

Many authors have done work on traffic modeling of voice and video. Heffes and Lucantoni [8] presented a Markov modulated Poisson Process (MMPP) that allows the exact queueing theoretic analysis to calculate the performance measures such as cell loss probability, cell transmission delay, etc. In [8], the MMPP model is applied to voice cell traffic. In [34], a four state MMPP model is applied to video cell traffic and the corresponding cell loss is analyzed.

The autoregressive process of order P (AR(P)) models the Mpeg traffic [8, 11, 23] but does not capture the scene changes particularly noticeable in the high bit rates of the intracoded (I) frames [1]. In [35], the authors endeavored to capture scene changes at the cell level. In this paper, we have captured scene changes by capturing the I-frames that is periodically transmitted in the Mpeg Group of Pictures (GOP), which is a pattern that repeats itself throughout the sequence. Recall that scene changes are mostly captured in the high bit rates of the intracoded (I) frames. An example of a 15-frame GOP is IBBPBBPBBPBBPBBI... Additionally, we present a performance analysis of our model and compare the result with that of the original trace and the AR(P) without scene changes. The performance of the AR(P) + IAP approximates closely that of the original trace.

The autoregressive model was presented earlier (section 4.3.1 and 4.3.2), is referenced with regard to parameter estimation, and model trace. In section 5.2, an improved version of the autoregressive process, that accounts for the scene changes is captured. Statistical justification for the improved version of the autoregressive process is presented, and section 5.3 describes a performance analysis of the original trace and

the new model, and compares the result to the AR(P) model. The chapter concludes with section 5.4 describing how our model is applied in a real network.

5.2 Our approach

The Mpeg2 video traffic exhibits periodicity as a result of its Group of Picture (GOP) structure. A typical gop structure is shown below:

I B B P B B P B B P B B P B B I B B P B B P B B P B B P B B I...

The I, P, B frames were previously defined. Whenever there is a scene change, more bytes are pumped into the intra-coded frames (I-frames) by the encoder before it settles into a more normal production. A "considerable" [10] change in consecutive I-frames can be attributed to a scene change. Mathematically, this can be represented as:

$|I_{i+1} - I_i| > \varepsilon_{thresh}$, where I_i is the i^{th} I-frame, and I_{i+1} is the $(i+1)^{th}$ I-frame; ε_{thresh} is the minimum number of bytes difference (minimum threshold) for a scene change. The autoregressive model alone does not capture such periodicity in the I-frames as shown above. Below we present the zorroty1 empirical trace versus the autoregressive model trace to illustrate this point.

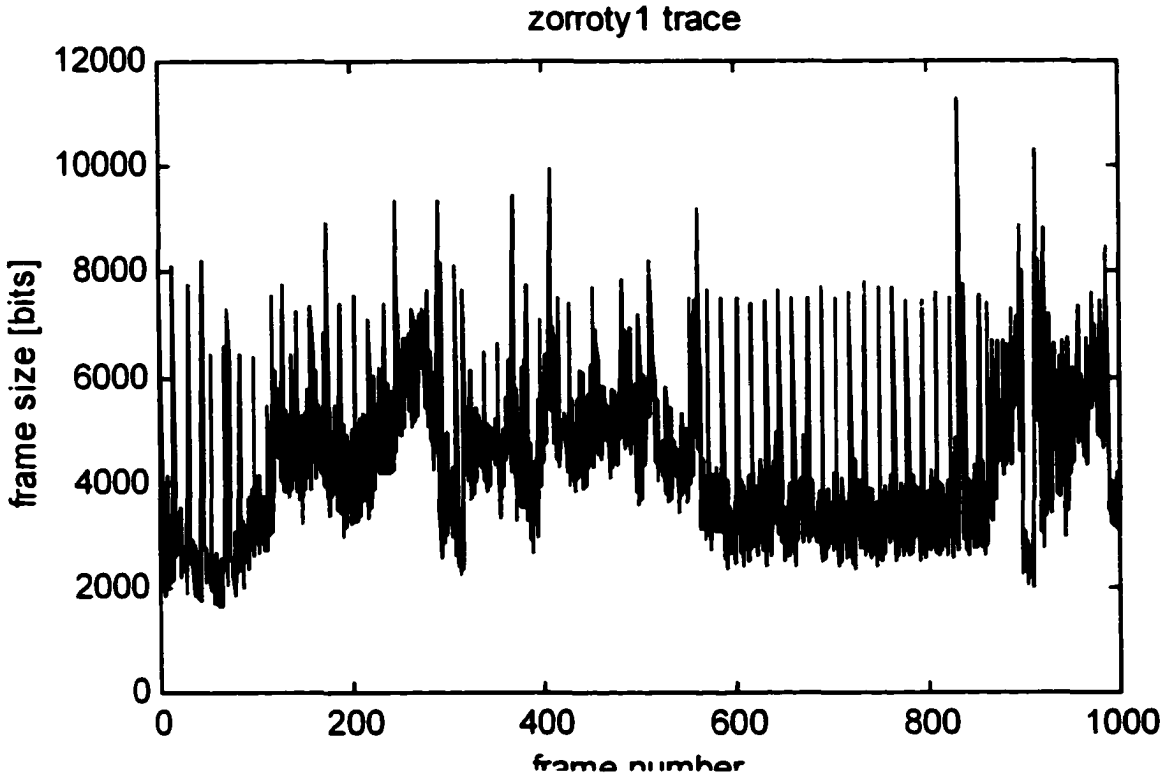


Figure 5.1: zorroty1 frames showing periodicity in I- frames

Notice the periodicity in the GOP structure by the periodic spikes in figure 5.1. Such periodic spikes are not captured in figure 5.2. To capture this periodicity, we introduce a new idea called AR(P) + IAP(S,F), where IAP is defined as IAP=mun + std x randn(0, 1); mun= mean of the I-frames, std = standard deviation about the mean. The video clips we collected has a GOP of 15. This means that every 16^{th} frame, an intracoded frame is generated. If there is a scene change, the bit rate dramatically increases, and is captured in the I-frame. If the change from the previous intracoded frame is

"considerable", an IAP is introduced. Note that randn(0, 1) generates normally distributed random numbers with mean zero, and standard deviation 1. IAP is a function of the mean scene length distribution, S, and the mean distribution of the I frames F. By inserting an I frame according to IAP(S F), we can model the original trace much closer to capture periodicity and scene changes. Care must be taken in choosing the standard deviation about the mean. By try and error, we can choose the appropriate standard deviation that closely models the original trace. The graph below shows our initial analytical result.

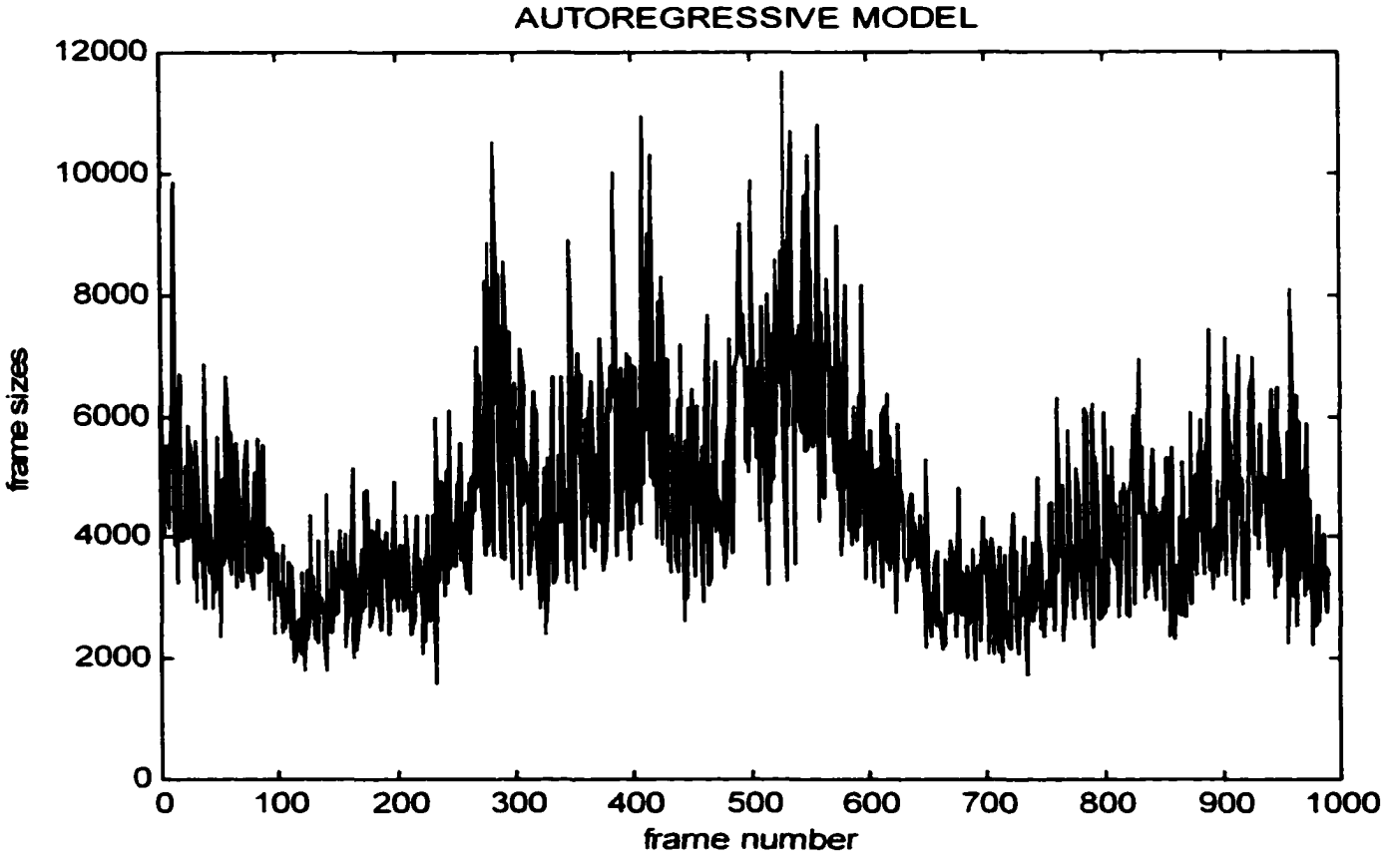


Figure 5.2: zorroty1 model trace without periodicity

By choosing the appropriate standard deviation between 0.2~0.3 of the mean value, the mean of the frames approximates that of the original. Between 200 and 500 frame

numbers, the average of original is 5000 bits, and our model is 4000 bits, whereas between 600 and 900 frame numbers, the mean of the original trace is 4000, whereas our model has mean 5000 bits. Within certain intervals our model overpredicts or underpredicts the mean values (see figures 5.1 and 5.3).

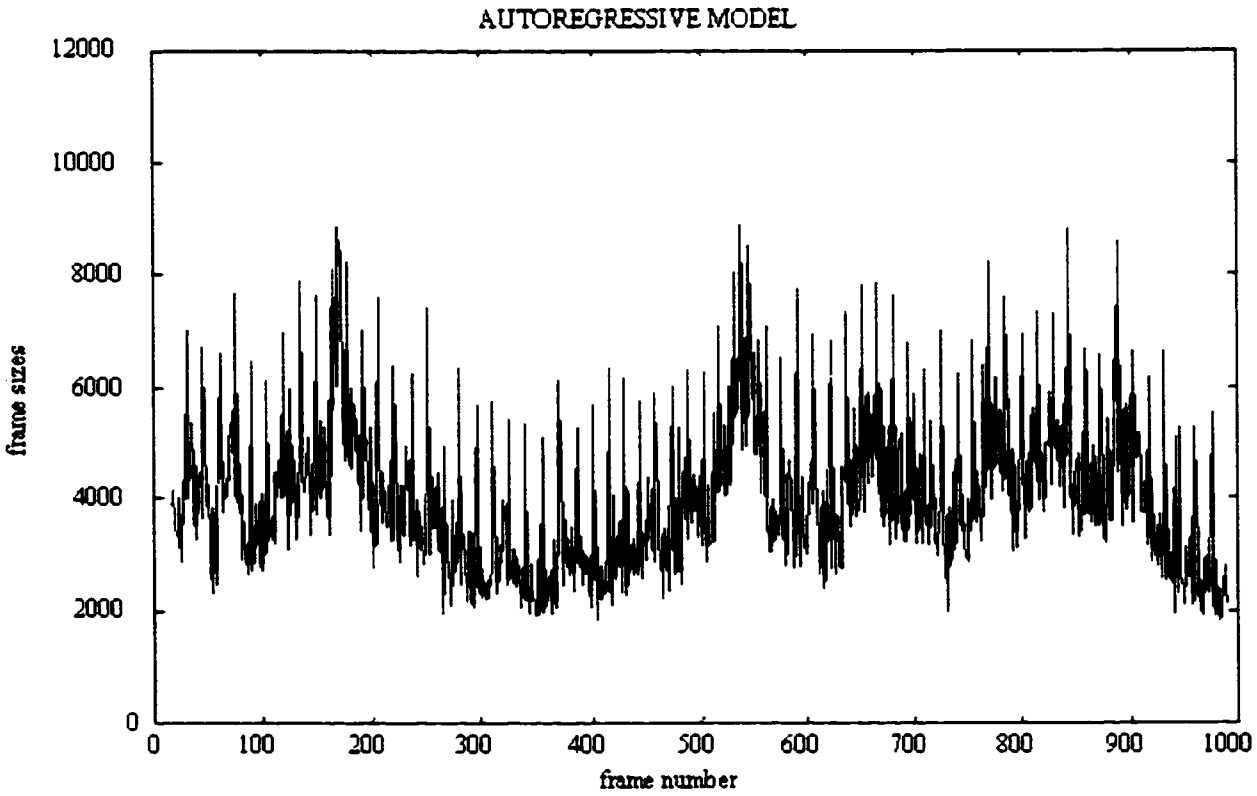


Figure 5.3: Autoregressive model of zorroty1 with periodicity captured

To see how our model captured scene changes, the I frames were examined in terms of the mean and standard deviation for the entire sequence. The mean of the I frames in our model is 520750 bits and is closer to the mean (530980 bits) of the original trace. The same is true for the standard deviation. The AR(10) has a lesser mean and standard deviation. This means that the mean distribution of the I frames in AR(10) is shifted to

the left, implying fewer I frames that met the "considerable " change criterion for a scene change. Table 5.1 presents the results.

Table 5.1: Shift to the left of I frame statistics for AR(10) model.

Data	Mean_I (bits)	Std_I
Original	530980	641390
AR(10)	427770	451830
AR(10)+IAP(S.F)	**520750**	**640150**

In section 3.2, under distributions, it was established that frame size distributions of the I-frames could be approximated by a normal distribution. The QQ plot was used to emphasize this conclusion. Hence we inserted an I-frame according to the mean scene length distribution, using the normal distribution of the I frames according to IAP = mean + standard deviation (about the mean) of the I-frame distribution.

5.3 Performance Analysis

To further test the validity of our model, we examined the performance in a queueing network using three different inputs, namely the original trace, our model and the AR model without scene changes. The network setup to measure loss rate is shown in figure 5.4 [36]. The "Loss_Node" node measures losses in case of buffer overflow. The service discipline was first come first served (FCFS. A buffer size of 20 cells (in this case of ATM), and an output transmission capacity of 155Mbps was used. Below,

we present the loss probability (**ratio of rejected cells due to buffer overflow, to the total number of arrivals in the ATM multiplexer. In the case of packet-switched networks with variable or fixed sizes, the loss probability is the ratio of rejected bytes due to buffer overflow, to the total bytes arriving at the multiplexer**) of the original trace, the AR(P) + IAP and AR(P). The loss probability was calculated as the ratio of the losses in the "Loss_Node" to the total number of arrivals after several hours. Table 5.2 shows the results for the various inputs.

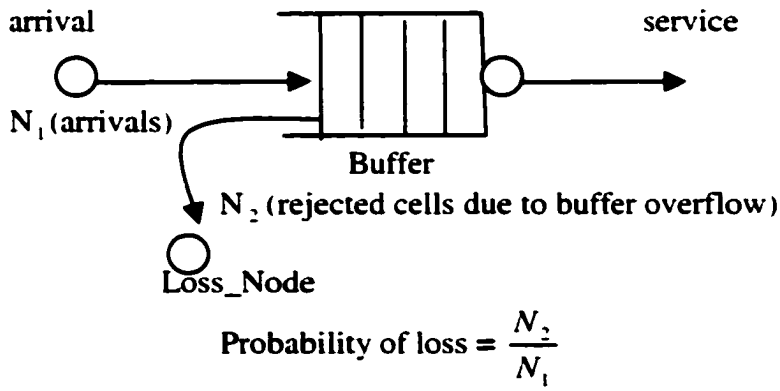


Figure 5.4: Measuring Loss Rate

When passed through the queueing system above, the loss rate for the original trace was .0035. The loss for our model was .0033, whereas the loss for the model without IAP was .0015. The loss rate of our model is within 94% of the cell loss rate of the original, whereas the loss rate of the AR(P) without scene changes is only 43% of the original trace loss. A good model should yield approximately the performance measures of the original trace. Since the loss probability of our model and the original trace is very close, the model statistically captures very well scene changes.

Table 5.2: Comparison of cell loss probabilities.

Data	Loss Probability
Original Trace	.0035
AR(10)	.0015
AR(10+ IAP(S,F)	**.0033**

5.4 Application Of Our Model In Real Network

We can apply our model in a real network, where the traffic source is an Mpeg Video Encoder. Traffic descriptors such as the peak rate (p), average rate (avg), standard deviaton (σ) are derived from the traffic. These parameters become the arguments for the traffic synthesizer, where we apply our model. A synthetic traffic is generated as input to the network simulator, see figure 5.5. To meet a particular quality of service (acceptable loss), bandwidth B is allocated to this traffic.

We define the allocation ratio as:

$$\text{Allocation ratio} = \frac{Actual \quad bandwidth \quad allocated \quad by \quad network(B)}{Peak \quad Rate}$$

If the allocation ratio is unity, this will imply that the lost traffic is equivalent to zero, and the peak bandwidth is allocated.

One of the challenges VBR traffic presents to bandwidth allocation is the random fluctuations of bit rates, due to scene changes. Since our model captures scene changes, network designers will be provided with mean scene length distribution, in addition to the mean, peak and standard deviation of the observed I frames. This knowledge will help with allocating bandwidth. As shown in figure 5.5, a buffer may be needed to store packets due to the bursty nature of VBR traffic.

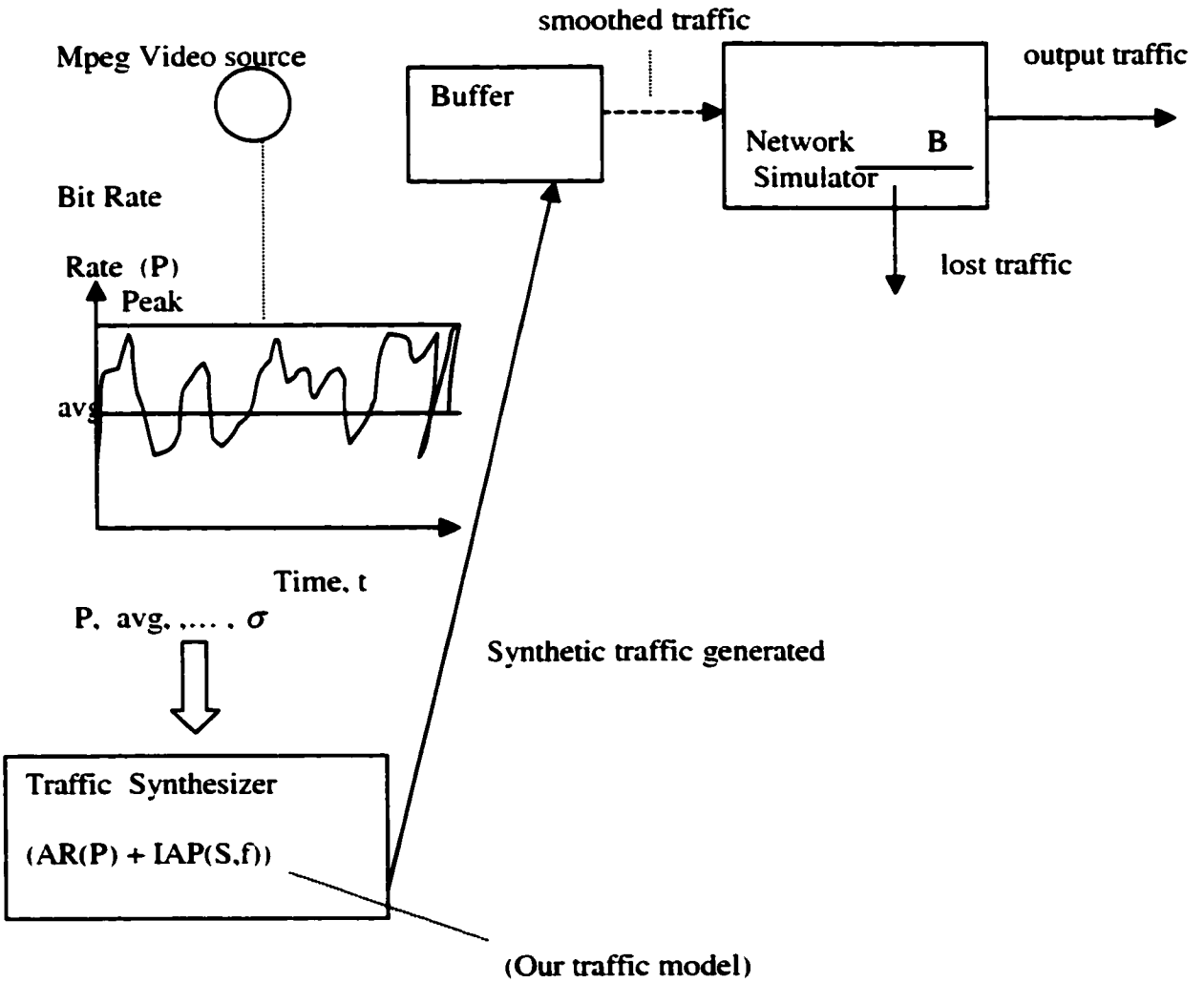


Figure 5.5: How our traffic model fits in a network

Summary

An autoregressive model that captures scene changes in the Mpeg-2 group of picture sequence has been proposed. Such scene changes are more noticeable in the increased bit rate of the intracoded frames. The probability of loss of the proposed model closely approximates that of the original trace, and the performance of the autoregressive model without scene changes is also presented. This model may be used as input in simulating a network and determining the performance in lieu of actual video clips. In the future, a robust design in terms of the range of standard deviations about the mean value of the intracoded bit rate that results in robustnessness of the model will be investigated. Also, the range of 'thresholds' for a scene change to occur will be investigated.

Our next contribution has been formulating **a relation between correlation and effective capacity**. Chapters 6, 7, and 8 will deal with our second contribution.

6. A RELATION BETWEEN CORRELATION AND EFFECTIVE BANDWIDTH (SINGLE SOURCE).

Having considered various traffic models, we want to now concentrate on the impact of such traffic models on network performance. Of particular interest is the characterization of effective capacity and the correlation that exists within a stream and among sources.

In the past, most effective capacity calculations have assumed the sources to be independent, not correlated. The effect of correlation, both intra and inter source, can no longer be neglected, as they may have a significant impact on system performance. We have developed an analytical relation between correlation and effective capacity and have substantiated through simulation, that correlation impacts the effective capacity in significant ways, and can greatly impact network performance, such as cell loss.

6.1 Effective capacity without correlation

The effective capacity formula developed by Elwalid and Mitra [3], F.P. Kelly [4], Anick et al [5], Gibbens and Hunt [6], and Kosten [7], consider the sources to be independent, i.e. the effect of correlation, both intrastream and interstream, is not considered. The effective capacity formulation as a function of correlation for a single source is considered in this chapter. Since intra-frame correlation, as in video[31,32,34] exists, the new effective capacity formula should take into account the intra-frame-correlation of the source. This can be expressed as:

C_eff ' = f (C_eff, corr (C_i, C_j)), where corr(C_i, C_j) describes the correlation between frames. It is further shown, that as the correlation approaches zero, the effective capacity formulation boils down to the formulation in [3-7]. As explained in detail in chapter 7, section 7.2, case2, it is possible for the effective capacity as a function of correlation to be below the steady state probability. In that case, $0 < C_{eff} \leq C_{eff}$.

In section 6.2, the effective capacity formula that incorporates correlation is developed. It is further shown analytically, that the formula has the same asymptote as that of [3-7].

In section 6.3, we apply our effective capacity notion to an actual video clip. Since our formula is based on the two-state on/off source model, with transitions that are exponentially distributed, we first convert our video clips into 'on' and 'off. We test to determine if these 'on' and 'off' times are exponentially distributed. This then allows us to use our formula to calculate the effective capacity for the video source. In our particular case, we had two on/off sources accessing a common buffer. The question is: does their combined effect have on/off transitions which are exponentially distributed? Can we use our formula to calculate the effective capacity as a function of some correlation for the two sources accessing a common buffer? How does it compare with the original combined source? Several on/off sources accessing a common buffer are then treated as one source. We have shown that the combined (equivalent single source) has 'on' and 'off' times which are approximately exponentially distributed. We have, particularly, developed a cumulative distribution function for the sum of two

exponentially distributed on/off sources. The analytic cdf results agree with measurements. We have then tested our formulation by simulation and by using actual video clips.

We examine the performance of the traffic generated by our analytical expression by comparing its probability of loss with that of the actual video trace and that of the steady state. Finally, we draw various conclusions.

6.2 Developing Effective Capacity Formula

An on/off source switching randomly at the rates α (off to on) and β (on to off) states, has a transition matrix, Q:

$$Q = \begin{bmatrix} -p_1(t) & p_1(t) \\ p_2(t) & -p_2(t) \end{bmatrix}, \text{ where}$$

$p_1(t)$ and $p_2(t)$ are the transition probabilities of being in the 'on state' and 'off state', respectively, at time t, and are given by [37]. The transition parameters β and α are exponentially distributed, and R_p is constant.

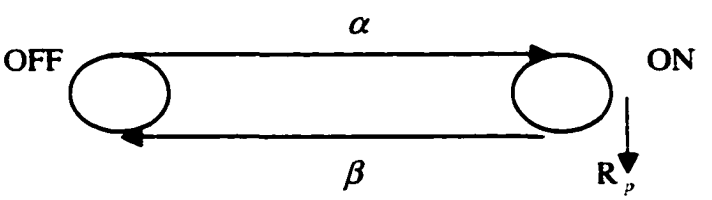


Figure 6.1: on/off source showing transition probabilities

This model could be applied to figure 1.2, where a buffered VBR traffic is transmitted over a constant output link. The 'on' times are the times taken to transmit, and the "off" are the silent period before the next transmission.

$$p_1(t) = \frac{\alpha}{\alpha + \beta}[1 - e^{-(\alpha+\beta)t}] + p_1(0)e^{-(\alpha+\beta)t}$$

$$p_2(t) = \frac{\beta}{\alpha + \beta}[1 - e^{-(\alpha+\beta)t}] + p_2(0)e^{-(\alpha+\beta)t}$$

In addition, the source steady state probabilities $\pi = [\pi_1 \quad \pi_2]$ can be calculated from the relation $\begin{cases} \pi Q = 0 \\ \sum_i \pi_i = 1 \end{cases}$

$Q^* = Q(\alpha+\beta) + \theta\Lambda$, where Λ = diagonal of rates λ_1 and λ_2. In this case $\lambda_1 = 0$ and $\lambda_2 = R_p$, the peak rate. To find the eigenvalues of the transition probability matrix Q^*, we set the determinant of $(ZI - Q') = 0$.

To calculate the effective capacity C_eff, the maximum eigenvalue, Z, is divided by θ, where θ =Log (1/PL)/x; PL is the probability of loss and x is the variable denoting the buffer size [1]. This results in the following equation:

$$(ZI - Q') = \begin{bmatrix} z + p_1(t) & -p_1(t) \\ -p_2(t) & z + p_2(t) - \theta R_p \end{bmatrix}$$

The determinant of $(ZI - Q')$ results in the quadratic equation below:

$$z^2 + z(p_1(t) + p_2(t) - \theta R_p) - \theta R_p = 0$$

After some algebraic manipulations with expressions of $p_1(t)$ and $p_2(t)$ substituted, we get, for a two-state Markov Chain, the effective capacity:

$$C_{eff} = Z/\theta = \left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta} + \frac{(\alpha+\beta)e^{-(\alpha+\beta)t}}{2\theta} - \frac{(\alpha+\beta)[p_1(0)+p_2(0)]e^{-(\alpha+\beta)t}}{2\theta}\right] +$$

$$\sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta} + \frac{(\alpha+\beta)e^{-(\alpha+\beta)t}}{2\theta} - \frac{(\alpha+\beta)[p_1(0)+p_2(0)]e^{-(\alpha+\beta)t}}{2\theta}\right]^2 - \frac{R_p}{\theta}[\{\alpha(1-e^{-(\alpha+\beta)t})\} + p_1(0)e^{-(\alpha+\beta)t}(\alpha+\beta)]}$$

--(6.1)

The exponential terms " $e^{-(\alpha+\beta)t}$ " capture the transient behavior. As $t \rightarrow \infty$, the effective capacity reaches a steady state value. This becomes, after some algebra,

$$C_{eff} = \frac{Z}{\theta} = \left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right] + \sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right]^2 - \left[\frac{\alpha R_p}{\theta}\right]} \text{ ---------(6.2)}$$

Observe that equation (6.2) is time independent and depends on the steady state transition rates α, β, the peak rate R_p and θ.

6.2.1 : Factoring in correlation effect

The autocovariance function for a two-state source can be written as, [1] :

$$Rxx(t) = MA^2 p(1-p) \exp(-(\alpha+\beta)t).$$

Here, M represents the number of quantization levels and A represents the quantized state transition rate. The exponential factor can be written as, $\exp(-(\alpha+\beta)\tau) = Rxx(t)/p(1-p)MA^2$. Inserting this into equation (1), and realizing $p_1(0) + p_2(0) = 1$, we have after some algebra,

$$C_{eff}(\text{corr}) = \frac{Z}{\theta} = \left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right] +$$

$$\sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right] + \frac{R_p}{\theta}\left[\alpha\left[1 - \frac{Rxx(t)}{MA\text{^}2p(1-p)}\right] + p_1(0)(\alpha+\beta)\frac{Rxx(t)}{MA\text{^}2p(1-p)}\right]} \quad \text{----(6.3)}$$

As Rxx(t) approaches zero, with t tending to infinity, equation (6.3) becomes the steady state effective capacity. This means that the effect of correlation is zero, and the effective capacity depends upon the steady state source parameters. Recall that the autocovariance is the autocorrelation less the first moment squared, and is a measure of the correlation between frames.

Figure 6.2 shows the analytical plot versus the time lag in sec.

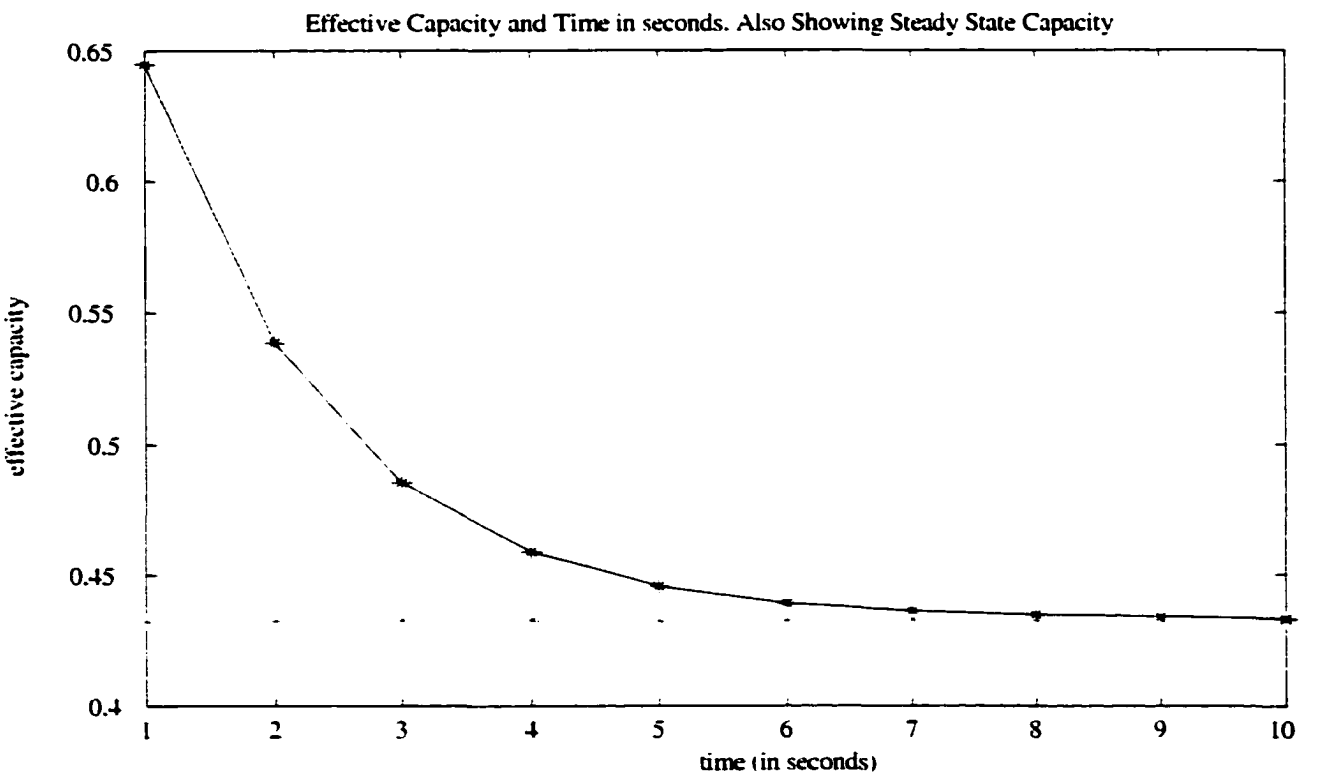


Figure 6.2: Analytic effective capacity with asymptote as steady state

6.3. Determining on/off parameters for video clips

The effective capacity formula assumes that the source is a two-state on/off, and that the transition times from 'off ' to 'on' and vice versa, are exponentially distributed. The Mpeg-2 frame rate is 30 fps. This means that there is 33ms or 1/30 seconds in between two consecutive frames. To find the distribution of the on/off states, two random variables are defined:

x ≡ time taken for source to emit frame ('on' period)

y ≡ .033 - x , off period time before next frame

The architecture of the transport system of Mpeg-2 video is presented in figure 1.2 and 1.3. The output is a variable bit rate with fixed duration. This is the encoder phase of the transport system. The Scheduler phase shows variable bit rates buffered and transmitted over a constant rate transmission line. The 'on' period is the transmission time and the 'off' period is the off time before the next transmission.

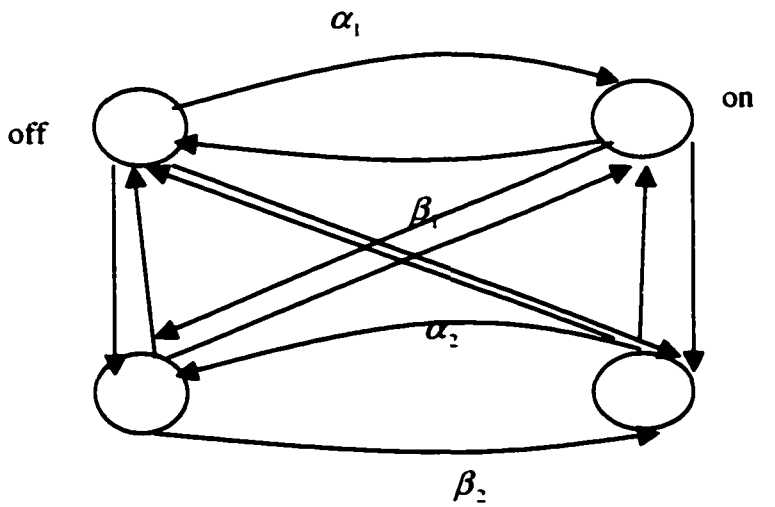
6.4 Probability Distribution for two on/off sources.

6.4.1: Simulation

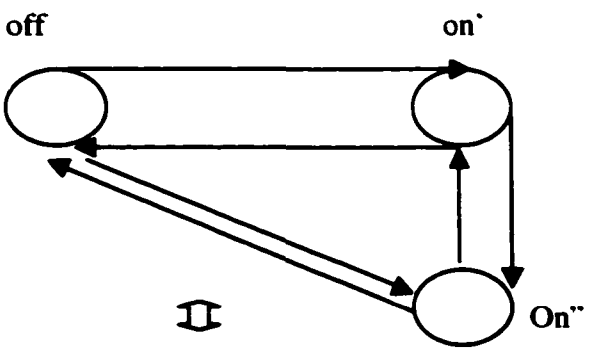
Correlation effect of two random sources and their traffic descriptors. We first develop an equivalent source model for two traffic sources. We then calculate the on/off equivalent transition rates. The two sources then appear to the buffer as a single source.

6.4.1.1: Equivalent source model for two sources

As shown in figure 6.3 below, each source can be represented by a two-state "on-and-off" minisource with transition rates as shown. The **inter** stream correlation is further shown by the arrows cross-connecting the independent sources. We further collapse these two-media sources into a single source with three states namely: "off state 0", "on state 1", and "on state 2". We then constructed rate diagrams to capture the different states in our equivalent model.



(a)



(b)

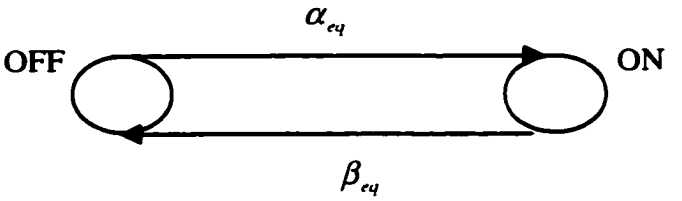


Figure 6.3: correlated sources collapsed into 3 states OFF, ON', ON'' and showing equivalent model as a single source

The rate diagram constructed for each source s1 and s2 and the combined or correlated source is represented below: Additionally, bandwidth (B) is allocated. Combined bandwidth from all the sources exceeding B will lead to packet losses. Proper scheduling will lead to full usage of the channel, and hence multiplexing gain.

Source 1 has an average off period of $1/\alpha_1$ and an average on period of $1/\beta_1$. Similarly, source 2 has an average off period as $1/\alpha_2$ and average on period of $1/\beta_2$. For the sake of simplicity in our simulation, we chose the amplitude of each source to be 1. This then gave us the amplitude of state 2 as 2 and state 1 as 1.

The purpose of the simulation studies was to find out whether the ON and OFF periods in the combined source s3, was also exponentially distributed. To this end, we generated random variates which were exponentially distributred for the 'off' state in source 1 and similarly for the 'on' state in source 1. A 'vector' of 'off' and 'on' states was formed by alternately mingling the 'off 'and 'on' random variables generated. The off periods were assigned zero amplitudes and the on periods, an amplitude of 1. We went through a similar procedure to create a vector s2. By 'combining' the vectors(see figure 6.4) of the same length, we formed the combined vector s3=s1+s2. The final thing we did was

to decompose s3 into a vector of 'off's and 'on's. We then analyzed the distribution of the OFF and ON states. Our results indicated that the OFF and ON states in the combined sources for large samples (N>1100) exhibited exponential distribution.

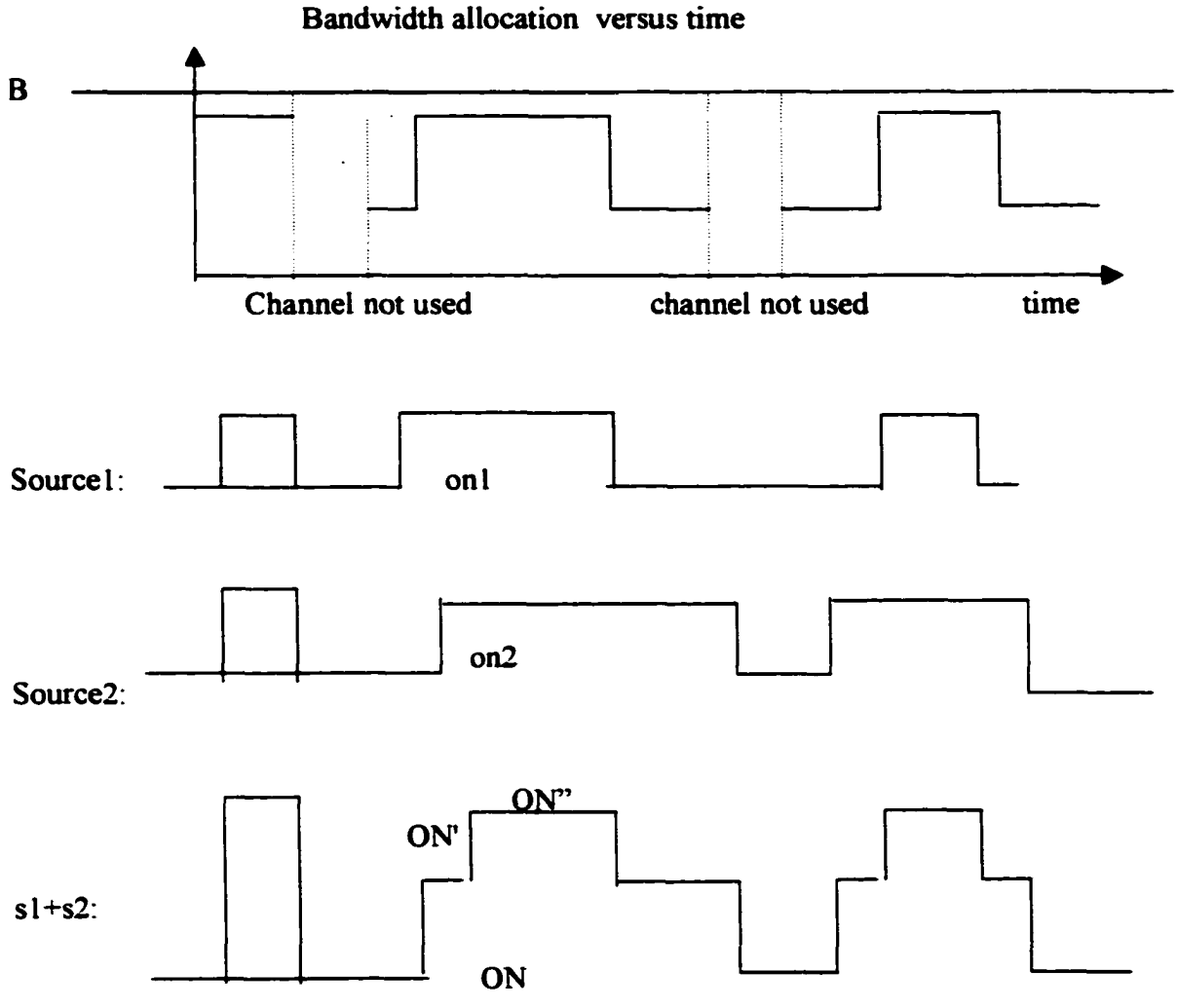


Figure 6.4: Rate diagrams for sources 1 and 2, and their combined effect s3=s1+s2

6.4.1.2: Simulation studies of 2 correlated sources.

We further analyzed the distribution of times in states 1 and 2 separately. These further showed exponential distributions. The following graphs demonstrate our results.

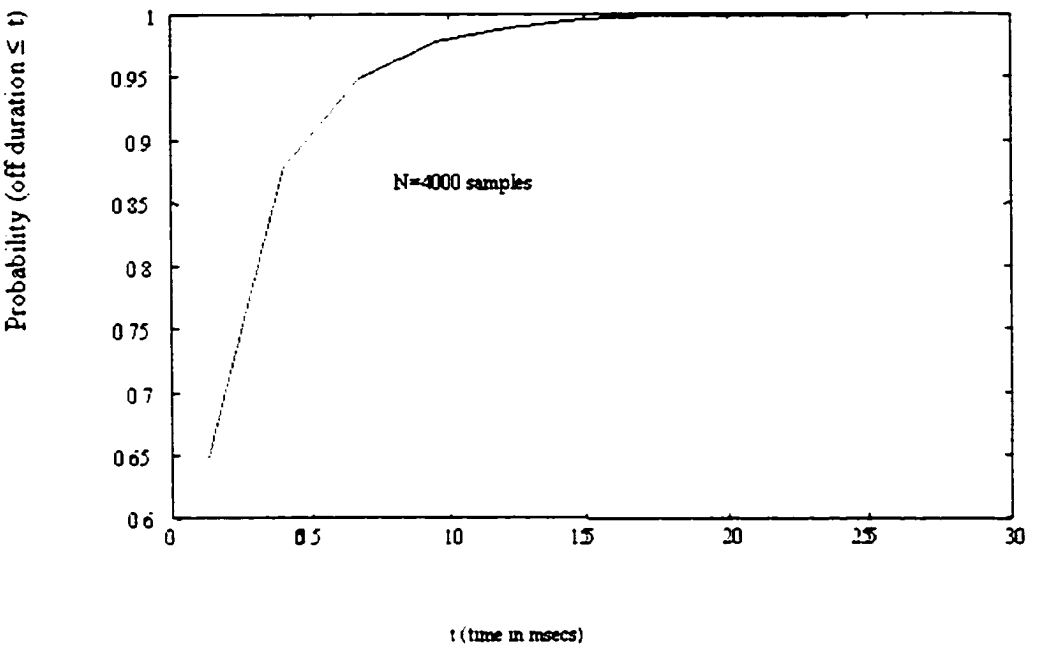


Figure 6.5: Probability distribution of the OFF duration

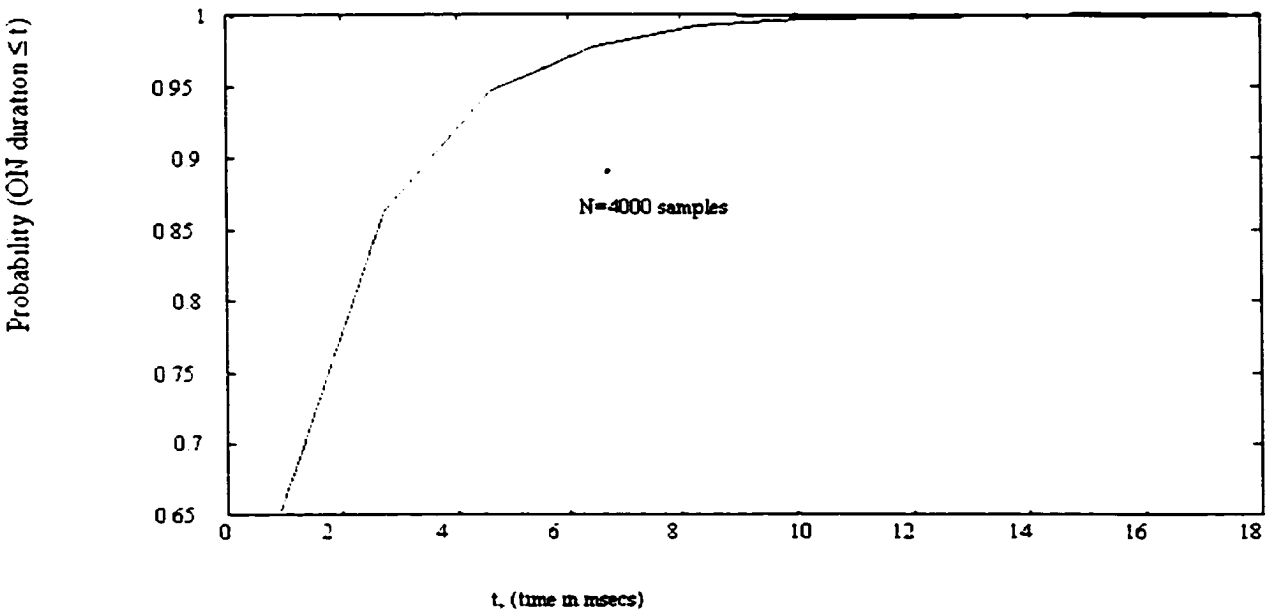


Figure 6.6: Probability distribution of ON duration in combined source s3

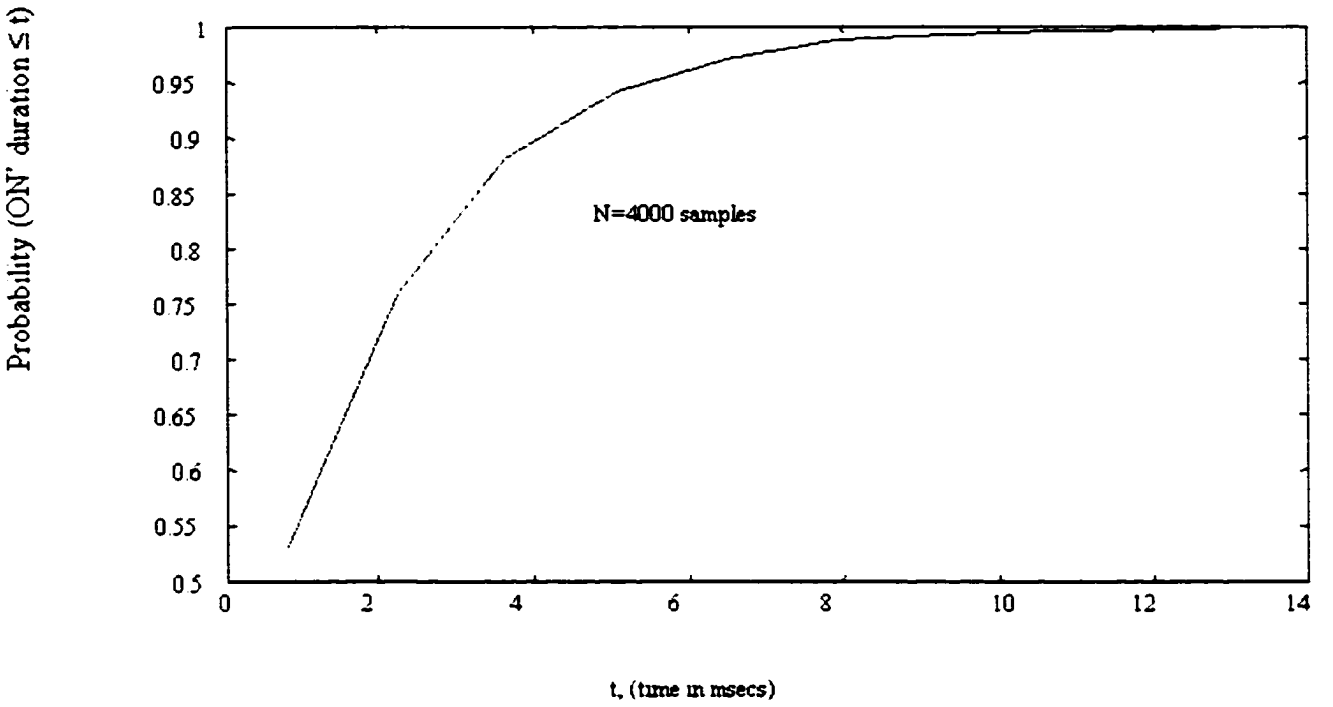


Figure 6.7: Probability distribution of the ON' duration in combined source s3

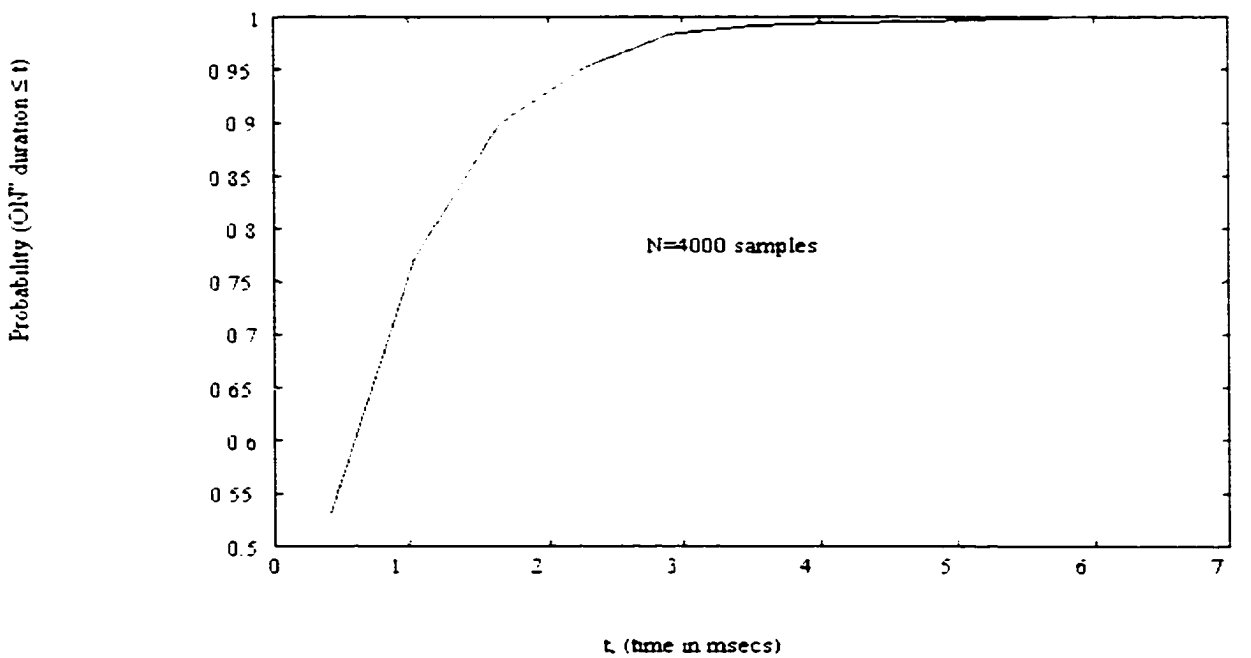


Figure 6.8: Probability distribution of the ON" state 2.

6.4.2: Analytical Method

Consider two on/off sources with 'on' and 'off' times exponentially distributed. We would like to determine the 'on' and 'off' distribution of the combined source. This then allows us to calculate the effective capacity of two sources accessing a common buffer. To this end, we derive the cumulative distribution of two on/off sources, and apply the formula to two video clips, and then compare their distributions.

Let $x_1 = \begin{cases} 0 \\ 1 \end{cases}$ and $x_2 = \begin{cases} 0 \\ 1 \end{cases}$ be two random numbers. The rate diagrams of these sources constructed from the random numbers generated from the definition above, are presented in figure 6.4, above.

We want to find the probability distribution of both sources being in the 'off state'. We can represent this mathematically as Pr (first source is off and second source is off}. Since these are independent events, this is equal to Pr{first source is off }*Pr{second source is off.}. The probability distribution of 'off' in the combined state can be written as Prob(off$\leq$x). Substituting the expressions, we have:

$$\text{Prob}(\text{off} \leq x) = \Pr\{x_1 = 0\}\ \Pr\{x_2 = 0\}$$

$$= \Pr\{x_1 \geq x \text{ and } x_2 \leq x\} + \Pr\{x_1 \leq x \text{ and } x_2 \geq x\} + \Pr(x_1 \leq x \text{ and } x_2 \leq x\}$$

$$= \Pr\{x_1 \geq x\}\ \Pr\{x_2 \leq x\} + \Pr\{x_1 \leq x\}\ \Pr\{x_2 \geq x\} + \Pr(x_1 \leq x\}\ \Pr\{x_2 \leq x\}$$

$$= (1 - \Pr\{x_1 \leq x\}\ \Pr\{x_2 \leq x\} + \Pr\{x_1 \leq x\}(1 - \Pr\{x_2 \leq x\} + \Pr(x_1 \leq x\}\ \Pr\{x_2 \leq x\}$$

$$= [1 - \Pr\{x_1 \leq x\}]\ \Pr\{x_2 \leq x\} + \Pr\{x_1 \leq x\}\ [1 - \Pr\{x_2 \leq x\}] + \Pr(x_1 \leq x\}\ \Pr\{x_2 \leq x\}$$

$$= [\exp(-x/\lambda_{off1})]\ [1 - \exp(-x/\lambda_{off2})] + [1 - \exp(-x/\lambda_{off1})]\ [\exp-(x/\lambda_{off2})] * [\ 1 - \exp(-$$

x/λ_{off})]*[1-exp(-x/λ_{off2})] . This is more like hyperexponential (6.4)

The average off period for source 1 is denoted by $1/\lambda_{off1}$ and the average off period for source 2 is denoted by $1/\lambda_{off2}$. The average on period for source 1 is denoted by $1/\lambda_{on1}$ and the average on period for source 2 is denoted by $1/\lambda_{on2}$.

Substituting the values for the 'off' parameters, we get the above probability of 'off' distribution in the combined state shown in expression (6.4) above. Figure 6.9 depicts the cumulative probability distribution of the ' off duration' in the combined source in the case of two sources.

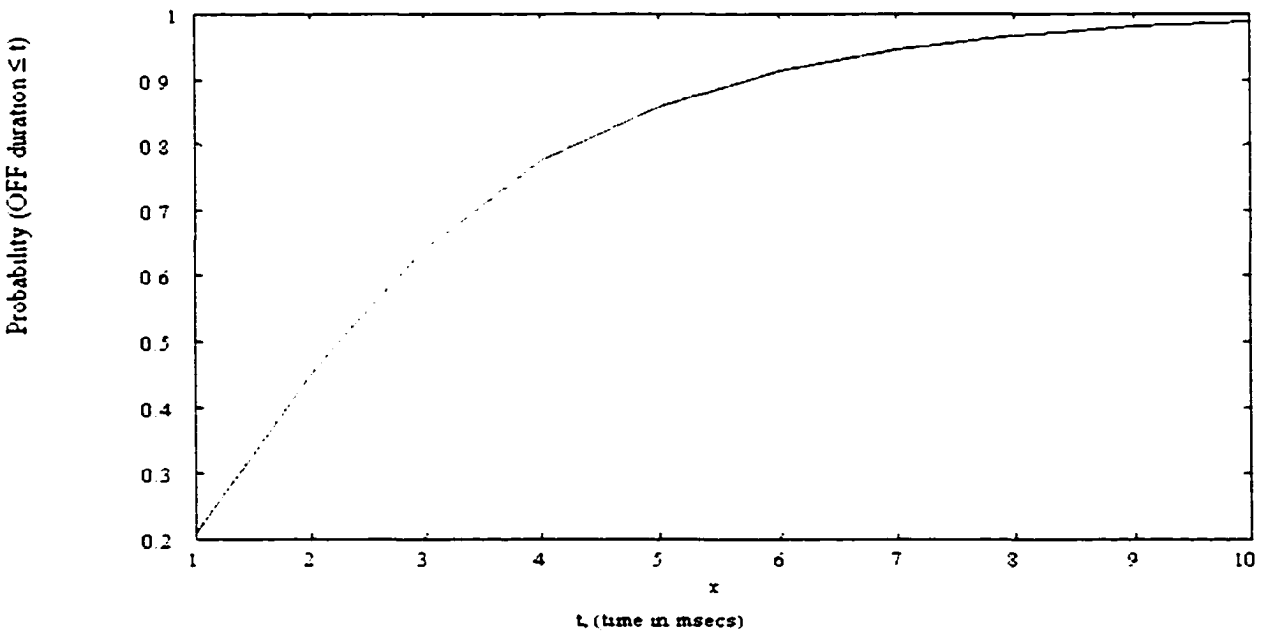


Figure 6.9: Probability distribution of OFF duration-analytical approach

Having found the probability distribution of the 'off state', we now derive a similar expression for the 'on state' for the combined source. Still, using the definition of the random variables, the probability of an 'on state' occurs when one of several cases

occur: when both sources are on; when source one is on and source two is off, or when source one is off and source 2 is on. Writing this probabilistically, we get:

$$\text{prob (on} \le x) = \Pr\{x_1=1\}\Pr\{x_2=1\} + \Pr\{x_1=1\}\Pr\{\text{source 2 is off}\} + \Pr\{\text{source 1 is on}\}\Pr\{x_2=1\} = \{1-\exp(-x/\lambda_{on1})\}\{1-\exp(-x/\lambda_{on2})\} + \{1-\exp(-x/\lambda_{on1})\}\{\exp(-x/\lambda_{off2})\} * \{\exp(-x/\lambda_{off1})\}\{1-\exp(-x/\lambda_{on2})\} \quad (6.5)$$

Substituting the values for the 'on state' and 'off state' for sources 1 and 2, we get the expression in equation (6.5). Figure 6.10 shows that the cumulative probability distribution of the 'on state' is exponentially distributed.

To complete this exercise, we then compared the analytical results with our simulation results. For the simulation, we used N = 8000 samples. Figure 6.11 depicts a comparison of the 'off state' in the combined state for both the analytical method and the simulation method. Similarly, we compared the analytical and simulation approach for the 'on state' in the combined state. The result is depicted in figure 6.10.

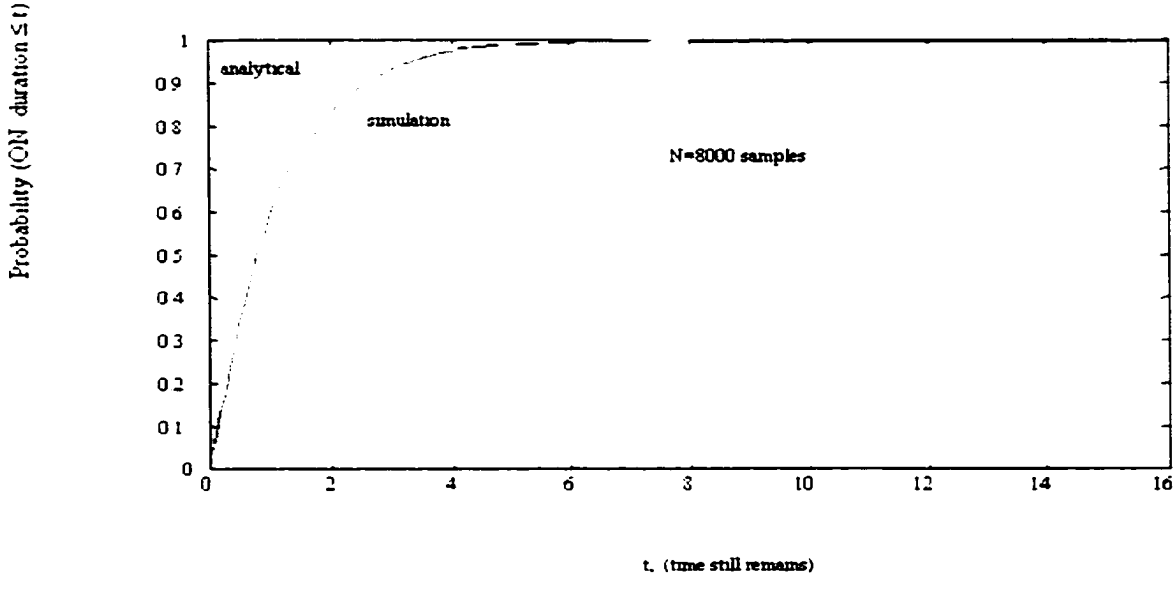


Figure 6.10: Probability distribution of the ON state in the combined source

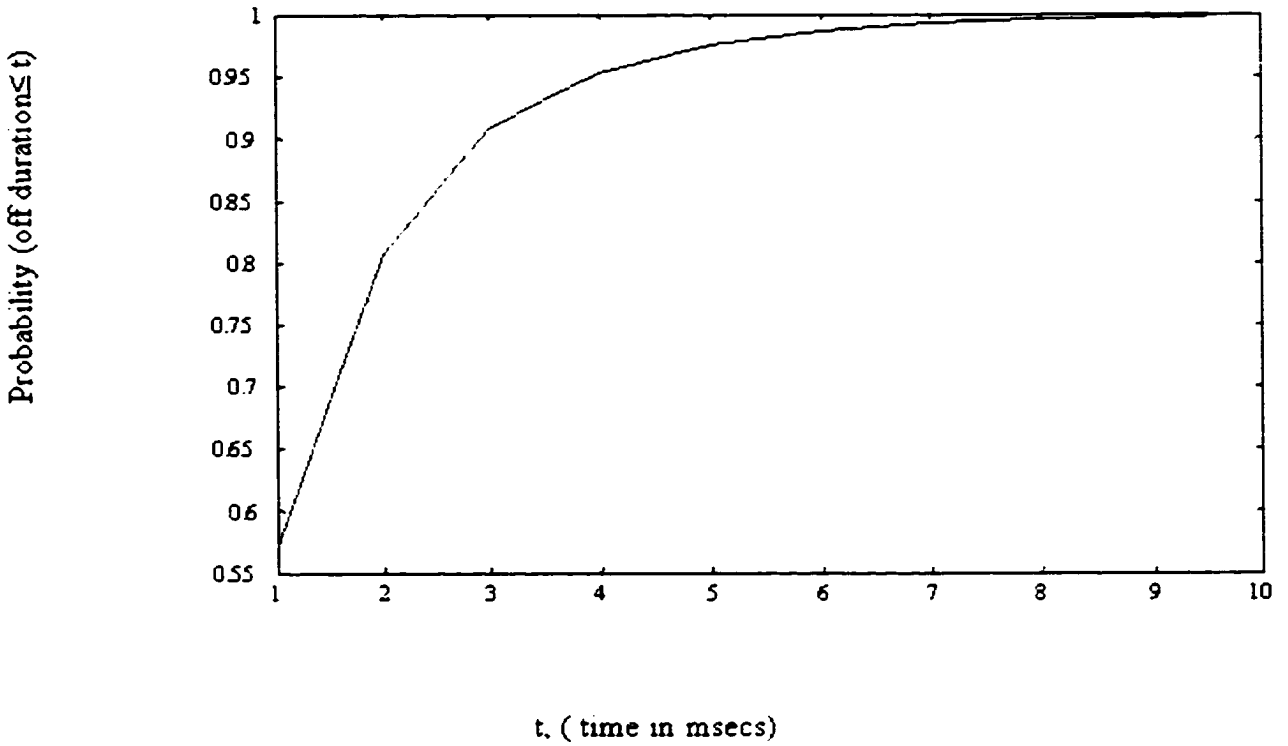


Figure 6.11: comparison of simulation and analysis CDF for off state

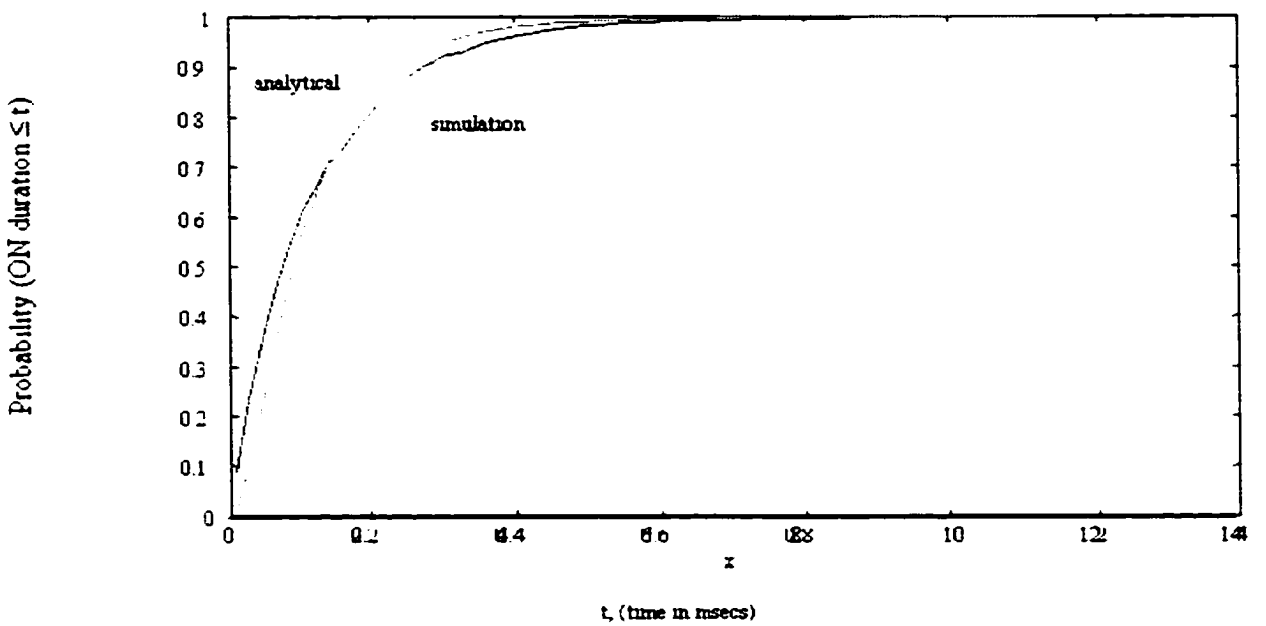


Figure 6.12: comparison of simulation and analysis CDF ON duration

To validate the formula for the cumulative distribution function of two on/off sources, we used two video clips and split them into on/off states using method described earlier. Figure 6.13 compares the analytic and measurement of the 'on' state of the combined source whereas figure 6.14 compares the off state.

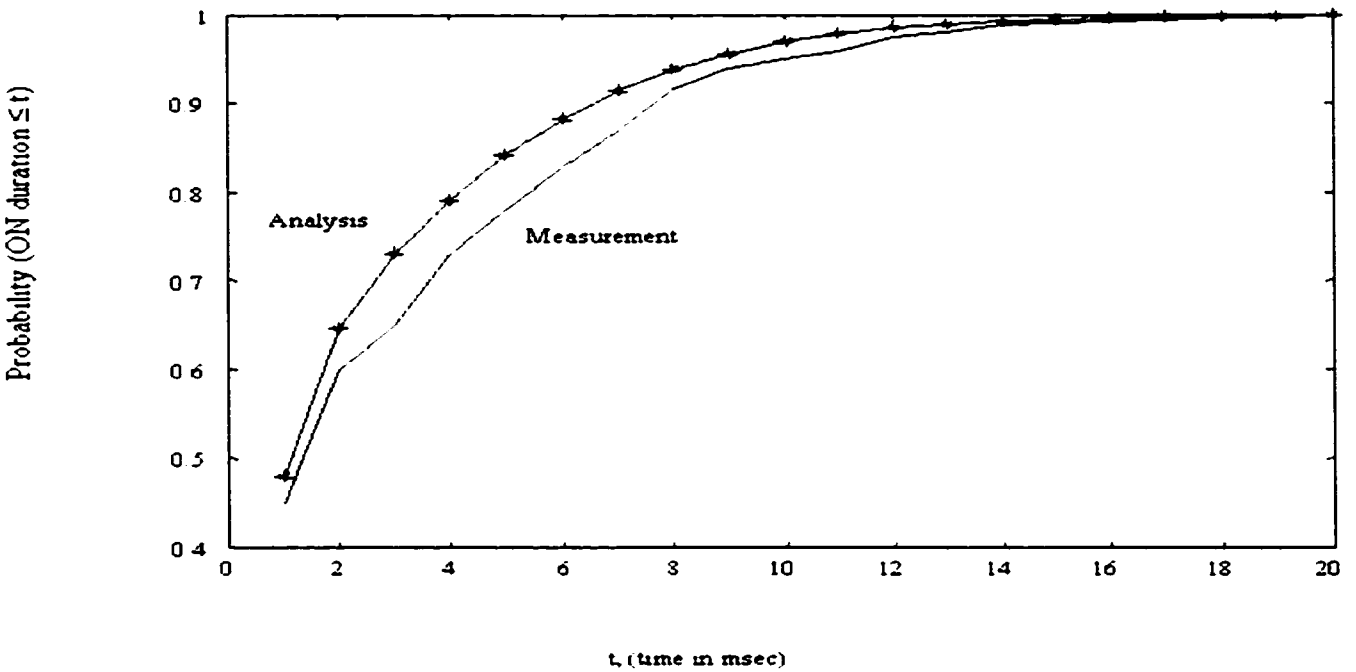


Figure 6.13: comparison of CDF of analytical and measurement for ON duration

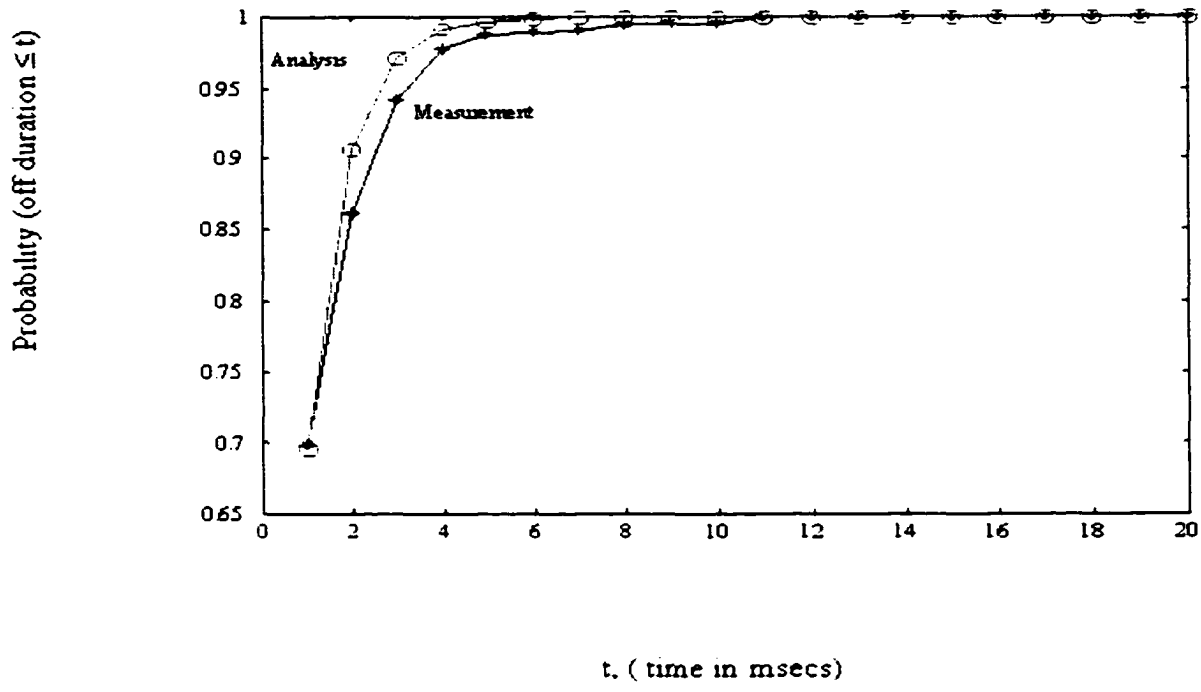


Figure 6.14: comparison of CDF of analytical and measurement for OFF duration

The analytical expression for the cumulative distribution function of two sources compares with actual measurements. In the case of the 'on state', the analytical values is initially below the measured value, then stays, for the most part above the measured value. But the difference in value decreases with time. Our choice of the threshold value accounts for the difference. Similarly, for the 'off state', the analytical values are initially below measurements, but deviates between times 4 and and 8 and then closes up again. It seems that the threshold will have to raised to decrease the rate of the cdf in both on/off states.

6.5 : comparison of analytical and measurements

Recall that we treated the combined source as one source, since the network sees only one source regardless of how many sources access the multiplexer. Below we present the combined sources (zorro and clapton) sequence and compare their actual bandwidths with the effective capacity calculation at various lags equation (6.3)

Table 6.1: Comparison of analytical and actual effective capacities (Bytes/sec) at various times in (sec).

Time (seconds)	0.5	1.0	1.5	2.3
Ceff	**103030**	**101210**	**100290**	**95972**
Actual	103000	101115	100295	96000
Steady state cap	89000	89000	89000	89000

The loss due to effective capacity without factoring in correlation is greater than the effective capacity with correlation factored in. It is observed that at times the effective capacity may be below the steady state capacity.

To test the validity of our effective capacity formula, a comparison of the cell loss was carried out from simulation and analytical point of view. The original trace (zorroty1) served as the input to a multiplexer of buffer size of 20 cells, and an output link capacity of 5Mbps. The loss rate was computed as the ratio of rejected packets, due to buffer overflow, to the total number of arrivals.

Using our effective capacity formula, as a function of correlation (equation 6.1), we computed the bit rate for a given time. The ratio of the difference between our capacity

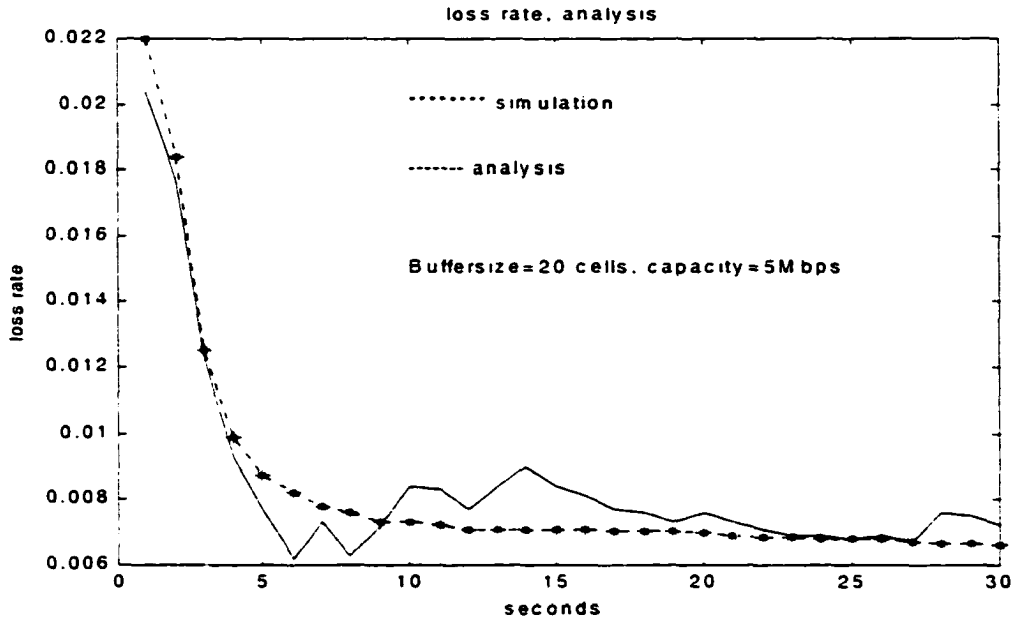


Figure 6.15: comparison of loss rate for analytical and measurement (LR and M).

at that particular time and the steady state capacity, to that of our capacity at that time, was the loss rate. The buffer size was as above (20 cells), and the output link capacity 5 Mbps. Figure 6.15 shows the loss rate for both analytical and measurement (LR and M) methods.

The procedure was repeated for the following pair of buffer sizes and link capacity: (20 cells, 10 Mbps; 10 cells, 5Mbps; 10 cells, 10 Mbps). Figures 6.16, 6.17, and 6.18 depict the corresponding "LR and M" .

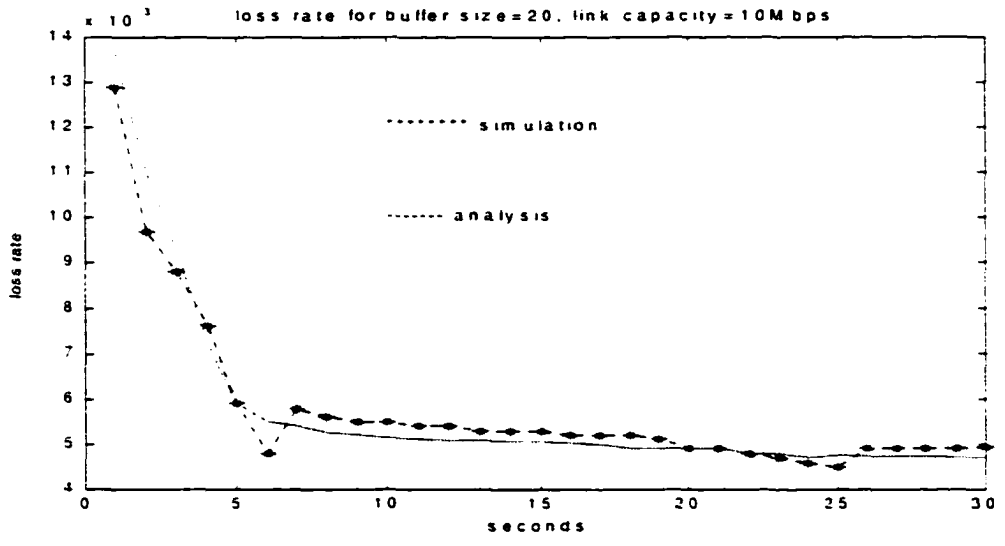


Figure 6.16: LR and M

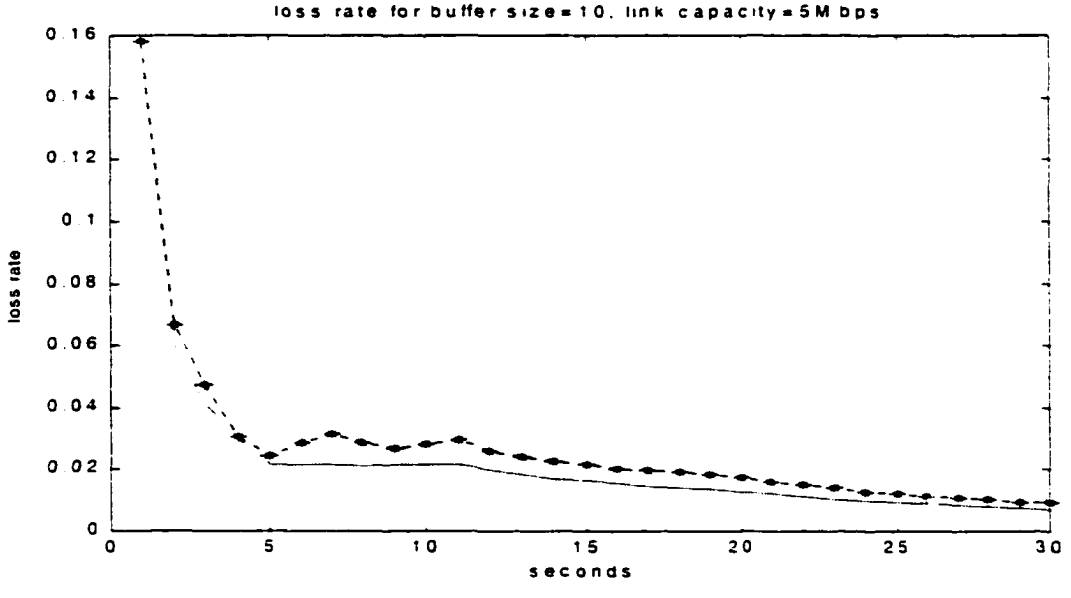


Figure 6.17: LR and M

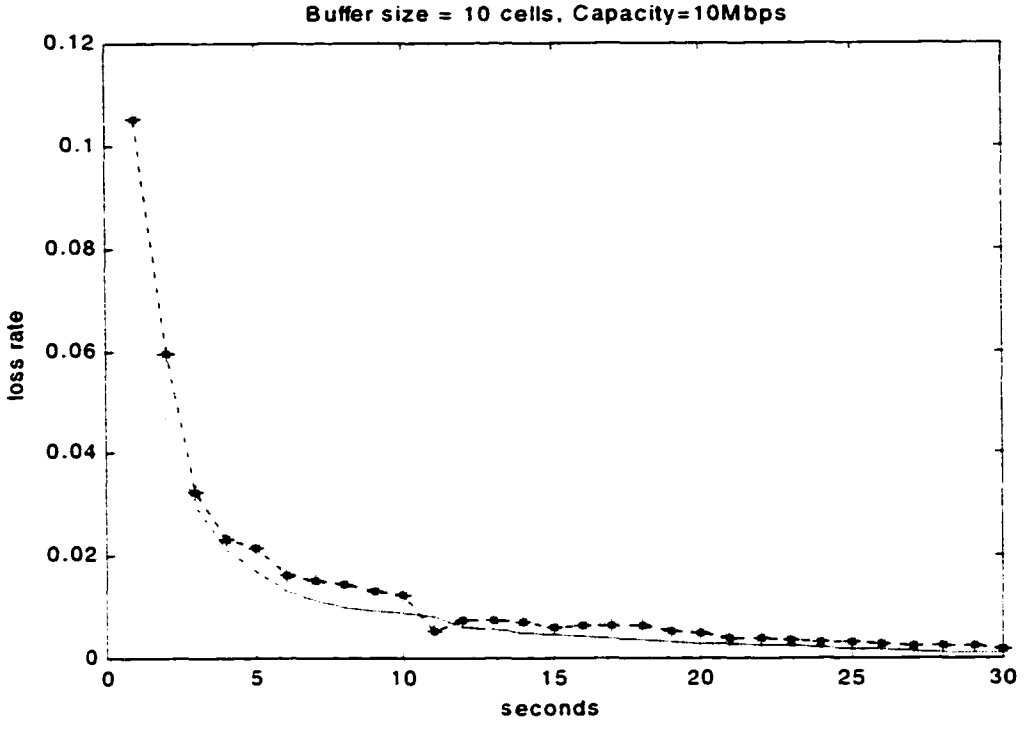


Figure 6.18: LR and M

It is observed that the larger the buffer size the less the loss rate. As the output link capacity increases, more cells are served, thus emptying the buffer. This leads to reduced cell loss, in the case of ATM.

6.6 Application of Model to a queueing Network

In figure 6.19, we show a block diagram of a practical network showing network elements between the encoder up to the traffic admission point of the network. Figure 6.19 shows where our on/off model is applied.

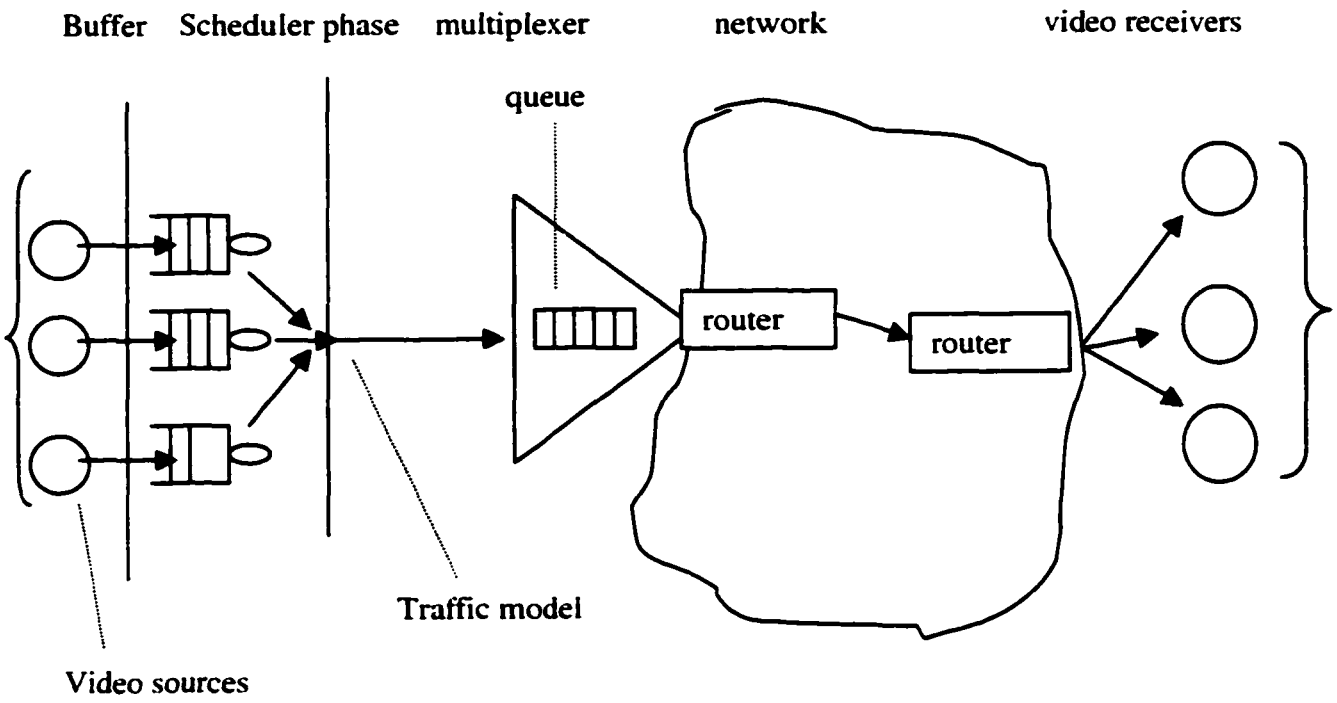


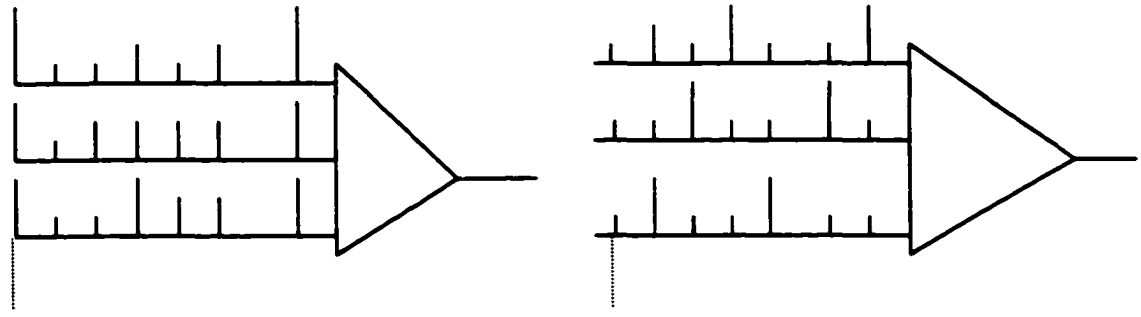
Figure 6.19: Encoder and point of application of our on/off model

The traffic model stated in equation (6.1) takes into consideration the on/off times, as well as the correlation among the traffic sources. In the past, correlation among resources had not been taken into consideration. This new traffic model considers correlation. This offers the optimal use of bandwidth, leading to loss reduction, or over allocation of bandwidth.

In using the model, the variable bit rates form the encoder are buffered and transmitted over a constant link capacity with 'on' times being the transmission times, and 'off' times being the time before the next transmission.

The effect of correlation could also be reduced by randomizing the starting frame of each source. If the starting frames of all sources are I-frames, we say we have the highest alignment. On the other hand, if the starting frames are B-frames, we have the

lowest alignment. Figure 6.20 shows the effect of highest and lowest alignment. The highest alignment leads to a higher loss, and the lowest alignment will lead to a lower loss. Random alignments in between the two extremes leads to loss in between [23].



a) Highest alignment b) Lowest alignment

Figure 6.20: Alignment scenarios

.

6.7: Bandwidth Allocation Ratio versus Autocorrelation

What bearing does knowledge of a traffic correlation have on bandwidth allocation?

From section 6.2.1, it was discussed that as correlation approaches zero, that is at the steady state, effective capacity approaches the steady state capacity. Table 6.2 lists the bandwidth allocation ratio versus correlation. The results are depicted in figure 6.21. Highly correlated traffic will tend to consume more bandwidth, and hence their allocation ratio will be high. Table 6.2 shows some numerical results of autocorrelation and bandwidth allocation ratio.

Table 6.2: Bandwidth Allocation ratio and autocorrelation

corr	.2588	.3070	.3597	.4009	.5474	.6579	.6798	.7149	.8596	1.000
Alloc	.6726	.6862	.7021	.7276	.7707	.7998	.8055	.8103	.8344	.8733

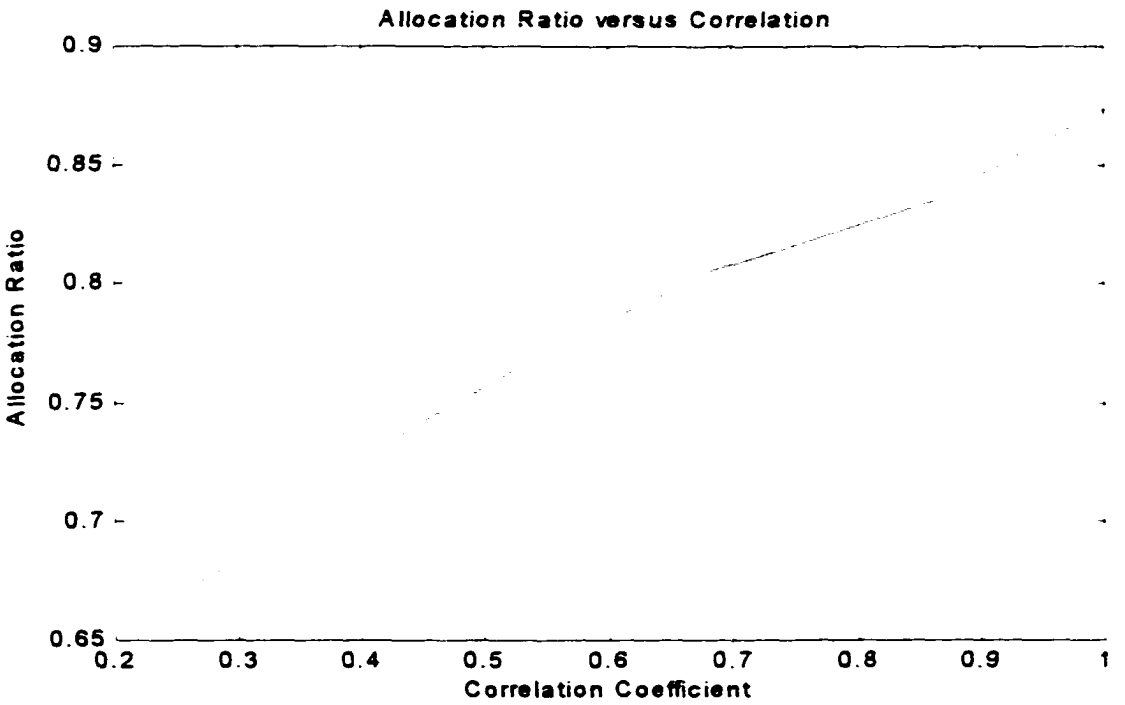


Figure 6.21: bandwidth allocation ratio vs. correlation coefficient

6.8: Effect of short and long range dependency on Bandwidth Allocation

In equation (6.1), the effective bandwidth in the transient duration decayed to a steady state. Recall also that the effective capacity in the short range increased and settled to a steady state, depending upon the 'initial on state probability'. In the long-range the effective capacity at an initial point of observation closely approximates the effective capacity at a future point of observation. This is the steady state capacity (see figure 6.2). We will revisit the short and long range dependency in figure 7.3, sections 7.1.3 and 7.1.4.

6.9: Mathematical characteristics of Bandwidth Allocation

Multimedia data often exhibits a high degree of burstiness in its flow(such as in MPEG video with variability in scene contents and scene changes). The burstiness manifests itself in the form of random fluctuations in data flow rates at various time scales[15, 38]], and plays a role in determining the bandwidth needs.

We define burst ratio η and an allocation ratio γ as follows[31, 39]:

$$\eta = \frac{avg}{P}; \qquad \gamma = \frac{b}{P};$$

where b is the actual bandwidth to be allocated, avg is the average rate and P, the peak rate. For Variable bit rate traffic, avg < P, and avg < b ≤ P. Let us now define burstiness as:

$$\text{Burstiness} = 1 - \frac{avg}{P}.$$

This definition may be misleading, for a traffic could exhibit a continuous flow and may be bursty at only a single instant of time, leading to a high index of burstiness. To overcome this problem, we observe the amount of data within a window of time, t to t + τ. The average, avg, is then computed as:

$$\text{avg} = r_{avg}(t, \tau) = \frac{\int_{t}^{t+\tau} r(t)dt}{\tau}, \text{ where } \tau > 0.$$

This smooths out the instantaneous burstiness. The average of the various peaks is then the new Peak used to calculate burstiness. Hence, instantaneous burstiness are smoothed out.

The question then is how burstiness impacts on bandwidth allocation. From the definition, burstiness increases as the peak rate increases, and decreases as the average increases. As the average approaches the peak, burstiness is zero. From figure 6.23, it can be seen that as the burst ratio increases, bandwidth allocation ratio increases. Also, as burst ratio decreases bandwidth allocation ratio decreases. Low burst ratios lead to high index of burstiness, and the network allocates more bandwidth. This is seen by the arrow above the diagonal. High burst ratios lead to low index of burstiness, and the network tends to under allocate. This is seen by the arrow below the diagonal. Table 6.3 gives some numerical results from the Zorrotyl sequence.

Table 6.3: Allocation ratio and burst ratio from zorroty1 sequence.

γ	.6726	.6862	.7021	.7276	.7707	.7997	.8055	.8104	.8344	.8733
η	.3453	.3724	.4043	.4552	.5415	.5994	.6110	.6207	.6688	.7446

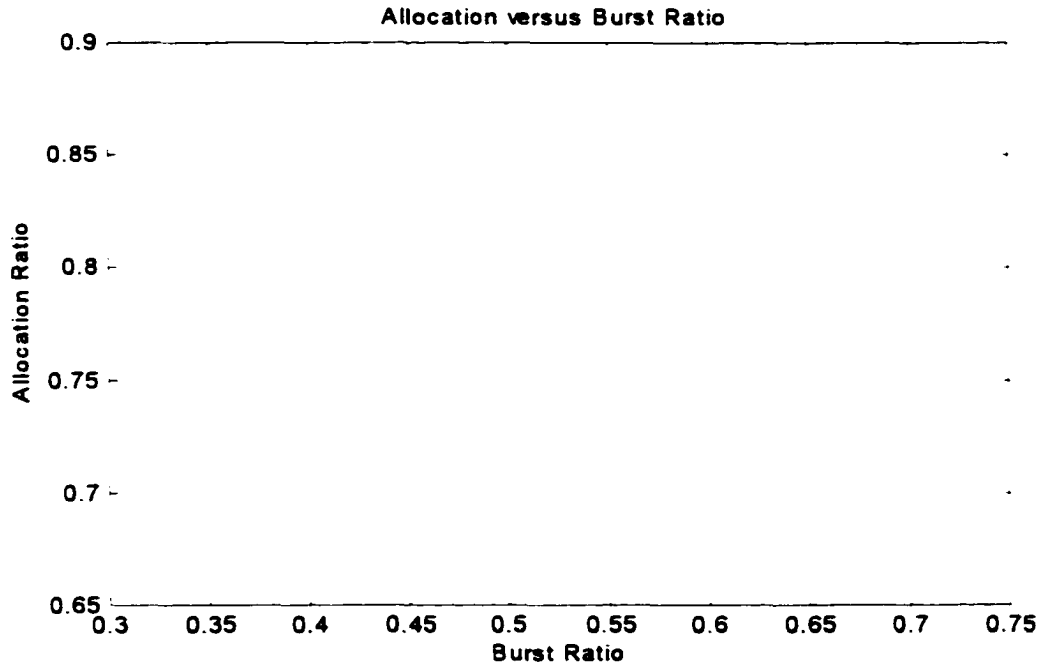


Figure 6.22: BW allocation ratio vs. burst ratio for zorroty1 sequence

Thus, we have shown that mpeg video traffic can be described in terms of its burstiness, correlation, and its long-range dependency. Since it is describable in terms of actual parameters, it can be controlled. Appropriate parameters can therefore be chosen when designing networks that will carry these traffic types.

Summary

We have characterized a relation between effective capacity and correlation, and have shown that as correlation becomes zero, our effective capacity formulation becomes that of the independent steady state effective capacity. We showed that the loss rate in the case of the correlated source was higher than that of independent steady state capacity. Network engineers need to consider this fact, particularly when considering variable bit rate Mpeg-2 video sources with strong correlation. As an extension of this work, we examine in the next chapter cases, where the effective capacity as a function of correlation is below the steady state effective value. This is termed 'effective capacity switchover.'

7. EFFECTIVE CAPACITY ABOVE AND BELOW STEADY STATE (SWITCHOVER)

Effective capacity as a function of correlation in several moments of video clips demonstrates interesting behavior. In several cases of analysis, it is observed that depending on the parameters of the source, effective capacity may be below or above the steady state value in short time interval. We call this "Switchover." In this chapter, we focus on such behavior and try to explain the implications.

Beginning with earlier effective capacity formulation [3-7], and for a two-state on/off source, the effect of correlation was captured analytically. It was observed, that depending upon the transition parameters, the effective capacity formula as a function of correlation, switched between positive and negative slopes. This means that the value of the effective capacity is sometimes above or below the steady state. This chapter investigates what causes the 'switch over', and what impact this has on network performance.

We begin by first developing the effective capacity formula as a function of correlation and time, and show that as time goes to infinity, our effective capacity formulation becomes the steady state formula developed by several authors [3-7]. Similarly, it is shown that as correlation tends to zero, our formula approaches, again, the steady state capacity. Effective capacity above steady state has a negative slope and tends to zero. On the other hand, effective capacity below the steady state has a positive slope, but also tends to zero. By examining the derivative of the

effective capacity expression with respect to time lag, the notion of a 'switch over' is arrived at. The probability of the 'on state' at time zero greater than the 'switch over' implies effective capacity above the steady state capacity. Conversely, the probability of the 'on state' below the 'switch over' implies effective capacity below the steady state. It is further shown that, if the probability of the 'on state' at time zero equals the 'switch over', the two effective capacities are equal, and the slope is zero. Finally, we examine the performance of the effective capacity as a function of correlation formula, both above and below the steady state capacity, in terms of its cell loss rate.

7.1 Attributes of the effective capacity formula

Recall in chapter 6, that for an on/off source switching randomly at the rates α (off to on) and β (on to off) states, we get, for a two-state Markov Chain, the effective capacity:

$$C\_eff = \frac{Z}{\theta} = \left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta} + \frac{(\alpha+\beta)e^{-(\alpha+\beta)t}}{2\theta} - \frac{(\alpha+\beta)[p_1(0)+p_2(0)]e^{-(\alpha+\beta)t}}{2\theta}\right] +$$

$$\sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta} + \frac{(\alpha+\beta)e^{-(\alpha+\beta)t}}{2\theta} - \frac{(\alpha+\beta)[p_1(0)+p_2(0)]e^{-(\alpha+\beta)t}}{2\theta}\right]^2 - \frac{R_p}{\theta}[\{\alpha(1-e^{-(\alpha+\beta)t})\} + p_1(0)e^{-(\alpha+\beta)t}(\alpha+\beta)]}$$

--(6.1)

As $t \rightarrow \infty$, C_eff reaches a steady state capacity defined by earlier authors [3-7]. This becomes, after some algebra,

$$C\_eff = \frac{Z}{\theta} = \left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right] + \sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right]^2 - \left[\frac{\alpha R_p}{\theta}\right]} \qquad (6.2)$$

M is the number of states; p is the steady-state probability of the 'on' state, and A is the average transmission rate. Incorporating the correlation, we get as in chapter 7,

$$C\_eff = \frac{Z}{\theta} = \left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right]$$

$$\sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right] + \frac{R_p}{\theta}\left[\alpha\left[1 - \frac{Rxx(t)}{MA\text{^}2p(1-p)}\right] + p_1(0)(\alpha+\beta)\frac{Rxx(t)}{MA\text{^}2p(1-p)}\right]} \quad (6.3)$$

7.1.1 Examining the slope

Taking the derivative of the effective capacity as a function of correlation, with respect to time lag, t, gives the following result:

$$\frac{[-\alpha R_p + (\alpha+\beta)R_p p_1(0)][-(\alpha+\beta)MA^2 p(1-p)\exp(-(\alpha+\beta)t)]}{2\theta MA^2 p(1-p)\sqrt{\left[\frac{R_p}{2} - \frac{\alpha+\beta}{2\theta}\right]^2 + \frac{R_p}{\theta}\left[\alpha(1 - \frac{Rxx(t)}{MA^2 p(1-p)}) + p_1(0)(\alpha+\beta)\frac{Rxx(t)}{MA^2 p(1-p)}\right]}}$$

> 0, for positive slope (7.1)

Setting the derivative greater than zero for effective capacity with positive slope, it is found, after some algebra, that $p_1(0) < \alpha/(\alpha+\beta)$. Conversely, $p_1(0) > \alpha/(\alpha+\beta)$ is found when the derivative is set less than zero, for effective capacity with negative slope. The ratio, $s = \alpha/\alpha+\beta)$ captures the 'switchover'. This ratio is the probability of the 'on state' at the steady state.

7.1.2: Insight into $p_1(0)$.

For the sake of simplicity, the probability of the 'on' state at time zero ($p_1(0)$, will simply be called P. If P is defined as $\alpha/(\alpha+\beta)$, then

$$\frac{\partial P}{\partial \alpha} = \frac{\beta}{(\alpha+\beta)^2} > 0 \text{, for } \beta \text{ kept constant, and}$$

$$\frac{\partial P}{\partial \beta} = \frac{-\alpha}{(\alpha+\beta)^2} < 0 \text{, for } \alpha \text{ kept a constant} \qquad (7.2)$$

This implies that the change of the probability of the 'on' state with respect to α is positive (β constant), and negative for all β (α constant). Figure 7.1 below, is a plot of P versus α, with β constant and figure 7.2, a plot of P versus β. If P* is defined as: $P^* = \frac{\alpha^*}{(\alpha^*+\beta^*)}$, the steady state probability, then P' greater than P* can be defined as:

$$P' = \frac{\alpha'}{(\alpha'+\beta')} > \frac{\alpha^*}{(\alpha^*+\beta^*)} \text{----------(7.3)}$$

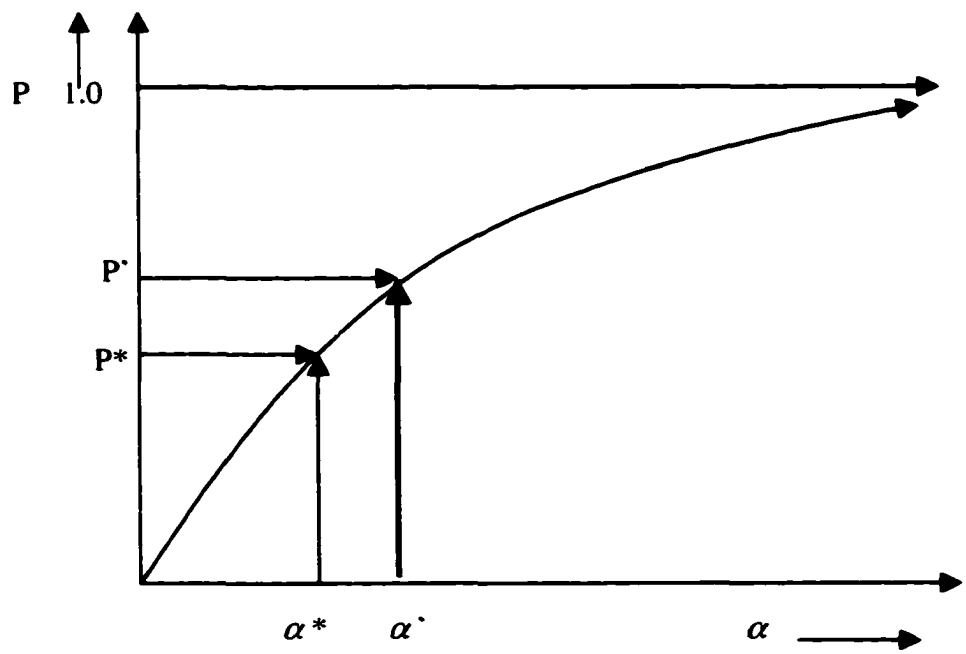


Figure 7.1: Probability of the "on" state at time zero vs alpha

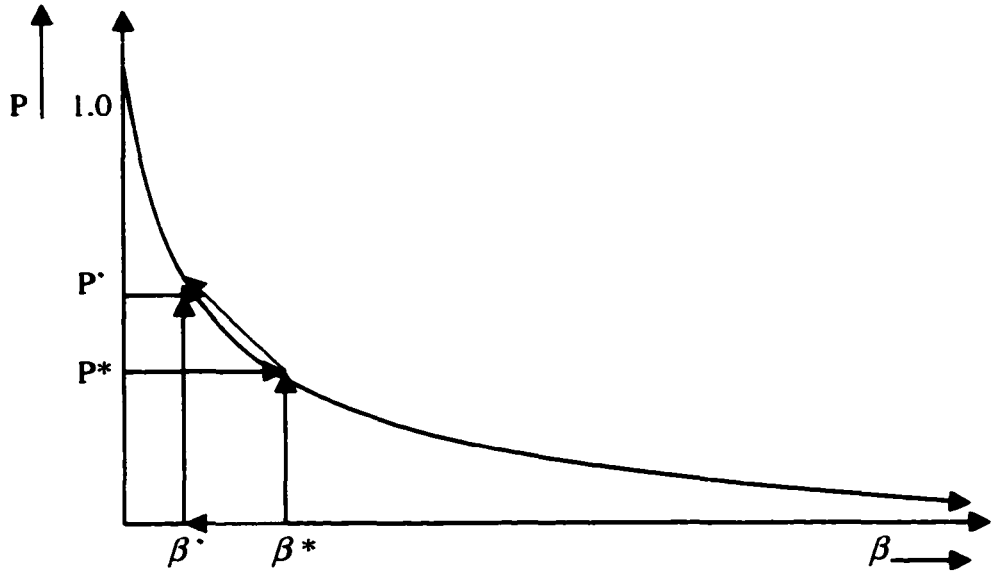


Figure 7.2: Probability of the "on" state at time zero vs. beta

From figure 7.1, P' corresponds to α', which is greater than α^*. From figure 7.2., β' has to be less than β^* for (7.3) to hold. $1/\beta'$ then has a longer duration during the 'on' state and leads to effective capacity above steady state. A similar analysis holds for the reverse process and leads to effective capacity below the steady state.

Intuitively, when $p_1(0)$ (the probability of the 'on state' at time zero), is chosen less than the 'switch over', the effective capacity will be less than the steady state. This means that the bit rate injected into the network initially will be less than at the steady state. Conversely, when $p_1(0)$ is chosen above the 'switch over', higher bit rates will be injected into the network initially. Below, we present $p_1(0)$ less than and above $\alpha/\alpha+\beta)$ to denote effective capacity below and above the steady state capacity, respectively.

7.1.3 (p1(0)< S)

The parameter α is chosen as 0.3 and β as 0.4. Then S = .3/(.3+.4)=.429. For effective capacity below the steady state. p1(0) is chosen less than S. As an example, p1(0) is chosen as 0.9 * S. Here, we see in (figure 7.3), the effective capacity below the steady state capacity and eventually reaches the steady state.

7.1.4 (p1(0) >S).

Here again, α = .3 and β = .4 and $p_1(0)$ is chosen 1.2 times the 'switchover'. The effective capacity is above the steady state capacity, (see figure 7.3).

Figure 7.3 further shows that when the 'switch over' point is exactly equal to the 'on state', the effective capacity is equal to the steady state capacity

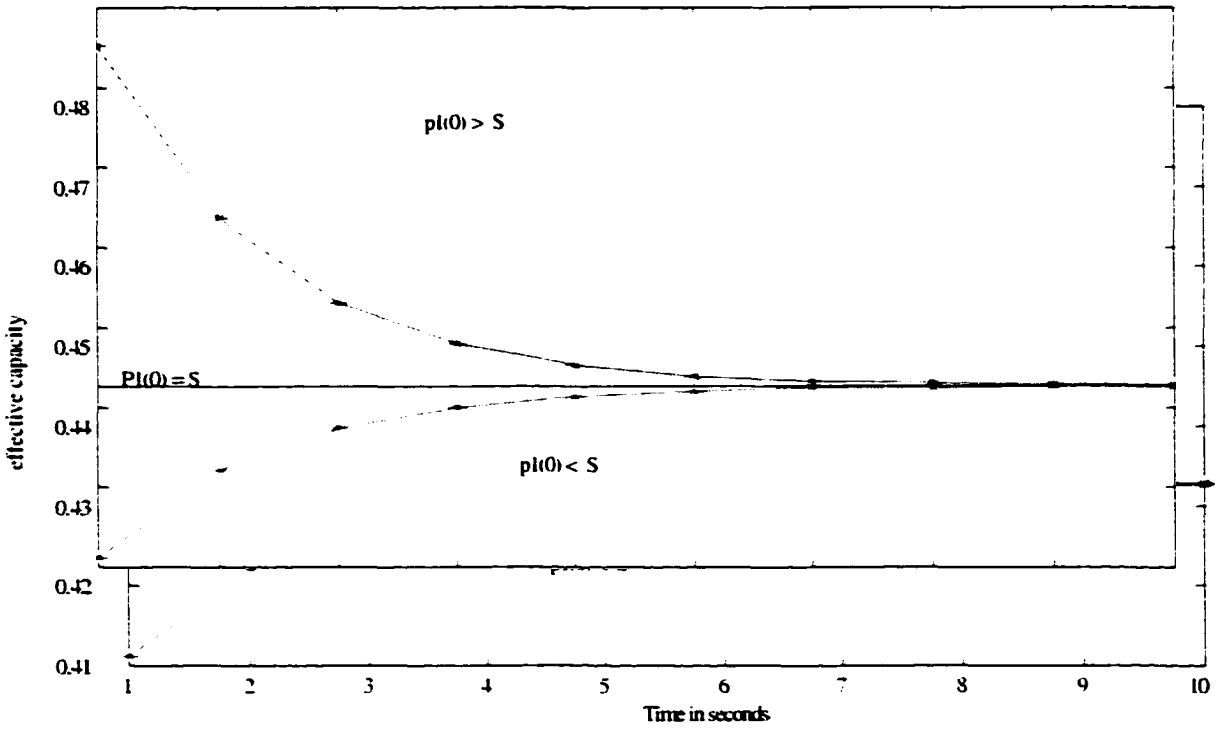


Figure 7.3: effective capacity above and below steady state

7.2 Steady state behavior of effective capacity

Case 1 : effective capacity above steady state (Negative slopes)

The slope of the effective capacity above the steady state capacity is negative. This means that, although higher bit rates are initially injected into the network, there is a gradual decrease (or rate of change) in the bandwidth injected until it reaches the steady state capacity.

Case 2: effective capacity below steady state (positive slopes)

The slope of the effective capacity below the steady state capacity is positive. This means that initially there is an increase in the bit rate, increasing it by a lesser amount until the steady state is reached. In other words, the rate of increasing the bit rate decreases till it reaches zero. Figure 7.4 demonstrates this behavior for the two cases discussed above.

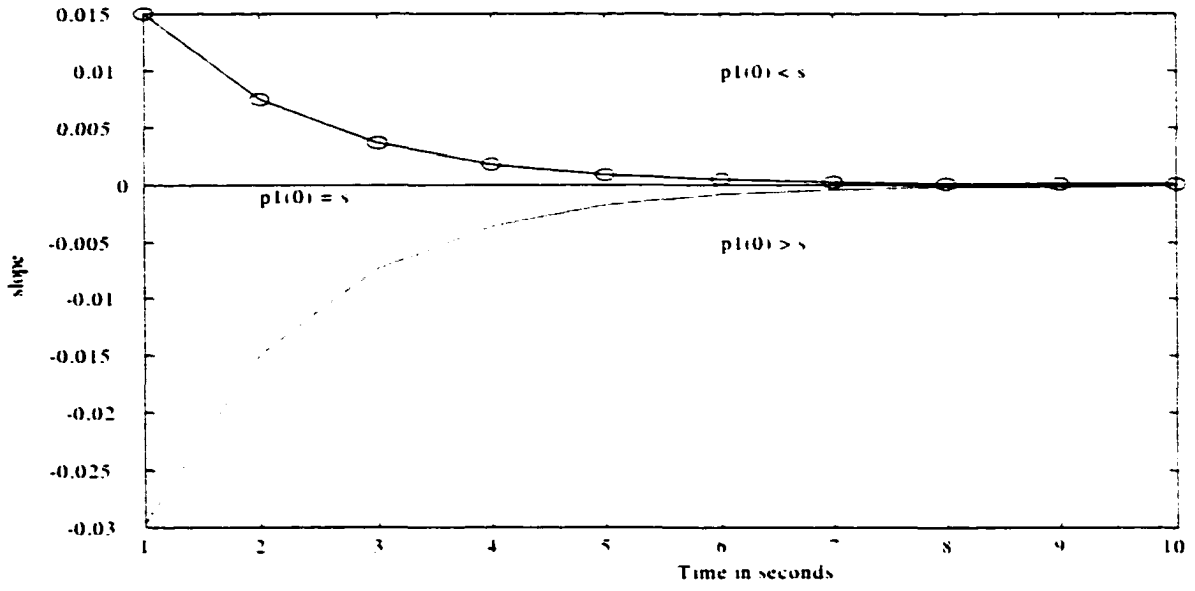


Figure 7.4: positive slope for p1(0)<S and negative slope for p1(0) < S

7.3 Performance Analysis of ATM networks

The simulation tool, Queuing Plus (Q+), used for our setup was developed by Bell Labs in their Performance Analysis Department [36]. We set up a simple queueing network with the link capacity from the source to the buffer as 155Mbps to ensure an 'on' state without spacing. The output link from the buffer to the network was 5 Mbps. The size of the buffer was 530 bytes (4240 bits). The loss rate was defined as the ratio of the number of losses as a result of buffer overflow to the total number of arrivals.

The analytical loss was defined as the difference between the bandwidth and the steady state bandwidth. This was done different times.

Below, we present a comparison of loss rates between simulation and analysis. Figure 7.5 shows that the analytical model closely matches that of simulation in terms of the loss rate. The graph further shows that the loss rate in the transient state may be substantial. Network designers must incorporate this fact in the allocation of bandwidth so as to meet the quality of service.

In the short time interval, the injection of bit rates into the network may be at a rate higher than the allocated steady state capacity, and may lead to losses. On the other hand, the injection of bit rates into the network may be less than the steady state allocation, leading to a wastage of bandwidth (over allocation).

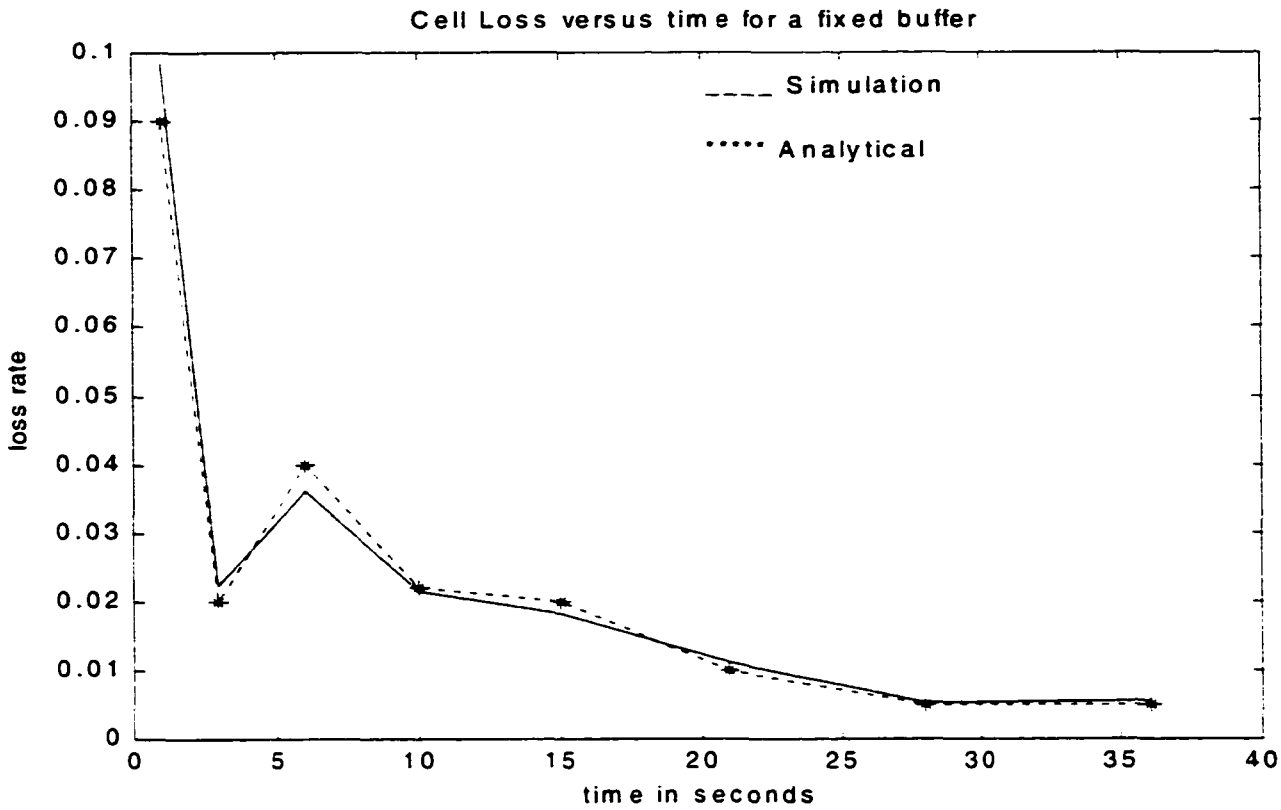


Figure 7.5: loss rate for effective capacity above steady state

Summary

By means of analysis, it has been shown that depending upon the choice of p1(0) in terms of $\alpha/(\alpha+\beta)$, the probability of the 'on state' in the steady state, the value of the effective capacity could be above or below the steady state capacity. It was shown that the slope of the effective capacity above the steady state was negative and tended to zero as time approached infinity. Conversely, the slope of the effective capacity below the steady state was positive and asymptotically approached zero. When the capacity is equal to the steady state capacity, the slope is zero. Finally, we examined the

performance of our effective capacity formula in terms of its cell loss probability. In the transient time interval, cell loss can be substantial, and may lead to degradation of picture quality. Network designers must consider this fact of cell loss for correlated sources when designing networks so as to meet the quality of service demands.

In practice, the number of sources could be several. Hence, we extend our treatment of chapters 6 and 7 to include several correlated sources. Chapter 8 will discuss this.

8. EFFECTIVE CAPACITY FOR MULTIPLE SOURCES

A two-state on/off source model can be used to calculate the effective capacity of several heterogeneous, correlated sources. The network access point can see only one stream despite the fact that several sources have combined to form this stream. Since the resultant stream is a single stream, we can model it as an on/off source, but with an α_{eq} (off to on) and a β_{eq} (on to off). The diagram below depicts this model:

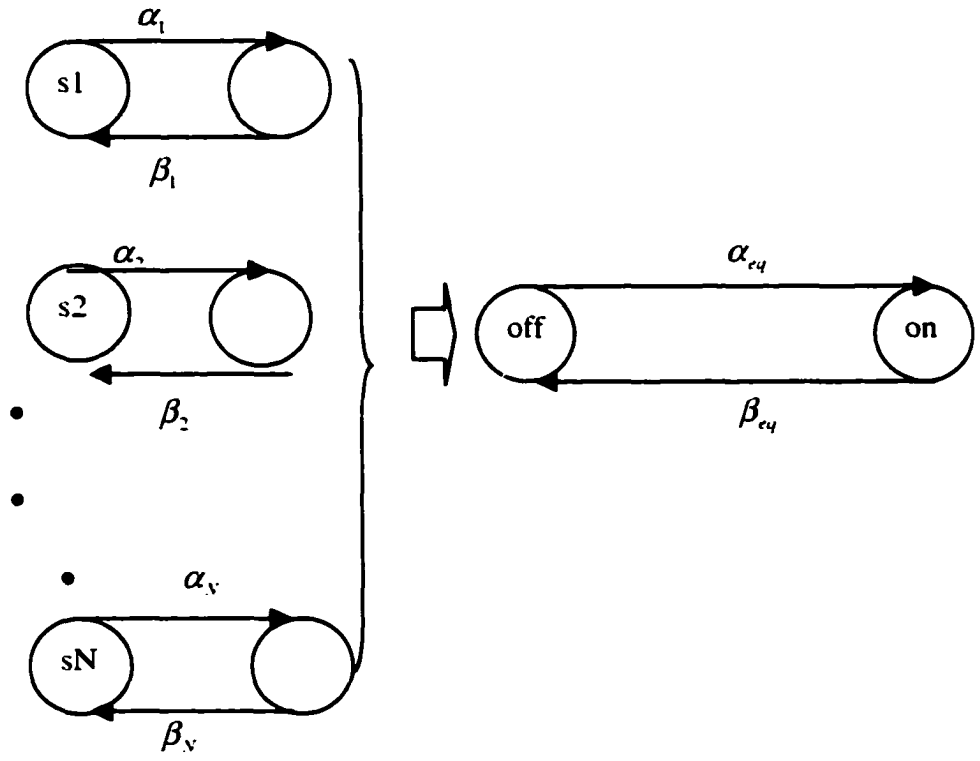


Figure 8.1: Several correlated sources seen as a singe source

$\alpha_{eq} = f(\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \alpha_N, \beta_N)$ and

$\beta_{eq} = g(\alpha_1, \beta_1, \alpha_2, \beta_2, \ldots, \alpha_N, \beta_N)$. (8.1)

Our analysis of a single source gets extended to the multiple source by combining all sources as a single on/off source with equivalent transition durations from off to on and vice versa, expressed as a function of source parameters (see 8.1). All analysis from chapters 6 and 7 carry over, and intra-frame correlation is replaced by an overall auto-and-cross correlations.

Earlier authors [3-7] have assumed the sources to be independent, not correlated. In this section, the effect of overall correlation has been incorporated. In the short time interval, the model introduces the 'initial on state probability' (IOSP) of the equivalent or resultant stream. Depending upon the value of the equivalent 'IOSP', the value of the effective capacity may be above or below the steady state capacity. A similar explanation for 'IOSP' holds as in the case of a single source. Additionally, the impact of correlation and 'IOSP' on network performance, such as cell loss in an ATM network, is investigated.

8.1 Equivalent formula.

An equivalent on/off source of several sources, has equivalent transition probabilities $p_{n1}(t)$ and $p_{n2}(t)$ has a transition matrix, Q_{eq}

$$Q_{eq} = \begin{bmatrix} -p_{n1}(t) & p_{n1}(t) \\ p_{n2}(t) & -p_{n2}(t) \end{bmatrix}$$

$p_{n1}(t)$ is the probability of being in the 'on state' at time t, and $p_{n2}(t)$ is the probability of being in the 'off state' at time t. The transition parameters β_{eq} and α_{eq} are exponentially distributed, and denote the rate from 'on' to 'off' state and 'off' to 'on'

respectively in the equivalent model. Table 8.1 shows the equivalent on/off times for two sources

Table 8.1: Equivalent alpha and beta transition rates for the equivalent model.

	Alpha	Beta	N (# of samples)
Source1	2.00	1.00	3400
Source2	1.90	1.10	3400
S3=S1+S2	1.55	1.30	3400

In [37], it has been shown that:

$$p_{n1}(t) = \frac{\alpha_{eq}}{\alpha_{eq}+\beta_{eq}}[1-e^{-(\alpha_{eq}+\beta_{eq})t}] + p_{n1}(0)e^{-(\alpha_{eq}+\beta_{eq})t}$$ and

$$p_{n2}(t) = \frac{\beta_{eq}}{\alpha_{eq}+\beta_{eq}}[1-e^{-(\alpha_{eq}+\beta_{eq})t}] + p_{n2}(0)e^{-(\alpha_{eq}+\beta_{eq})t}$$

$p_{n1}(0)$ is the initial 'on' state probability that was discussed earlier.

In addition, the source steady state probabilities $\pi = [\pi_{n1}, \pi_{n2}]$ can be calculated from the relation $\begin{cases} \pi Q_{eq} = 0 \\ \sum_i \pi_{ni} = 1 \end{cases}$

$Q^* = Q_{eq} * (\alpha_{eq} + \beta_{eq}) + \theta \Lambda$, where Λ = diagonal of rates λ_1 and λ_2. In this case $\lambda_1 = 0$ and $\lambda_2 = R_{p\,n}$, the peak rate.

To calculate the effective capacity C_eff, the maximum eigenvalue, Z, is divided by θ, where θ =Log (1/PL)/x; PL is the probability of loss and x is the variable denoting the buffer size.

To find the eigenvalues, we set the determinant of $(ZI - Q') = 0$.

After some algebraic manipulations, we get, for a two-state Markov Chain the effective capacity:

$$C\_effN = \frac{Z}{\theta} =$$

$$\left[\frac{R_{pn}}{2} - \frac{\alpha_{eq}+\beta_{eq}}{2\theta} + \frac{(\alpha_{eq}+\beta_{eq})e^{-(\alpha_{eq}+\beta_{eq})T}}{2\theta} - \frac{(\alpha_{eq}+\beta_{eq})[p_{n1}(0)+p_{n2}(0)]e^{-(\alpha_{eq}+\beta_{eq})T}}{2\theta}\right] +$$

$$\sqrt{\left(\frac{R_{pn}}{2} - \frac{\alpha_{eq}+\beta_{eq}}{2\theta}\right)^2 + \frac{R_{pn}}{\theta}(\alpha_{eq}(1-e^{-(\alpha_{eq}+\beta_{eq})T}) + p_{n1}(0)e^{-(\alpha_{eq}+\beta_{eq})T}(\alpha_{eq}+\beta_{eq}))} \quad (8.2)$$

$p_{n1}(0) + p_{n2}(0) = 1$

As $t \to \infty$, the effective capacity reaches a steady state capacity. This becomes, after some algebra,

$$C_{eff\_N} = \sum_{i=1}^{N}\left[\frac{R'_p}{2} - \frac{(\alpha_i+\beta_i)}{2\theta}\right] + \sqrt{\left[\frac{R'_p}{2} - \frac{(\alpha_i+\beta_i)}{2\theta}\right]^2 + \frac{\alpha_i R'_p}{\theta}} \quad (8.3)$$

This is the sum of the effective capacities of the individual independent sources.

Observe that equation (8.3) is time independent and depends on the steady state transition rates α_i and β_i, and the peak rate, R_p' and θ.

8.2 Factoring in correlation effect

The correlation derivation shown in chapter eight can be extended to multiple sources. Consider two processes x (t) and y(t) which are jointly wide-sense stationary

(WSS – meaning their means are constant and autocorrelation depends only $\tau = t_1 - t_2$, and $E\{x(t+\tau)x^*(t)\}=R_{XX}(\tau)$, and $E\{y(t+\tau)y^*(t)\}=R_{YY}(\tau)$). Then the process z(t) = ax(t) + b y(t) , will have autocorrelation $R_{ZZ}(t) = |a|^2 R_{XX}(t) + ab^* R_{XY}(t) + a^* b R_{YX}(t) + |b|^2 R_{YY}(t)$.

Notice that this is equivalent to the autocorrelation of the individual sources plus their cross-correlations. For three processes, x(t), y(t) and z(t), the autocorrelation of the sum of such processes will be $R_{XX}(\tau) + R_{XY}(\tau) + R_{XZ}(\tau) + R_{YY}(\tau) + R_{YX}(\tau) + R_{YZ}(\tau) + R_{ZZ}(\tau) + R_{ZX}(\tau) + R_{ZY}(\tau)$. For N such sources, we can form a NxN matrix of auto and cross correlations.

$$R_{WW}(\tau) = \begin{bmatrix} R_{11} & R_{12} & \dots & R_{1N} \\ R_{21} & R_{22} & & R_{2N} \\ & & & \\ R_{N1} & R_{N2} & \dots & R_{NN} \end{bmatrix} \text{----------(8.4)}$$

Here, R_{ii} is the autocorrelation of source i, i =1,2,....,N, and R_{ij} is the cross-correlation of sources i and j, with $i \neq j$.

The effective capacity formulation, as a function of correlation, can be extended to include N such on/off sources. In that case the modified formula would be the same as (6.3) by replacing

α with α_{eq}, β with β_{eq}, A with A_n, P with P_n, Rxx with R_{ww}, $p_1(0)$ with $p_{n1}(0)$, R_{p} with R_{pn}.

The physical meaning of this substitution is that multiple correlated sources could behave like a single source. When the sources are not correlated, the formula boils

down to the sum of N independent heterogeneous sources shown below:

$$Ceff\_N = \sum_{i=1}^{N}\left[\frac{R_p'}{2} - \frac{(\alpha_i + \beta_i)}{2\theta}\right] + \sqrt{\left[\frac{R_p'}{2} - \frac{(\alpha_i + \beta_i)}{2\theta}\right]^2 + \frac{\alpha_i R'_p}{\theta}}$$

Below is a graph of three sources treated as correlated and independent sources.

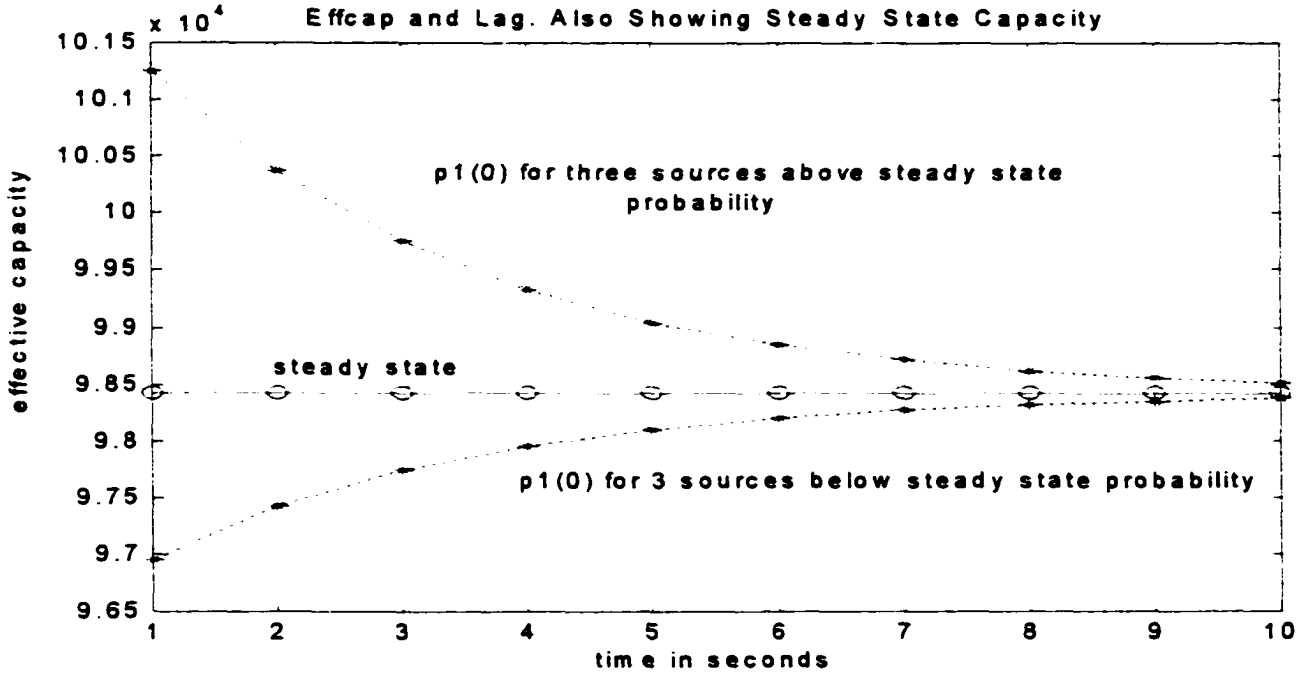


Figure 8.2: Three sources treated as independent and correlated

As Rww(t) approaches zero, with t tending to infinity, equation (8.2) becomes the steady state capacity. As was mentioned in the introductory statements, a network will carry a variety of traffic streams multiplexed together. The formula above captures a relation between correlation and bandwidth. Appropriate relation between bandwidth and correlation will aid network providers to allocate the necessary resources.

8.3 Transient behavior of effective capacity

Taking the derivative of the effective capacity as a function of correlation, with respect to time lag, t, gives the result as in a single source (7.1) by replacing α with α_{eq}, β with β_{eq}, A with A_n, P with P_n, Rxx with R_{ww}, $p_1(0)$ with $p_{n1}(0)$, R_p with R_{pn}.

The analysis in section 7.1.2 through 7.2 hold for multiple sources. Following the analysis in the sections above, the slope for the correlated sources can be derived for above and below steady state. Figure 8.3 presents our results.

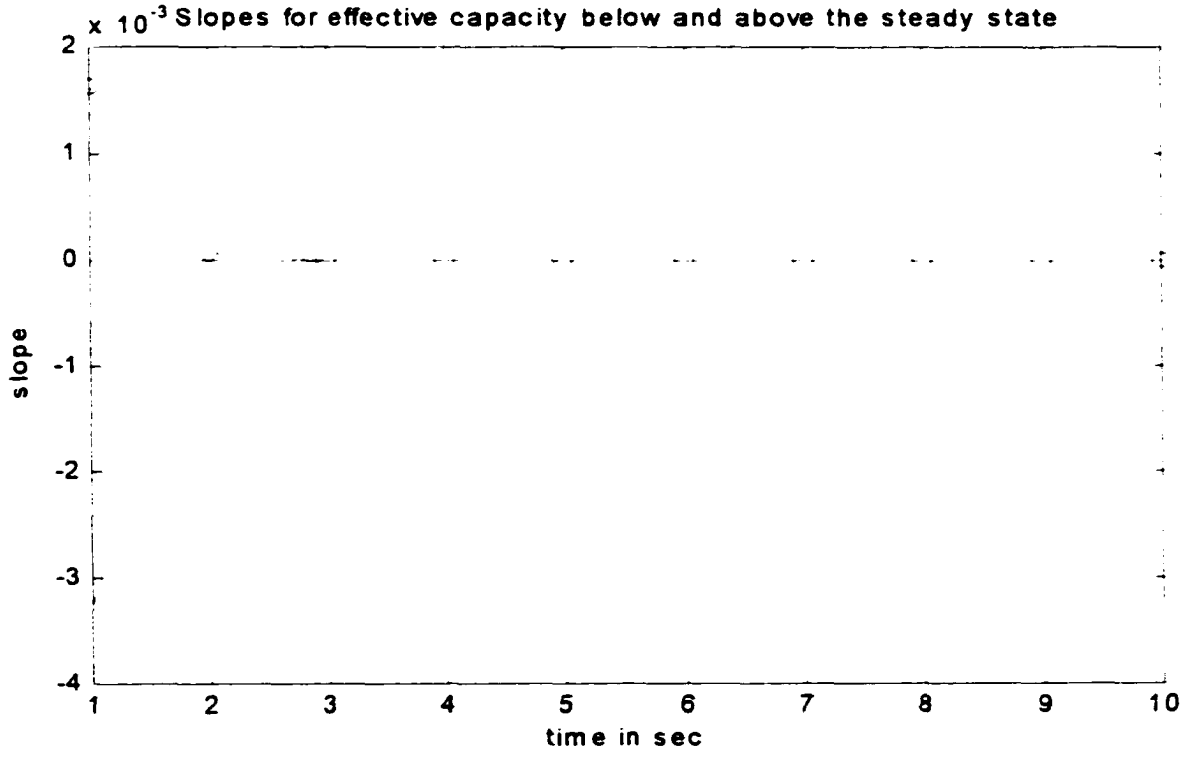


Figure 8.3: Positive slop for pnl(0) < S and negative slope for pnl(0) > S

8.4 Transition rate parameters

The effective capacity formula assumes that the source is a two-state on/off, and that the transition times from 'off ' to 'on' and vice versa, are exponentially distributed. The Mpeg-2 frame rate is 30 fps. This means that there is 33ms or 1/30 seconds in between two consecutive frames. To find the distribution of the on/off states, two random variables are defined:

x ≡ time taken for source to emit frame ('on' period)

y ≡ .033 - x, off period time before next frame

Consider several on/off sources with 'on' and 'off' times exponentially distributed. We would like to determine the 'on' and 'off' distribution of the combined source. This then allows us to calculate the effective capacity of several sources accessing a common buffer. To this end, we generalize the cumulative distribution of two on/off sources to include several sources, and apply the formula to several video clips.

Let $x_1 = \begin{cases} 0 \\ 1 \end{cases}$, $x_2 = \begin{cases} 0 \\ 1 \end{cases}$, ..., $x_n = \begin{cases} 0 \\ 1 \end{cases}$ be several random numbers. The rate diagrams of these sources constructed from the random numbers generated from the definition above, are shown for three sources.

For each source, 8,000 random variates which are exponentially distributed with mean 'off' times equal to .2 .2 .3 for source 1 and 2 respectively, and mean 'on' times were .1, .1 and .2 repectively.

We want to find the probability distribution of both sources being in the 'off state'. We can represent this mathematically as Pr (first source is off and second source is off and

nth source is off}. Since these are independent events, this is equal to Pr{first source is off }*Pr{second source is off.}*pr(nth source is off). The probability distribution of 'off' in the combined state can be written as Prob(off$\leq$x). Substituting the expressions, we have: Prob(off$\leq$x)

$= Pr\{x_1 \leq x \text{ and } x_2 \geq x \text{ and } x_3 \geq x\} + Pr\{x_1 \geq x \text{ and } x_2 \leq x \text{ and } x_3 \geq x\} +$

$Pr(x_1 \geq x \text{ and } x_2 \geq x \text{ and } x_3 \leq x\} + Pr\{x_1 \leq x \text{ and } x_2 \leq x \text{ and } x_3 \geq x\} +$

$Pr\{x_1 \leq x \text{ and } x_2 \geq x \text{ and } x_3 \leq x\} + Pr\{x_1 \leq x \text{ and } x_2 \geq x \text{ and } x_3 \leq x\} +$

$Pr\{x_1 \leq x \text{ and } x_2 \leq x \text{ and } x_3 \leq x\}.$ (8.5)

For n sources, there will be $2^n - 1$ terms.

The number of 'offs' in each term is given by

$\sum({}^nC_1 + {}^nC_2 + \ldots + {}^nC_n)$. Here, n = the number of sources.

Similarly the Prob(On$\leq$x) for 3 sources will be:

$= Pr\{x_1 \text{ is on}\}*Pr\{x_2 \text{ is on}\}*Pr\{x_3 \text{ is on}\} + Pr\{x_1 \text{ is on}\}*Pr\{x_2 \text{ is on}\}*Pr\{x_3 \text{ is off}\} + Pr\{x_1 \text{ is on}\}*Pr\{x_2 \text{ is off}\}*Pr\{x_3 \text{ is on}\} + Pr\{x_1 \text{ is on}\}*Pr\{x_2 \text{ is off}\}*Pr\{x_3 \text{ is off}\} + Pr\{x_1 \text{ is off}\}*Pr\{x_2 \text{ is on}\}*Pr\{x_3 \text{ is on}\} + Pr\{x_1 \text{ is off}\}*Pr\{x_2 \text{ is on}\}*Pr\{x_3 \text{ is off}\} + Pr\{x_1 \text{ is off}\}*Pr\{x_2 \text{ is off}\}*Pr\{x_3 \text{ is on}\}.$

The number of terms for n sources is $2^n - 1$.

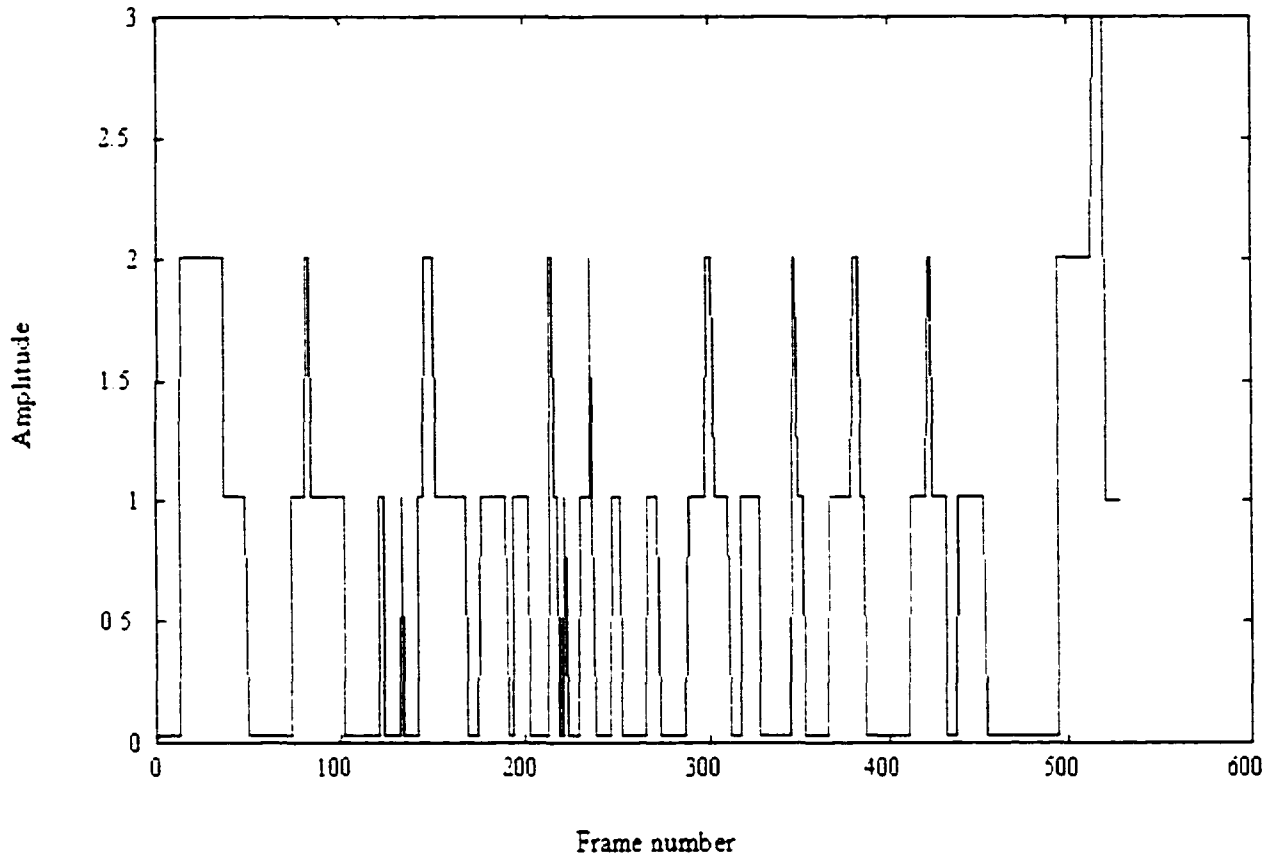


Figure 8.4: Combined effect of 3 sources

The average off periods for sources 1, 2, and 3 are denoted by $1/\lambda_{off1}$, $1/\lambda_{off2}$ and $1/\lambda_{off3}$, respectively. Similarly, the average on periods for sources 1, 2 and 3 are denoted by $1/\lambda_{on1}$, $1/\lambda_{on2}$ and $1/\lambda_{on3}$, respectively.

Substituting the values for the 'off' parameters, we get the probability of 'off' distribution in the combined state shown in expression (8.5) above. Figure 8.5 depicts the cumulative probability distribution of the ' off state' in the combined source in the case of two sources.

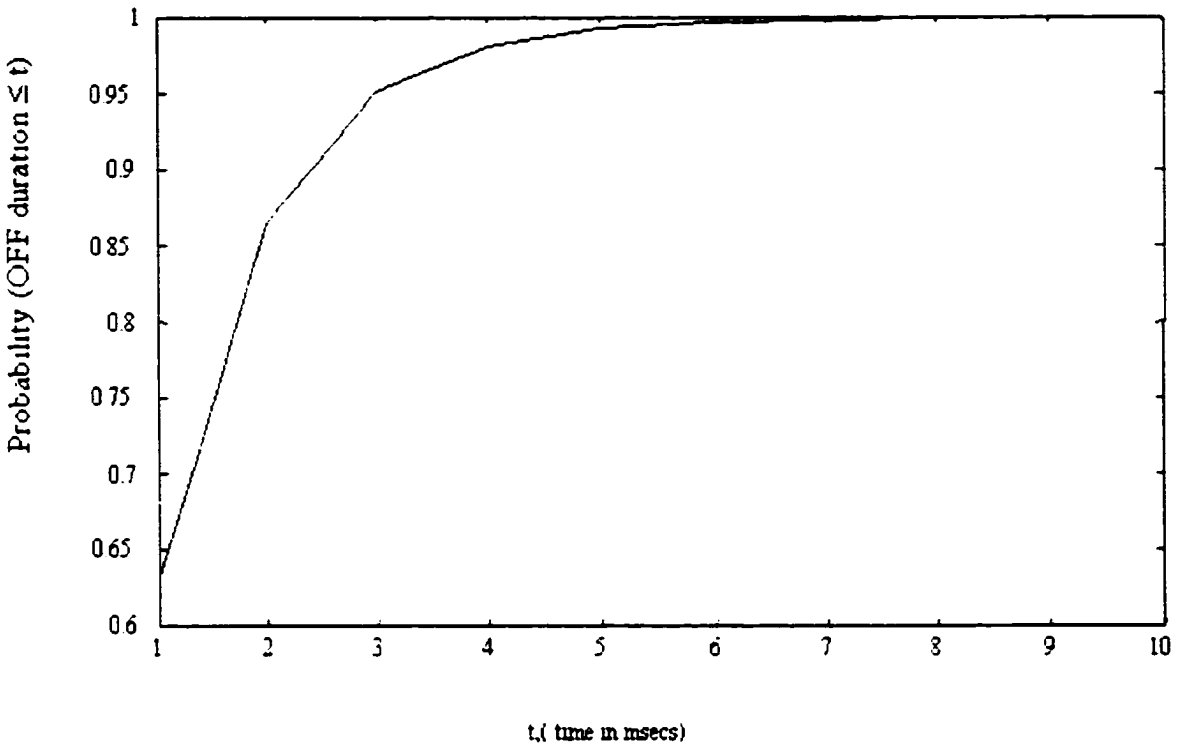


Figure 8.5: Probability distribution of OFF duration

Having found the probability distribution of the 'off state', we now derive a similar expression for the 'on state' for the combined source. Still, using the definition of the random variables, the probability of an 'on state' occurs when one of several cases occur: when both sources are on; when source one is on and source two is off, or when source one is off and source 2 is on. Writing this probabilistically, we get:

$$\text{prob (on} \le x) = \Pr\{x_1=1\}\Pr\{x_2=1\} + \Pr\{x_1=1\}\Pr\{\text{source 2 is off}\} + \Pr\{\text{source 1 is on}\}\Pr\{x_2=1\} = \{1-\exp(-x/\lambda_{on1})\}\{1-\exp(-x/\lambda_{on2})\} + \{1-\exp(-x/\lambda_{on1})\}\{\exp(-x/\lambda_{off2})\} * \{\exp(-x/\lambda_{off1})\}\{1-\exp(-x/\lambda_{on2})\} \quad (8.6)$$

Substituting the values for the 'on state' and 'off state' for sources 1 and 2, we get the expression in equation (8.6). Figure 8.6 shows that the cumulative probability distribution of the 'on state' is exponentially distributed.

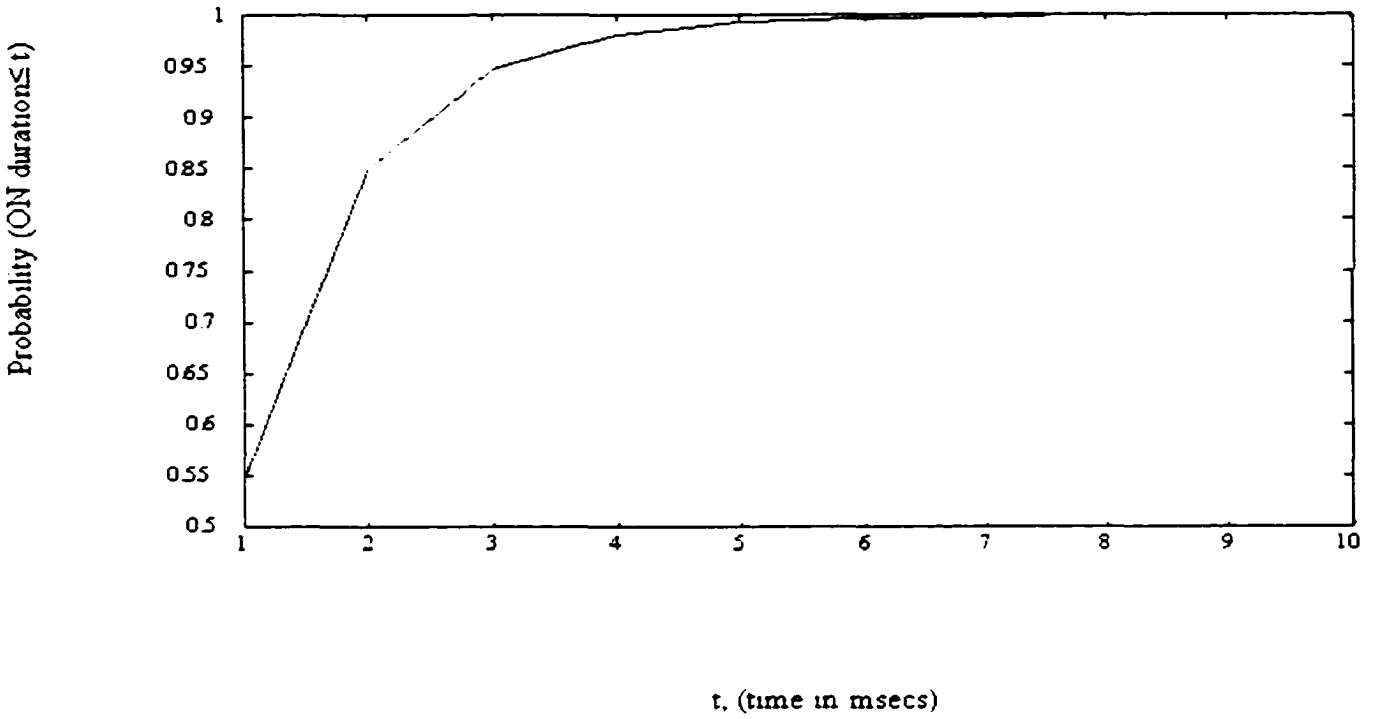


Figure 8.6: Probability distribution of ON state in the combined sources

8.5 Comparison of simulation and analytical expression

We used actual video clips and split them into on/off states using the approach in section 3 described earlier.

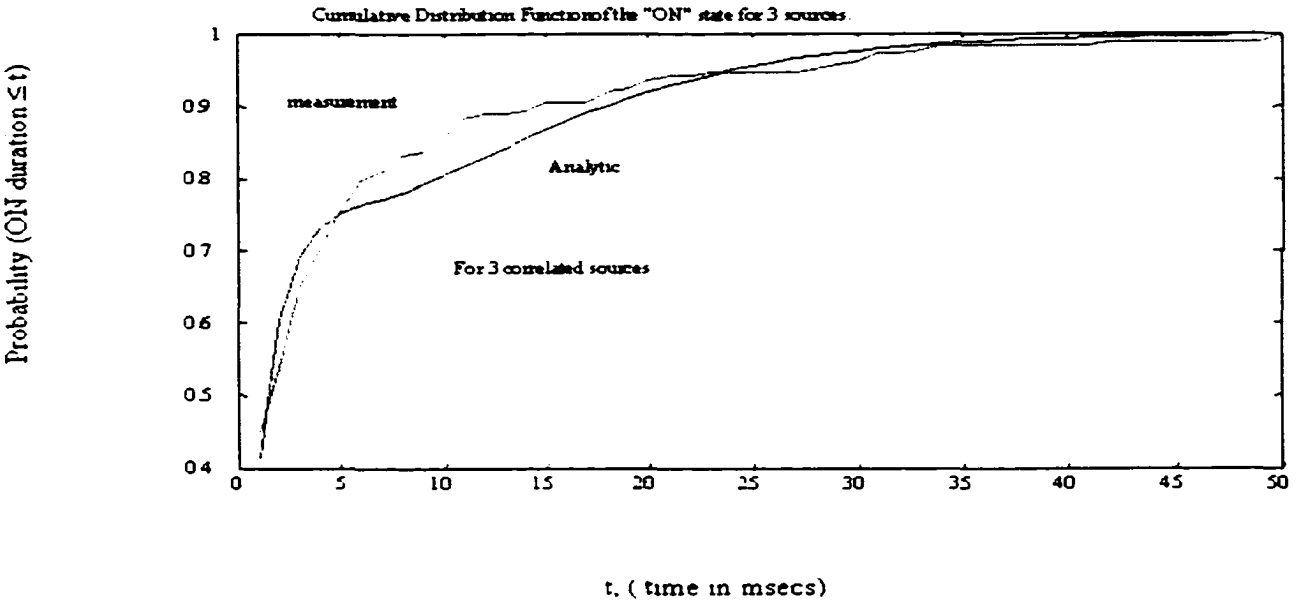


Figure 8.7: Comparison of CDF for ON state-analytical and measurement

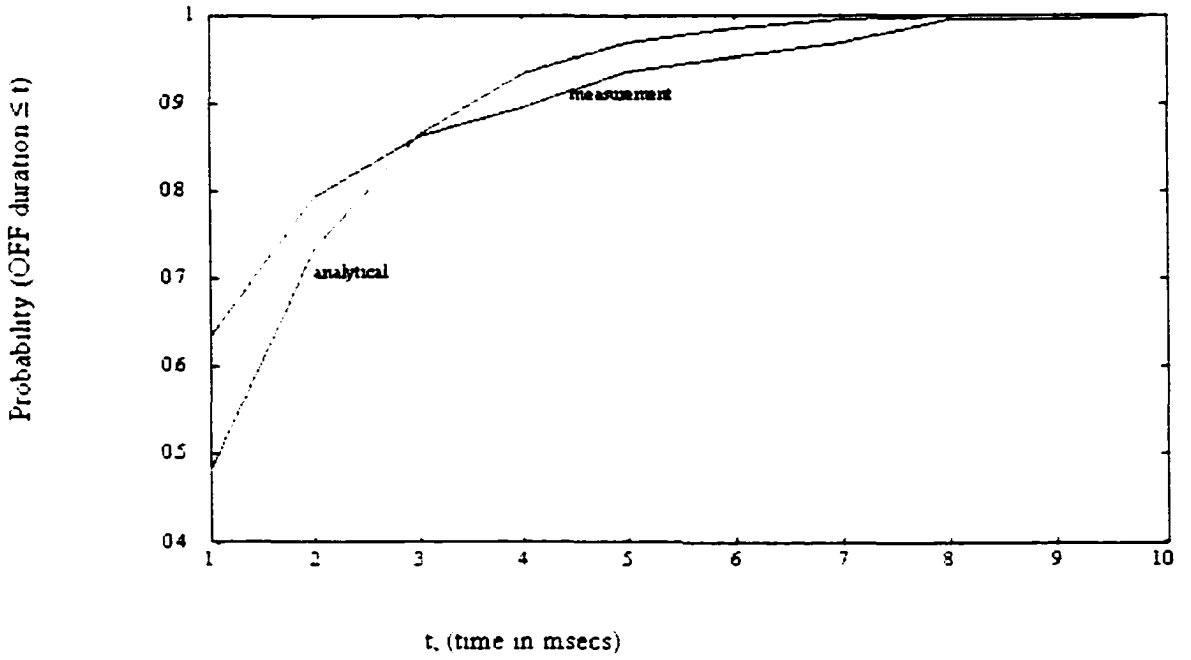


Figure 8.8: Comparison of CDF OFF state-analytical and measurement

In the case of the 'on state', the analytical values is initially below the measured value, then stays, for the most part above the measured value. But the difference in value decreases with time. Similarly, for the 'off state', the analytical values are initially below measurements, but deviates between times 4 and 8 and then closes up again.

9. CONCLUSIONS

We have examined various traffic models, and have particularly worked on the autoregressive model that captures scene changes. We have done so by introducing the periodicity of the I-frames according to the mean scene-length distribution and the mean bit rate distribution of the I-frames. It was shown that more I-frames passed the 'threshold mark test' for a scene change in our model. This is because the mean and standard deviation of the I frames in our model closely matched that of the original trace. The autoregressive model without capturing scene changes, had a mean and standard deviation of I-frames much smaller than the original trace. This shift to the left indicated not too many I-frames passed the 'threshold mark test' for a scene change. When we passed all three traces (the original trace, the AR without scene changes captured, and the AR with scene changes), in a simple queueing network, the loss due to AR model with scene changes closely matched that of the original trace; whereas, the loss due to AR model without scene changes was much lower than the original. This means that our model outperformed the autoregressive model without capturing scene changes.

Additionally, we have established a relation between correlation and effective capacity, and have shown that as the time approaches infinity, that is, at steady state, the effective capacity becomes that of the steady state capacity. At the steady state, correlation is zero, and the effective capacity as a function of correlation boils down to the steady state capacity. Also, we have tried to explain the concept of the 'initial

on state probability,' that depending upon the 'initial on state probability,' effective capacity could be above or below the steady state capacity. Intuitively, when $p_1(0)$ (the probability of the 'on state' at time zero), is chosen less than the 'switch over', the effective capacity will be less than the steady state. This means that the bit rate injected into the network initially will be less than at the steady state. Conversely, when $p_1(0)$ is chosen above the 'switch over', higher bit rates will be injected into the network initially. Network designers ought to take this into consideration, especially when assigning bandwidth to highly correlated traffic such as Mpeg-2 traffic so as to avoid cell loss, or over allocation of bandwidth.

CONTRIBUTIONS.

Our main contribution has been in two areas, namely:

a) The Autoregressive Model b) Impact of correlation on effective bandwidth.

Previous work in the autoregressive model had been applicable to low activity scenes such as videoconferencing. This model successfully captured the frame-to-frame correlation, because frame-to-frame content did not vary that much. With variable bit rate, as in Mpeg –2. With the introduction of Mpeg-2, a modification to the autoregressive model became necessary

Our work in the autoregressive model, not only captures the frame to frame

correlation, but also scene changes. Such scene changes are normally captured in the generation of the intracoded (I) frames. Since the intracoded frames occur periodically, according to the length of the Group of Picture (GOP), these I frames are introduced as such. A further study indicated to us that since our objective was to capture scene changes, then the I frames must be introduced according to the mean of the scene length distribution, and the I frames bit rate distribution.

Another contribution has been in the area of traffic correlation, particularly when it comes to video traffic. Previous authors [3-7] realized the existence of correlation but have assumed the traffic sources to be independent, since they were concentrating on the steady state. Hence the total effective capacity has been the additive sum of the individual effective capacities. Our model takes into consideration both the individual capacities, as well as the correlation among sources. Hence, given a traffic source, we can find the on/off times and the correlation, and hence can calculate the effective capacity at a particular time. Of particular interest is the transient time interval. The difference between the transient capacity and the steady state capacity (no correlation considered), could be significant, leading to considerable loss.

Our effective capacity formula out performs that of models without considering correlation. In the steady state, correlation approaches zero, and hence our model boils down to the steady state formula. Another interesting feature of our model is that, depending upon a parameter called 'switchover,' effective capacity can either be above or below the steady state. Allocating bandwidth to capacity below steady state will lead

to over allocation, and a wastage of bandwidth and under allocation will lead to substantial losses. Network designers can factor this fact when allocating bandwidth.

FUTURE WORKS

As a future work, we will examine the range of standard deviations of the scene length distribution and the distribution of the bit rate of the I frames that lead to a more robust design. Attention will be given to P frames. These also change when there is a scene change. Additionally, we will look at our effective capacity formulation as we increase the number of sources. As further research, we shall look at what happens when there is more than one multiplexer.

APPENDIX

For AIC and Yule-Walker estimates consider this: Given an order P, we use Yule-Walker estimates $\alpha_1,\ldots,\alpha_p$ as parameter for the model AR (P) process of a given time series y_t. They are obtained by solving the following system of linear equations:

$$\begin{bmatrix} \hat{\rho}_0 & \hat{\rho}_1 & \cdots & \hat{\rho}_{p-2} & \hat{\rho}_{p-1} \\ \hat{\rho}_1 & \hat{\rho}_0 & \cdots & \hat{\rho}_{p-3} & \hat{\rho}_{p-2} \\ \vdots & \vdots & & \vdots & \vdots \\ \hat{\rho}_{p-1} & \hat{\rho}_{p-2} & \cdots & \hat{\rho}_1 & \hat{\rho}_0 \end{bmatrix} \begin{bmatrix} \hat{\alpha}_1 \\ \hat{\alpha}_2 \\ \vdots \\ \hat{\alpha}_p \end{bmatrix} = \begin{bmatrix} \hat{\rho}_1 \\ \hat{\rho}_2 \\ \vdots \\ \hat{\rho}_p \end{bmatrix}$$

We estimate the process order P by minimizing Akaike's Information Criterion (AIC) which is defined by

$$AIC \quad (P) = \log \quad \hat{\sigma}_p^2 + \frac{2 \cdot P}{N}$$

where σ_{p^2} is the variance of the residuals.

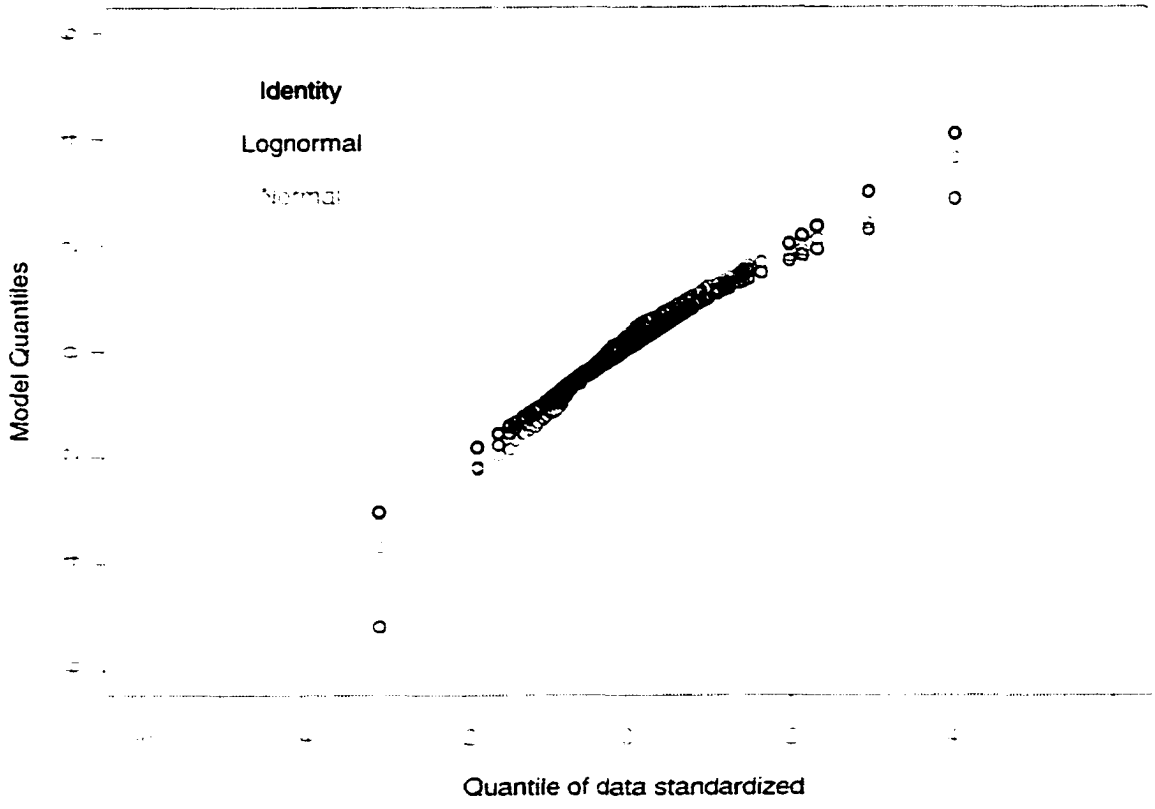


Figure A.1: Q-Q plot for the Zorro GOP

For the probability density function fitting, we use Q-Q plots. Q-Q plots are used to assess whether data has a particular distribution, or whether two data sets have the same distribution. If the distributions are the same, then the plot will be approximately a straight line. The 'quantile' indicates data being segmented into quarters. The line quantile plot connects the first and third quarters.

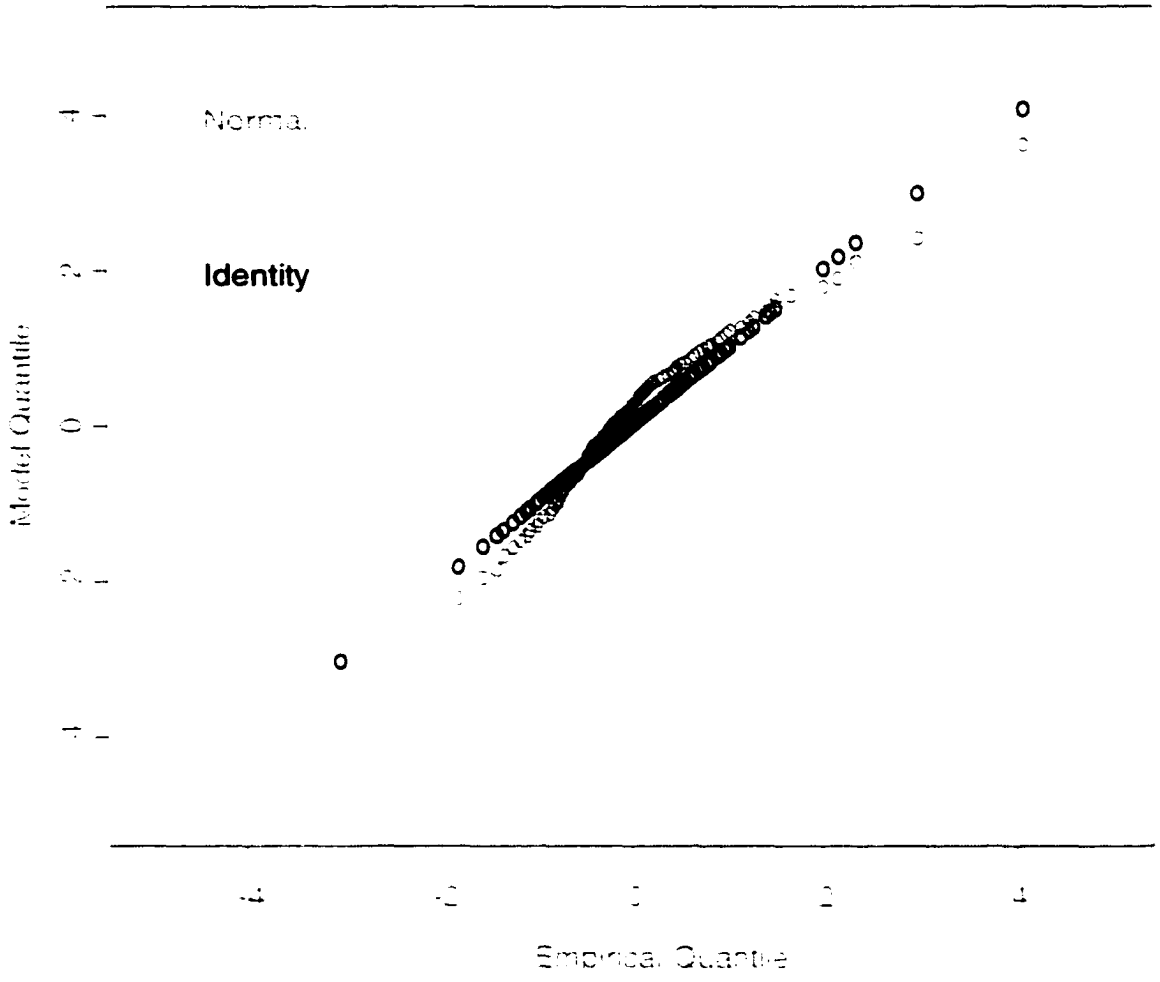


Figure A.2: Q-Q plots for the Zorro GOP

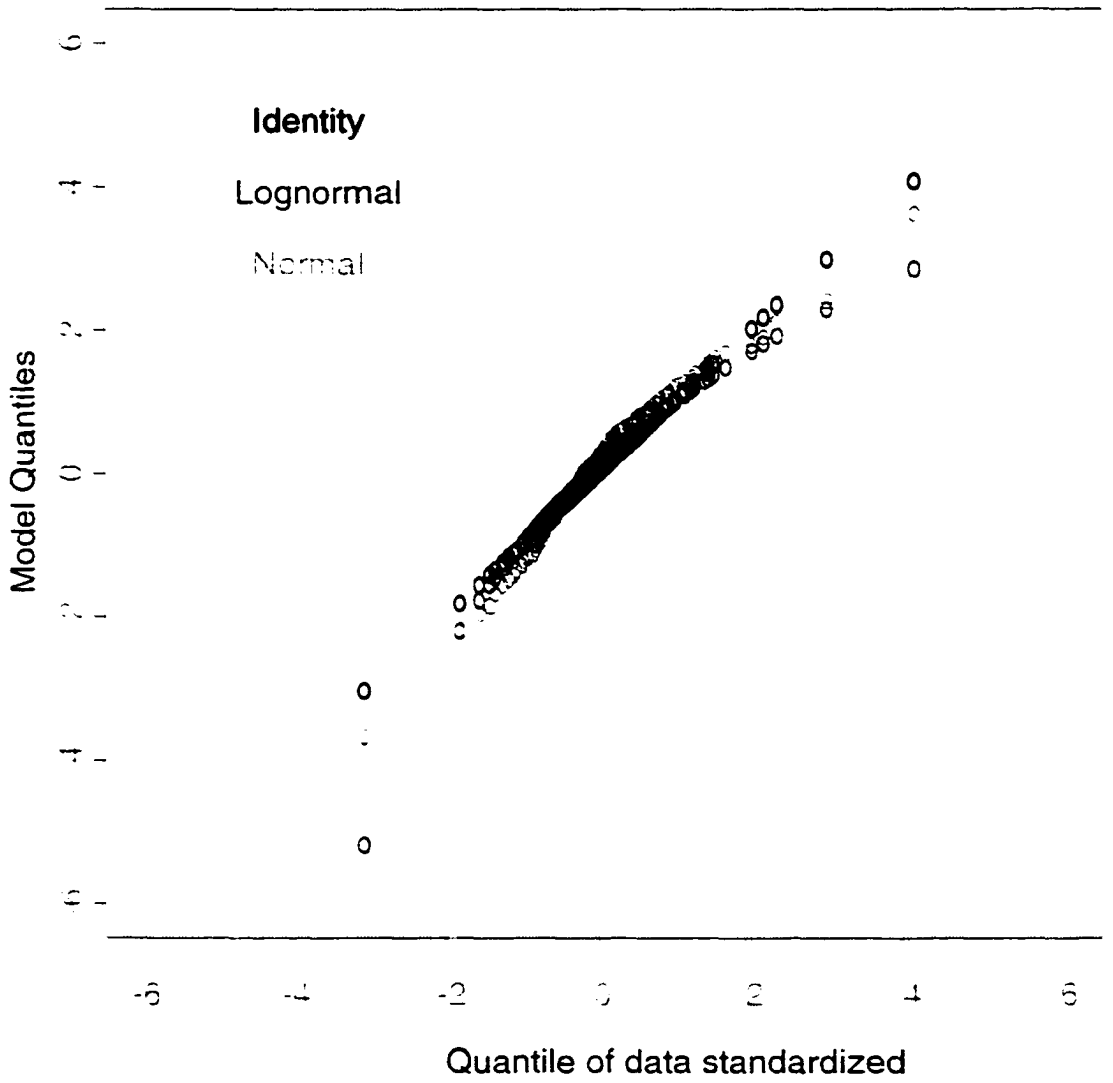


Figure A.3: Empirical vs. Model Q-Q plot

REFERENCES

[1]. Mischa Schwartz. Broadband Integrated Networks. Department of Electrical Engineering, Columbia University, New York, N.Y.

Globecom 1997.

[2] Roberts, J.W., J. Guibert, and A Simonian (1991). Network performance considerations in the design of a VBR codec. In Proceedings of the ITC-13 Workshop on Queueing, Performance and Control in ATM, pp.77-82.

3] Elwalid, A.I., and D. Mitra, "Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High-Speed Networks,"IEEE/ACM Trans. on Networking, 1,3 (June 1993): 329-343.

[4] Kelly, F.P., "Effective Bandwidth of Multi-Class Queues," Queueing Systems, 9 (1991):5-16.

[5] Anick, D., et al ., "Stochastic Theory of a Data-Handling System with Multiple Sources," B System Tech. Journal., 61, 8 (Oct. 1982): 1871-1894.

[6] Gibbens, R.J., and P.J. Hunt, "Effective Bandwidths for the Multi-Type UAS Channel, "Queueing Systems, 9 (1991):17-28.

[7] Kosten, L., "Stochastic Theory of Data-Handling Systems with Groups of Multiple Sources, "Proc. 2nd Internatl. Symp. On Performance of Computer Commun.Systems, North-Holland, 1984.

[8] H. Heffes and D. Lucantoni. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. IEEE J. Selec. Areas in Commun., SAC-4, 6, pp.856-868(1986).

[9] Barry G. Haskell, Atul Puri, Arun N.Netravali. Digital Video: An Introduction to Mpeg-2. New York. 1997.

[10] Marwan Krunz and Satish K. Tripathi, ``On the characterization of VBR MPEG streams." Performance Evaluation Review (Proceedings of ACM SIGMETRICS'97 Conference), Vol. 25, No. 1, pp. 192-202, June 1997

[11] O. Rose. Statistical Properties of Mpeg Video Traffic and their impact on traffic modeling in ATM systems. Ieee Computer Society Press, 1995

[12]M.E. Crovella and A. Bestavros. Self-Similarity in World-Wide Web Traffic: Evidence and Possible Causes. In IEEE/ACM Transactions on Networking, vol.5, no.6, pp.835-846, Dec. 1997.

[13] M. Garrett and Willinger. Analysis, Modeling and Generation of Self-Similar VBR Video Traffic. In Proc. Communication Architectures, Protocols and Applications, ACM SIGCOMM, 1994.

[14] Hosking, J. R. M. (1984) Modeling persistence in hydrological time series using fractional differencing. Water Resources Research 20(12), 1898-1908.

[15] P. Skelly, M. Schwartz, and S. Dixit(1993). A histogram-based model for video traffic behavior in an ATM multiplexer. IEEE/ACM Transactions on Networking 1(4), 446-459.

[16] Shroff, N. and M.Schwartz (1994). Video modeling within networks using deterministic smoothing at the source. In Proceedings of the Infocom '94, pp. 342-349.

[17]A.M. Law and W.D. Kelton. Simulation Modeling and Analysis. McGraw-Hill, Inc., 2nd edition, 1991. New York

[18] Heyman, D. P. and T.V. Lakshman (1994). Source models for VBR broadcast video traffic. In Proceedings of the Infocom '94, pp. 664-671.

[19] Maglaris, B., D. Anastassiou, P. Sen, G. Karleson, and J. D. Robbins (1988). Performance models of statistical multiplexing in packet video communications. IEEE Transactions on Communications 36(7), 834-844.

[20] Nomura, M., T. Fujii, and N. Ohta (1989). Basic characteristics of variable rate video coding in ATM environment. IEEE Journal on selected Areas in Communications 7(5), 752-760.

[21] Ramamurthy, G. and B. Sengupta (1992). Modeling and analysis of a variable bit rate video multiplexer. In Proceedings of the Infocom '92, pp. 6C.1.1-11.

[22] Chowdbury, S. and K. Sohraby (1994). Bandwidth allocation algorithms for packet video in ATM networks and ISDN Systems 26, 1215-1223.

[23] Enssle, J. (1994). Modeling and statistical multiplexing of VBR MPEG compressed video in ATM networks. In Proceedings of the 4th Open Workshop on High Speed Networks, Brest, France, pp 59-67.

[24] Adas, A. and A. Mukherjee (1995). On resource management and QoS guarantees for long range dependent traffic. In Proceedings of the Infocom '95, pp. 779-787.

[25] Stokes, O.L. (1995). Transmission of MPEG Compressed Video Through B-ISDN ATM Networks. Ph.D thesis, North Carolina State University.

[26] Norros, I. (1994). A storage model with self-similar input. Queueing Systems 16, 387-396.

[27] Likhanov, N., B. Tsybakov, and N.D. Georganas (1995). Analysis of an ATM buffer with self-similar ("fractal") input traffic. In Proceedings of Infocom '95, pp.985-

992.

[28] Huang, C., M. Devetsikiotis, I. Lambadaris, and A.R. Kaye (1995). Modeling and simulation of self-similar variable bit rate compressed video: A unified approach. In Proceedings of the ACM SIGCOMM '95 Conferene.

29]W.C. Feng and J. Rexford. Comparison of Bandwidth Smoothing Techniques for theTransmission of Pre-recorded compressed Video. In Proc. INFOCOM, IEEE Comm. Society, 1997

[30]J. Rexford, S. Sen, J. Dey, W. Feng, J. Kurose, J. Stankovic, and D. Towsly. Online Smoothing of Live, Variable- Bit-Rate Video

. [31]. K. Sohraby, A. Hussain, M.A. Ali. Traffic Correlation in a Real-Time Multimedia Conference. Proceedings of the 2nd Ieee Symposium on Computers & Communications Alexandria, Egypt, pp 390-396, July, 1997.

[32] A. Hussain, K. Sohraby, M.A. Ali. A Novel Two-Queue Model for ATM Networks.

[33]Daniel P. Heyman, Ali Tabatabai, T.V. Laksman. Statistical Analysis and Simulation Study of Video teleconferencing Traffic in ATM Networks. Ieee Transactions on circuits and systems for video technology,. vol.2,.No.1, March 1992

34] A. La Corte, A. Lombardo, G. Schemba. Modeling of Superposition of ON-OFF Correlated Traffic Sources. IEEE Infocom 1995 . 993-1000.

[35] K. Onda and K. Nakagawa. Approximation of Video Cell Traffic by AR(1) + IPP Model. Electronics and Communications in Japan, Part 1, Vol. 78, No. 8, 1995

[36] Lucent Technologies/ Bell Labs, R.J.T.Morris (Editor) et al. Queueing + Analysis Software. Vol. 1, User's Guide and Reference Manual Version 1.0 (04-01-98)

[37] A. Pappoulis. Probability, Random Variables and Stochastic Processes, 3rd Edition. McGraw-Hill, Inc., New York.

[38]. Krunz, M., R. Sass, H. Hughes (1995). Statistical Characteristics and Multiplexing of MPEG streams. In Proceedings of the Infocom '95, pp.455-462.

[39] K. Ravindran. Private Discussions on "network bandwidth allocation strategies", 1999

[40] Chi-Tsong Chen. Linear System Theory and Design., CBS COLLEGE PUBLISHING., Holt, Rinchart and Winston. New York.

[41] Stephen Wolfram. Mathematica, 2nd edition. Addison-Wiley Publishing Company, Inc. New York.

[42] Williams L.Hays. Statistics, 5th edition. Harcout Brace College Publishers, New York.

[43]Walter Willinger, Murad S. Taqqu, Robert Sherman, Daniel V. Wilson.Self-Similarity Through High- Variability:Statistical Analysis of Ethernet LAN Traffic at the Source Level. IEEE/ACM TRANSACTIONS ON NETWORKING VOL. 5, NO. 1, FEBRUARY 1997.

BIOGRAPHY

Christopher Amo-Quarm was born at Prestea, a busy gold-mining town in Ghana (formerly called the Gold Coast), West Africa. He earned a Bachelor's degree in Physics in 1981, and taught various high schools in Ghana, Nigeria and the United States. He was privileged to teach in Peter Stuyvesant, a prestigious high school, at Chamber's Street in New York City.

Chris decided to change his career by getting a second degree in Electrical Engineering in 1992 while still teaching. In 1996 he earned a Master's degree in Electrical Engineering and is currently a Ph.D candidate.

In 1998, Chris worked as a summer intern at Bell Labs, Lucent Technologies, at Holmdel, New Jersey, the research arm of Lucent Technologies. He worked on GSM reliability and joined a team of three in presenting the work in Nuremberg, Germany. He had the rare privilege of continuing to do his research at Bell labs.

Recent Publications Include:

C. Amo-Quarm, K. Sohraby, K. Ravindran. "A Relation Between Correlation and Effective Capacity", "International Symposium on Performance Evaluation of Computer and Telecommunication Systems", Florida, July 15 – 19, 2001, p306 – 311

C. Amo-Quarm, M. Mezhoudi. "Capturing Scene-Changes in the Autoregressive Model", Communications Networks and Distributed Systems Modeling and Simulation, San Antonio, Texas, January 27-31, 2002.

C. Amo-Quarm, M. Mezhoudi, K. Ravindran. "Improved Autoregressive Model." 2002 International Zurich Seminar on Broadband Communications. February 19-21, ETH, Zurich, Switzerland.

C. Amo-Quarm. "Effective Capacity in Transition." ." 2002 International Zurich Seminar on Broadband Communications. February 19-21, ETH, Zurich, Switzerland.