A Survey of VBR Video Traffic Models

Savera Tanwir and Harry Perros, *Fellow, IEEE*

***Abstract*—We have seen a phenomenal growth in video applications in the past few years. An accurate traffic model of VBR video is necessary for performance evaluation of a network design and also for creating synthetic loads that can be used for benchmarking a network. In view of this, various models for VBR video traffic have been proposed in the literature. In this paper, we classify and survey these models. In addition, we implemented four representative video traffic models and compared them using the H.264 AVC video traces available at the Arizona State University video traces library. These models are: the Markov Modulated Gamma (MMG) model, the Discrete Autoregressive (DAR) model, the second order Autoregressive AR(2) model, and a wavelet-based model. The results show that the MMG and the wavelet-based models are suitable for both video conference and IPTV, while the DAR model is good for video conference traffic only. According to our results, the AR(2) model is not suitable for generating any type of H.264 video. A brief overview of SVC, HD, and 3D video is also provided.**

***Index Terms*—VBR video, Video traffic model, H.264.**

I. Introduction

OVER the last decade, we have seen an exponential growth in media applications, and particularly in video applications. Today, most viewers receive digital television with high definition services and greater choice of channels. High definition DVDs, Blu-Ray disks and NetFlix streaming are increasing in popularity. An ever-increasing number of users upload and download videos using sites like YouTube. Recording and sharing of videos using mobile phones is widespread. Video calling over the Internet is commonplace with applications, such as Skype. Large businesses and organizations use video conferencing applications like Cisco's TelePresence [1] and WebEx [2] for face-to-face collaboration across different geographic regions. The growing number of multimedia users has increased exponentially the bandwidth requirements. Consumers are increasingly discerning about the quality and performance of video-based products, and therefore, there is a strong incentive for continuous improvement in multimedia technologies.

A major component of multimedia networking is the data compression (source coding) of multimedia data sources i.e. speech, audio, image and video. Video compression or video encoding is the process of reducing the amount of data required to represent a digital video signal, prior to transmission or storage [3]. Once the data is compressed, the bit stream is packetized and sent over the Internet. The complementary operation, decompression or decoding, recovers a digital video signal from the compressed representation before display. Rate control is an essential part of most video encoders. It determines the number of bits or the quality level of the encoded frame. There are two types of rate control: constant bit rate (CBR) and variable bit rate (VBR). For CBR video coding, rate control designers focus on improving the matching accuracy between the target bit rate and the actual bit rate and satisfying the low latency and buffer constraints [4]. As a result we see fluctuations in video quality due to scene changes and other video content. In cases where rate or delay constraint is not as strict as in real-time video, VBR can be used to maintain constant quality.

Various standards have been developed for video encoding, i.e., H.261 [5], H.263 [6], MPEG-1, MPEG-2, MPEG-4 [7] and H.264 [8]. H.264/MPEG-4 AVC represents a big leap in video compression technology with typically a 50% reduction of the average bit rate for a given video quality compared to MPEG-2 and about a 30% reduction compared with MPEG-4 Part 2 [9]. Auwera et al have studied the traffic characteristics of variable bit rate MPEG-4 part 2, H.264/AVC and H.264/SVC [10] single-layer encoded video in [11]. They generated video traces for video sequences using the MPEG-4 part 2 Microsoft v2.3.0 software, the JM reference software (version 10.2) for H.264/AVC and the JSVM SVC reference software (version 5.9). The authors in [11] used the Rate Distortion and Variability Distortion parameters to compare the different codecs. Rate Distortion is the video quality (PSNR) as a function of average bit rate. The Variability Distortion curve is a plot of the Coefficient of Variation (CoV) of the frame size as a function of PSNR, where CoV is defined as the ratio of the variance of the frame sizes divided by the mean frame size. The results show that H.264/AVC, and H.264 SVC codecs lead to significant average bit rate savings with respect to the MPEG-4 Part 2 codec. At the same time, the variability of the H.264/AVC and H.264 SVC video traffic is significantly higher than the variability of the MPEG-4 Part 2 video traffic. The comparison between classical B frames (default in H.264/AVC) and hierarchical B frames (H.264 SVC), based on four GoP structure patterns indicates that hierarchical B frames outperform classical B frames at the expense of higher rate variability.

In this paper, we classify and survey VBR video traffic models proposed in the last twenty years. In addition, we implemented four representative video traffic models and compared them using H.264 AVC video traces available at the Arizona State University video traces library. A brief review of models for Scalable Video Coded (SVC), High Definition (HD), and Three Dimensional (3D) video is also given. The paper is organized as follows. In the following

Manuscript received April 5, 2012; revised September 20, 2012.

S. Tanwir is a graduate student at the Department of Computer Science, North Carolina State University, Raleigh, NC, 27607 USA (e-mail: stanwir@ncsu.edu).

H. Perros is a professor at the Department of Computer Science, North Carolina State University, Raleigh, NC, 27607 USA (e-mail: hp@csc.ncsu.edu).

Digital Object Identifier 10.1109/SURV.2013.010413.00071

section, we present five classes of VBR video models, and in the subsequent sections 3 to 7 we survey the relevant literature within each class. In section 8, we provide a brief overview of SVC, 3D and HD video. In section 9, we compare the four models. The conclusions are given in section 10.

II. Video Traffic Modeling

An accurate traffic model of VBR video is necessary for evaluating the performance of a network design [12]. We can do this by performing a live experiment using real networks and real sources. However, testing real networks is quite expensive and often it is difficult to generate reasonable results. An alternative to this is to model the traffic using mathematical analysis or simulation. Trace-driven simulations are considered credible as they represent an actual traffic load, but they are static and they provide only a point representation of the workload space [13]. Another disadvantage of using traces is the difficulty in adjusting parameters and extending the trace if there is a need to continue the simulation beyond the number of packets/frames in the trace file. Statistical and mathematical traffic models, on the other hand, are considered better as they can be used to provide a better understanding for various traffic characteristics. This is because they are stochastic in nature, and hence different realizations that represent the actual data can be obtained by varying model parameters.

A good traffic model should capture the characteristics of video sequences and accurately predict network performance (e.g., end-to-end delay and packet loss). Among the various characteristics of video traffic, the following two are of major interest: 1) the distribution of frame sizes; and 2) the Autocorrelation Function (ACF) that captures common dependencies between frame sizes in VBR video [14]. A common method used to match distributions is the Q-Q plot [15], [16]. Capturing the ACF structure of VBR video traffic is more challenging due to the fact that VBR traces exhibit both Long-Range Dependent (LRD) and short-range dependent (SRD) properties [14]. Stochastic processes can be classified, from the ACF point of view, into three types: independent, Short-Range Dependent (SRD) and long-range dependent (LRD). An independent process is always uncorrelated. If the autocorrelation function is summable (e.g., when it decays exponentially fast), then it is referred to as an SRD process, but if it is not summable (e.g., when it decays slowly), then the source is referred to as an LRD process [17].

In [12] the author suggests that a good video model can be evaluated by four criteria. First, the model should match certain statistical characteristics of a real video sequence, i.e., probability density function, mean, variance, peak, autocorrelation and coefficient of variation of the bit rate. Second, the synthetic video sequence should be similar to the real video sequence. Third, the model should be simple and be able to generate a synthetic video sequence with low computational complexity. Lastly, the model should characterize a wide range of video sources ranging from low to high motion activity.

Several models for VBR video traffic have been proposed in the literature. These models can be broadly classified into the following five categories:

1) *Autoregressive (AR) models*: The next frame size in a video sequence is obtained as an explicit function of previous ones within a time window.
2) *Models based on Markov processes*: Markov processes/chains are used to represent bit-rate regimes or frame/GOP sizes.
3) *Self-similar and fractional ARIMA models:* They capture the long range dependence of compressed video traffic.
4) *Wavelet models*: Wavelet transform techniques are used to capture both LRD and SRD properties of video traffic.
5) *Other approaches:* It includes different models such as the M/G/∞ process and the Transform-Expand-Sample (TES) models.

In the following sections, we survey the video traffic models within each category. A few survey papers have previously been published in the area of VBR video modeling [12], [17], [18], [19]. Two of these papers [17], [18] survey the models falling in the AR, Markovian, and self-similar categories. However, since these papers were published in 1999, they do not cover the VBR traffic models proposed in the last thirteen years. The survey paper by Al Heraish [12] reviews the AR models for video-conference type traffic only and it also does not cover many of the latest models. Similarly, in [19] the author surveys the AR models for full motion videos only. To the best of our knowledge, a detailed survey of the different types of VBR models falling in the categories described above has not been conducted and this is the first detailed survey covering these VBR models proposed in the last two decades.

III. The Autoregressive Models

Many VBR source models are based on the Autoregressive (AR) process. In this section, we first provide an overview of the AR process, and then we discuss the various types of AR based models that have been proposed in the literature.

A. Review of the Autoregressive Process

In an AR process, the current value is a function of a weighted linear combination of past values. An AR process is generally described as:

$$x\,(n) = \sum_{i=1}^{p} a_i x\,(n-i) + \, e(n)$$

where $a_1, a_2, \ldots, a_p$ are AR coefficients and p is the order of the AR process. The sequence $e\,(n)$ consists of i.i.d (independent and identical distributed) random variables, known as the residual (or error process), that give the AR its stochastic nature. The residuals are uncorrelated and often assumed normally distributed with zero mean and variance σ^2. There are a number of methods to estimate the parameters of AR process, one such method is linear prediction.

The AR process is simple as it requires few parameters. When modeling video traffic using the AR process, $x\,(n)$ represents the bit rate of the coded video during the nth frame or the size in bytes of the nth frame, $e\,(n)$ is assumed to be a Gaussian process with zero mean and variance σ^2 (estimated from an empirical video trace) and lastly a_i, $i = 1, 2, ..., p$ is the lag i autocorrelation of the successive frame rates.

Autoregressive models, in general, appear to capture the autocorrelation behavior of compressed video sources, which is an essential element for any model of compressed video sources. However, owing to the nature of video traffic that exhibits a dynamic and complex structure generated from different compression schemes, finding an appropriate AR model that can depict different statistical characteristics can be quite challenging [12].

Generally, video models are classified based on two types of video: video conference and full motion video. Video conference type video consists of video scenes in which one or several people are talking with very little movement and almost unchanged background. The basic characteristics of such traffic are: bell-shaped distribution for frame sizes and high inter-frame auto-correlations which typically decay exponentially [12]. Several models based on AR process have been proposed for video conference type of video only. These include simple AR models, the Discrete Autoregressive process model (DAR(1)), the Gamma Autoregressive model (GAR), the Gamma-Beta Autoregressive model (GBAR), the Continuous DAR (C-DAR) model and the general AR model. These models are discussed in detail later in this section.

In contrast to video conference, full motion video exhibits a wide range of video scenes with low, medium or high activity level. It includes background and foreground with frequent scene changes as in movies, TV programs and sports broadcast. This type of video requires algorithms that employ sophisticated techniques like block-based motion compensation and Discrete Cosine Transform (DCT) based compression. The use of motion compensation reduces the temporal redundancy in the video sequence while DCT reduces spatial and perceptual redundancy. The fluctuations in the overall scene activities produce a series of variable bit rate sequences. Scenes with high degree of movement generate frames with a high bit rate and scenes containing lower movement generate frames with low bit rate. The highest bit rates arise during scene changes. This results in video traffic with different statistical characteristics during different motion periods and higher bit rate relative to video conference traffic. Within each motion period, there is a strong correlation between the bit rates of successive frames. Many AR models for full motion video have also been proposed in the literature. However, unlike video conference, full motion video cannot be represented by a single AR process and more elaborate models that capture the statistics associated with different time scale present in the video sequence and coding schemes are required. The AR models for full motion video that have been reported in the literature include: the frame based AR(2) model, the projected AR model, the nested AR model, the scene-based AR(1) model, and a GBAR based model for MPEG video with different frame types called GOP-GBAR.

For video traffic that constitutes of different types of frame I, P and B like MPEG video, a single AR process is not sufficient for the three types of frames. This is because these different frame types have different frame-size distributions, mean frame sizes and auto-correlation function. Therefore, a combination of AR processes with a different residual process for each frame type is required. This applies to both video-conference and full motion video traffic.

B. Survey of Autoregressive Video Traffic Models

In this section, we review and discuss the AR process based models that have been proposed in the literature for both video conference and full motion video traffic. We have categorized the models based on the type of AR process.

1) Simple AR Models: The AR(1) is one of the earlier models proposed for video conference type video. In [20], Nomura et al have used a simple AR(1) model for VBR video conference traffic in ATM environment. The AR(1) process is of the form $x(n) = a_1 x(n-1) + e(n)$, where $x(n)$ is the bit rate of the coded video during the nth frame, a_1 is the lag-1 correlation of the successive frame rates and $e(n)$ is a residual following the normal distribution. Two 20-second video sequences were analyzed, one containing scene changes and the other without scene changes. The results suggested that video without scene changes can be accurately modeled by the AR process. However, if the conference video is composed of several scenes, the video source should be modeled by multiple AR processes.

Heyman et al presented different flavors of the AR model in the 90's for teleconference traffic over an ATM network. In [21], Heyman et al analyzed multiplexed traffic from teleconference video sources and proposed a source model for teleconference traffic over ATM networks. The model was evaluated using 30 minutes sequence of video teleconference data with 25 frames/sec. A single-stage multiplexer model was used where the multiplexer consists of a server transmitting cells at a specified line rate and a buffer whose size is determined by the delay constraints on cell transmissions. Cells arrive to the multiplexer from a number of video sources. During an interframe period, each source generates a frame consisting of a variable number of cells. The interframe period is 40 ms for PAL standard systems. Unlike some previous studies, it was observed that the number of cells per frame is not normally distributed. Instead, it follows a gamma (or negative binomial) distribution. It was concluded that the autoregressive model of order 2 fits the data well, but it does not produce enough large values to be a good model for traffic studies. Therefore, the authors proposed a better model by using the Discrete Autoregressive DAR(1) process [22] described in III-B2 below.

2) The DAR(1) process based Models: A discrete autoregressive model of order p, denoted as DAR(p), generates a stationary sequence of discrete random variables with an arbitrary probability distribution and with an autocorrelation structure similar to that of an autoregressive model [22]. DAR(1) is a special case of a DAR(p) process and it is defined as follows: let $\{V_n\}$and $\{Y_n\}$be two sequences of independent random variables. The random variable V_n takes the values 0 and 1 with probability 1 - ρ and ρ respectively.The random variable Y_n has a discrete state space S and $P\{Y_n = i\}= \pi(i)$. The sequence of random variables $\{X_n\}$which is formed according to the linear model:

$$X_n = V_n X_{n-1} + (1 - V_n) Y_n \tag{1}$$

is a DAR(1) process. A DAR(1) process is a Markov chain with a discrete state space S and a transition matrix: $P = \rho I + (1-\rho) Q$, where ρ is the lag-1 autocorrelation coefficient,

I is the identity matrix and Q is a matrix with $Q_{ij} = \pi(j)$for $i, j \epsilon S$. The Q matrix consists of the negative binomial probabilities $\{f_0, f_1, \ldots, f_k, F_K\}$, where $F_k = \Sigma_{k>K} f_k$ and K is the peak rate. Each k, $k < K$, corresponds to a possible source rate less than the peak rate K. Therefore, the parameters that are needed are now only the peak rate, mean, variance, and the lag-1 autocorrelation coefficient; the same parameters needed for the two-state model plus the variance. The model was shown to be accurate when several sources are multiplexed, but not as effective for a single video source.

In another paper, Heyman et al [23] described an ATM frame-layer model for video sequences incorporating scene changes. The sequences were generated using a DPCM [24] codec. The model consists of three different stochastic processes: (1) scene length, (2) size of the first frame after a scene change, and (3) size of frames within a scene. The authors used 11 different video sequences with varying peak-to-mean ratios. It was found that scene length distributions fit Weibull, gamma and Pareto distributions, while the number of cells within a scene change frame fit Weibull, gamma, and in one case normal distributions (some sequences, however, did not appear to fit any distribution). The frame following a scene change frame is generated using a linear prediction of the form

$$Y_i = a + bX_i + \varepsilon_i \tag{2}$$

where a and b are fixed coefficients, X_i is the number of bytes in frame i and ε_i is white noise. Since the frame size within a scene is correlated, a Markov chain is used where each state represented the integer part of $X_i/50$ and the transition probability matrix was determined using a DAR(1) process. This groups similar frame sizes in each state. The Q matrix consists of the Pareto probabilities $\{f_0, f_1, \ldots, f_k, F_K\}$,where $F_k = \Sigma_{k>K} f_k$ and K is the peak rate. The simulation comprised of multiplexed transmission of several video connections on a single ATM link. Two PAL standard video sequences were used for the simulation tests. The results showed that the DAR(1) model was an accurate model for one sequence but it overestimated the cell loss for the other. If more accuracy is needed, a Markov chain model can be used but the drawback is that there are many more parameters. A single model that can be used for all sequences did not seem possible.

Recently, Lazaris and Koutsakis [25] proposed a DAR(1) model which captures the behavior of multiplexed H.264/AVC videoconference sources. The authors used traces of content and traffic characteristics (high autocorrelation, low or moderate motion) similar to those in videoconference traffic. They studied four different long sequences of H.264 VBR encoded videos in 20 formats, from the publicly available video trace library of [26], [27], in order to derive a statistical model that fits well the real data. Results show that the best fit among these distributions is achieved for all the traces studied with the use of the Pearson type V distribution [28]. Thus, in the proposed model the rows of the Q matrix consist of the Pearson type V probabilities. The authors also conducted a simulation study, where the video model was fed into a queue. The results for the packet dropping probability and the cumulative density functions of the waiting time in the queue were compared against those obtained by simulating the queue with the real traces. The results obtained verified the validity of the model. Results also show that, due to the fact that the modeling of I frames sizes is not perfectly accurate, there is a small difference in video packet dropping between the models and the actual traces at higher loads.

In [29], Xu et al proposed a continuous-time Markov chain model, referred to as a Continuous Time Discrete State AR (C-DAR(1)), which is based on the DAR(1) model. This model is suitable for theoretical analysis. The C-DAR model has the same steady-state probability distribution and exponential autocorrelation function as the DAR(1) model. The main contribution of this approach is a method to calculate the transition rate matrix Q (generator) of C-DAR(1) from the transition matrix P of the DAR(1) model.

3) Frame based AR Models: In [30], [31], Krunz and Tripathi presented a model for MPEG video for all frame types, i.e., I, P and B, incorporating scene changes. The authors analyzed the ACF and frame size distributions for multiple video traces and showed that the frame size distribution of all three types of frames is lognormal while the scene length distribution can be approximated by a geometric distribution. The scene changes are detected by a considerable change in the size of consecutive I frames. The effect of a scene change on P and B frames is negligible and can be ignored. Therefore, the size of I frame is modeled by two random components: a scene related component and an AR(2) component that accounts for the fluctuations within a scene. The sizes of P and B frames were modeled by i.i.d random processes with lognormal marginal. The complete model is obtained by mixing the three sub-models based on the GOP pattern.

In [32], [33], Koumaras et al presented a model that uses two discrete processes for inter and intra frame generation for the three different types of frames. The first process models the intra-scene state, within which the frame size, for frames of the same type, retains the characteristics of the previous frame. The size of I frames, which are strongly responsible for the inter-GOP correlation of the video stream, is obtained using the expression $F_n^I = \alpha_1 F_{n-1}^I$, where F_n^I is the size of nth I frame. The second process models the inter-scene state and the frame size of each type is generated using an AR(1) process of the form $F_n^x = \alpha_1 F_{n-1}^x + e(n)$, based on the size of the previous frame size, where α_1 is the lag-1 autocorrelation parameter, x is the frame type and $e(n)$ is a residual following the normal distribution. Although inter-GOP correlation, described by the ACF of I frames, is an important measure, another aspect of video traffic is the lag-1 correlation between I, P, and B frames within the same GOP (intra-GOP correlation). These correlation coefficients are calculated between the first neighboring I-P, I-B and P-B frames of each GOP structure over the entire trace. The proposed algorithm uses these correlations to determine the P and B frame sizes in a new GOP as shown in figure 1. However, it is not clear from the paper how the scene changes are modeled and when does the model shifts from one process to the other. The authors presented results for the H.264 video encoded with quantization scale 20 for all the frames and GOP length 12. Various synthetic traffic of short 3 min duration each were generated by the proposed model and compared against various actual samples.

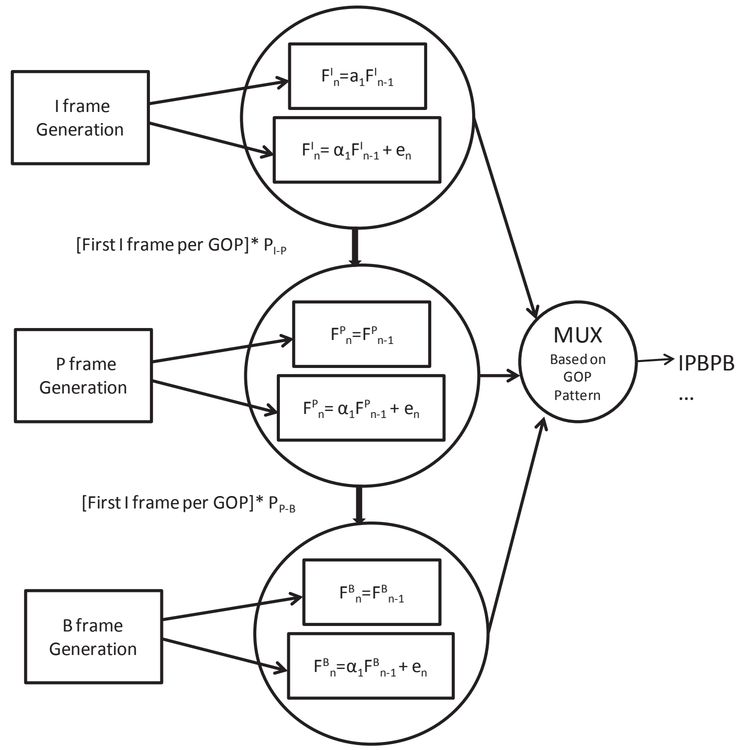

Fig. 1. Block Diagram of the model by Koumaras et al [30]

4) Nested AR Model: Building on the work by Krunz and Tripathi [31] described earlier, Liu et al [34] proposed a nested AR model which is a modified version of the scene based AR model. The nested AR model takes scene changes into account and uses the hybrid Gamma/Pareto distribution for all three types of frames in MPEG-encoded video sequences. The scene changes are incorporated in modeling the I frame sequence using two second-order AR processes nested within each other. The first AR process models the long-range dependence and is used to generate the main frame size of the scenes, and the second AR process models the short-range dependence and is used to generate the fluctuations within the scene. The nested AR model was compared to the scene based AR model and it was shown that it gives rise to better autocorrelation at both small and large lags than scene based AR model.

5) AR Models with Gamma Residual Process: Xu et al [35], [36] analyzed a first-order Gamma AR (GAR) process with gamma distributed residuals for modeling video conference traffic. The model generated a sequence with nearly similar statistical characteristics to real video trace and the results also showed that the GAR model outperformed the DAR model. However, the problem of the GAR model is that it is difficult to generate the residual process if the order of the AR process is increased. It performed well only for the special case of first-order AR process.

The Gamma-Beta Autoregressive (GBAR) model was first introduced by McKenzie and later used by Heyman [37] who validated its use as a source model for VBR video conferencing. In the GBAR model the AR coefficients A_n are beta distributed and the residuals B_n are gamma distributed. Hence, the first order AR process X_n is given by the following expression:

$$X_n = A_n X_{n-1} + B_n \tag{3}$$

Implementing the GBAR process only requires the ability to simulate i.i.d. gamma and beta random variables. The parameters of the gamma and beta distribution were estimated from the real video traces by matching the mean and variance. The GBAR model was simulated using three different video conference sequences of 30 minutes each with moderate motion and scene changes. All coders used a version of H.261 video coding standard. The authors state that although the GBAR model is shown to be more accurate than the DAR model, it is not suitable for studying admission control algorithms in ATM networks. Based on the GBAR model, Frey and Nguyen-Quang [38] developed a model for an MPEG video containing I, P, and B frames referred to as the GOP-GBAR model. This model explicitly accounts for GOP cyclicity and is a generalization of the GBAR model for video teleconferencing. The model was fitted to six video frame sequences. The results for the fitted GOP GBAR model were similar to the original video sequences but also showed some differences. The GOP-GBAR model generated frames with smaller sizes when compared with the actual traces and also under-estimated the packet loss in the queueing study. According to the authors, this is because the GOP GBAR model is stochastic and it actually represents, for any fixed set of model parameters, an ensemble of possible video sequences. This gives a range of statistic values.

6) The General AR Model: Zhang proposed a general AR model in [39]. This model generates gamma-distributed traffic with arbitrary correlation, model order and shape parameter. The model works in two steps. It first decomposes a given gamma sequence into a linear combination of a number of chi-square $(\chi^2(1))$ sequences and each chi-square sequence is then obtained by squaring a Gaussian process, which is efficiently generated by using an AR model from the given covariance matrix. Although the general AR model proves to be a viable modeling approach for video conference traffic in ATM networks, it is not suited to MPEG video sources [40]. Alhareish et al [40] proposed a Gaussian Autoregressive and Chi-Square (GACS) process for MPEG coded video traffic as an extension of the general AR model. The GACS model consists of two steps. First, it models the sizes of MPEG I, P, and B frames such that at a given point in a video sequence the size of I-frame would be bigger than that of a (neighboring) P-frame, which would be bigger than that of a (neighboring) B-frame. In the second step the model decomposes the gamma process of a given frame size into a weighted sum of a number of $\chi^2(1)$ sequences. Each $\chi^2(1)$ sequence is then obtained by squaring a Gaussian process, which is generated by using an AR model whose parameters are determined from an estimated covariance matrix. Results show that this model outperforms GBAR and nested AR models in terms of the autocovariance function and Q-Q plot. One limitation of this model is that the empirical video sequence must be drawn from a gamma distribution.

7) Projected AR models: In [41], Wu et al extended the AR model to a Projected AR (PAR) model for the purpose of fitting the frame-size histogram of MPEG video and also preserving the autocorrelation. The main concept of the PAR model is to project the data generated by the AR model to new data in such a way that its frame-size distribution is closer to that of the real video. To project the data requires calculating, sampling, and storing the frame-size histograms of the real traces. The PAR model uses the CDF as the projection function because the CDF statistically corresponds to the density function and it is very useful in the one-to-one projection. The empirical CDF of frame sizes can be obtained from the real video traces and the AR model. The PAR model produced sample data which was shown to match the frame-size histogram of the real video data. Casilari et al [42] proposed an extension of the PAR model to incorporate scene changes. To model scene changes, a Markov chain is used. For the GOP level, a modification for the PAR model is used to generate traffic within each scene. The model was used to imitate two MPEG-1 video sequences. The results show that it is able to accurately capture the behavior of the real traffic in a queue.

8) Summary: Autoregressive models (AR) have received a considerable amount of attention in the literature during the past twenty years. These video traffic models use different types of AR processes based on the type of video and encoding scheme. Table 3.1 summarizes the AR models with some of their attributes.

AR models, in general, appear to capture the autocorrelation behavior of compressed video, which is an essential element when modeling compressed video sources. The coefficients for these models are simple to estimate from empirical data. However, it is not possible to find a single AR model that can capture different statistical characteristics. Hence, there is no single video model that is suitable for all video sequences and all purposes. Similar observations were made in the survey papers on AR models [12] and [19].

IV. Models based on Markov Processes

In this section, we provide a review of Markovian and Markov-modulated models that have been used to model VBR compressed video. The Markov-modulated video traffic models, proposed in the literature, date back to the early 1990's. Like the AR models, they have also been very popular for modeling different type of videos and with different compression schemes.

A. Review of Markov Process Models

Models based on a Markov process use states to represent ranges of bit rates of a video sequence or ranges of frames or GOP sizes of a video sequence. A stochastic process $\{X_k\}$, $k = 1, 2, 3, \ldots$, with state space $S = \{1, 2, 3, \ldots\}$is Markovian if for every n and all states $i_1, i_2, \ldots$ where $i_n \epsilon$ S, it satisfies the Markov property,

$$\begin{aligned} P[X_n|X_{n-1} = i_{n-1}, X_{n-2} = i_{n-2}, ..., X_1 = i_2] \\ = P[X_n = i_n|X_{n-1} = i_{n-1}] \end{aligned} \quad (4)$$

In simple words, the current state of a Markov process depends only on its previous state, and not on any additional previous states. Markov processes and Markov chains are often used to modulate other processes such as Bernoulli, Poisson, gamma and AR. Each state of the Markov process represents a different set of parameters for the particular process. That is, while in a particular state, the model generates samples according to the set of parameters associated with that particular state. This is done for a period of time until the process switches to a different state, where it generates samples using a different set of parameters. Models of this type are referred to as Markov-modulated. Well-known examples of such models are the Markov-modulated Bernoulli process (MMBP) and the Markov-modulated Poisson process (MMPP).

The earlier models that employed Markov chains were for Differential Pulse Code Modulation (DPCM) type of traffic and were used for traffic studies in ATM networks. The states of these Markov chain models were usually set to represent different ranges of bits per frame or ATM cells per frame. For example, in a two-state Markov chain model proposed by Heyman [21], the states are: low rate (25 cells/frame) and peak rate (625 cells/frame). But this two-state model is not accurate enough and another Markov chain model was proposed in the same paper that used a larger number of states. The proposed Markov chain model is created as follows. Let X_n be the number of cells in frame n and Y_n be the integer part of $X_n/10$. The authors proposed to model $\{Y_n,\ n = 1, 2\ ,$ ... $,N\}$as a Markov chain with transition matrix $P = (\hat{p}_{ij})$, where,

$$\hat{p}_{ij} = \frac{\text{number of transitions from state } i \text{ to state } j}{\text{number of transitions out of state } i} \quad (5)$$

when the denominator is greater than zero. The smallest number of ATM cells per frame for their sample video sequence was 25 and the largest was 500. This model requires many parameters due to the transition probability matrix.

TABLE I
SUMMARY OF AR MODELS FOR VIDEO TRAFFIC

Model Type	Video Coding	Level	Scene Changes	Residual (Error)	Sources	Publication Date
AR(1) [20]	DPCM	Frame	No	Gaussian	Single	1989
AR(2) [21]	DPCM	Frame	No	Gaussian	Single	1992
DAR(1) [21]	DPCM	Frame	No	Negative Binomial	Multiple	1992
PAR [41]	MPEG	I/B/P Frames	No	Gaussian	Multiple	1995
DAR/MC [23]	DPCM	Frame	Yes	Negative Binomial	Multiple	1996
GAR [35], [36]	DPCM	Frame	No	Gamma	Single	1996
AR(2) [31]	H.261	I/B/P Frames	Yes	Log-normal	Single	1997
GBAR [37]	H.261	Frame	No	Gamma	Single	1997
PAR/MC [42]	MPEG	I/B/P Frames	Yes	Gaussian	Single	1998
General-AR [39]	DPCM	Frame	No	Gaussian	Single	1999
C-DAR [29]	-	Frame	No	Negative Binomial	Multiple	2000
GOP-GBAR [38]	MPEG	I/B/P Frames	No	Gamma	Single	2000
GACS [40]	MPEG	I/B/P Frames	No	Gaussian	Single	2004
Frame based AR(1) [32]	H.264	I/B/P Frames	Yes	Gaussian	Single	2009
DAR(1) [25]	H.264	I/B/P Frames	No	Pearson V	Multiple	2010
Nested AR [34]	MPEG	I/B/P Frames	Yes	Gaussian	Single and Multiple	2011

Most of the VBR video models that are based on Markov processes use Markov chains to modulate other processes. For example, in [43] the proposed Markov-modulated model uses the states of the Markov chain to represent scene-activity in videos containing scene changes. A scene is classified into three states: low, medium, and high activity. The classification is based on thresholds of the bit rate during a scene. These thresholds can be selected by inspection of the bit rate histogram of the actual video trace. Based on this scene classification, the scenes can be modeled by a three-state Markov chain, where each state represents a different scene type. The transition probability matrix of the Markov chain is calculated from the video trace. The bit generation during each state is modeled by an independent AR(1) process whose parameters depend on the current state of the Markov chain.

For these Markov chain models, the synthetic trace is generated by starting from an initial random state and then generating frame sizes according to the transition probability matrix till the required number of frames is generated.

B. Survey of Markov Process Models

In this section, we review and discuss the Markov process based models that have been proposed in the literature for different types of video traffic.

1) Simple Markov Chain Models: In [44], Rose proposed three models that generate GOP size sequences, where the number of different GOP sizes is represented by the number of states of the Markov chain. These models are: a histogram model consisting of a zero-order Markov chain, a simple Markov Chain model consisting of a lst-order Markov chain, and a scene-oriented model that uses several nested Markov chains. The first model lacks any GOP correlation information. The second model includes the correlations from one GOP to the next one but no correlations over larger lags. The third model consists of a Markov chain which controls the scene change process and a number of Markov chains which generate the GOP sequence within each scene class. After creating the transition probability matrix of the scene change process, the matrices for the GOP process of each scene class have to be computed. Within each scene class, the number of states of the Markov chain is determined by dividing the maximum GOP size in that scene by the standard deviation of the GOPs in that scene. As expected, the third model performs better than the other two.

In [45], Pancha and El Zarki proposed a frame and slice-layer Markov chain model for MPEG video. A slice is a single row of macro-blocks (16x16 pixels) in a video frame. The number of states in the Markov chain is determined by the ratio of the peak rate to the standard deviation of the ATM cell arrival rate. Thus, the states have step sizes equal to the standard deviation of the cell arrival rate. The range of each state is calculated starting from the mean. Transition matrices were given for different GOP sizes. The larger the GOP size, the more states required in the Markov chain. It was observed that the slice layer required more states than the frame layer.

2) A Markov Renewal Process Model (MRP): Lucantoni et al [46] proposed a single source video traffic model using a discrete-state, continuous-time Markov renewal process and compared it with the DAR model [21]. They partitioned the range of possible rates (i.e., bits per frame) into 40 equidistant levels, each represented by a different state of the Markov chain. The transition matrix P is estimated empirically. The geometric distribution and sometimes a mixture of two geometric distributions were fitted to represent the sojourn time at each state. The results show that this model performs better than DAR model.

3) Markov-modulated AR Models: Ramamurthy and Sengupta [47] proposed a hierarchal model which uses a Markov chain to capture the effects of scene changes. The model uses three stochastic processes. The first two processes use first-order AR to capture the short term and long term autocorrelation of the bit rate. The third is a Markov chain that is used to incorporate the extra bits generated during scene changes. The authors used a 3-state Markov chain because they observed that scene changes last for two frames. Furthermore, the first frame after a scene change has significant more bits than other frames. Therefore, most of the time the Markov chain stays in state 0 and no extra bits is added. If this state is left, it will take two frames to get back to state 0. During this period the bit rate is increased. The model was used to study the performance of a video multiplexer that multiplexes several full motion video sources. By varying the model parameters

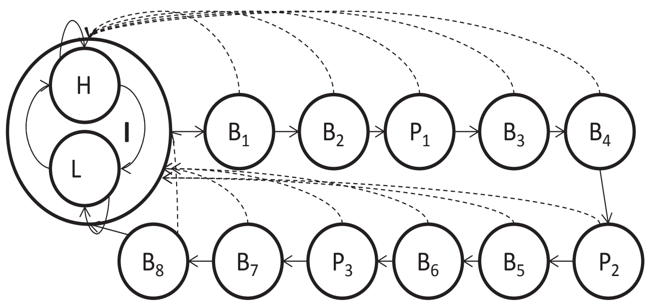

Fig. 2. Scene-based generalized Markov chain model [48]

the model was shown to represent a variety of videos from low to high bit rate with large movement and scene changes.

Yegenoglu et al [43] analyzed a full-motion video sequence of 500 frames encoded by the Discrete Cosine Transform (DCT), Differential Pulse Code Modulation (DPCM) and motion compensation. They proposed a first-order Gaussian AR process whose parameters are determined by the state of a Markov chain. The model works by classifying the frames into low, medium, and high activity scenes. The scenes are modeled by a three state Markov chain where each state represents the degree of motion activity. The model quantizes bit rates into N levels, where a quantization level loosely corresponds to a scene class. The transition probability matrix of the Markov chain is determined from the video trace. The bit generation during each state is modeled by an independent AR(1) process.

In [48], Chiruvolu et al proposed a similar model, i.e., a scene-based generalized Markov chain model for MPEG video traffic. The Markov chain consists of 12 states, as shown in figure 2, with each state corresponding to a frame within the GOP. The state for the I frame consists of two states, high (H) and low (L). A scene change may occur at any frame of a GOP. In this case, the process jumps back to the I frame state (dotted lines) and with some probability it starts from state H or state L. The Markov chain subsequently moves from state I to B_1, then to B_2, and so on until it gets back to state I where probabilistically it joins state H or L. At any state other than the two I frame states, it may move to the next state if there is no scene change or back to the I state if there is a scene change. The two state H and L correspond to high and low activity respectively in terms of bit rate. The I, P and B frames (bit generation) are modeled as independent AR(1) processes for low activity scenes. However, for high activity scenes, the cross-correlation with the I frames is taken into account by the AR(1) processes for the P and B frames. The P and B frames are generated based on the bit rate of the I frame of the current GOP.

Chen and Chen [49] modeled a video sequence of I and P frames. These I and P frames are classified in high, medium and low activity levels, each corresponding to different bit rates. The resulting Markov process consists of six states. The transitions between these states were obtained from the video trace. Within each state, frames are generated using an AR model that the authors referred to as the punctured AR process. According to [14], the algorithm presented in this paper has high computational complexity.

4) A Markov-modulated Gamma Model: In [16], Sarkar et al proposed two models for generating synthetic video traces. First, the video trace is partitioned into clips. A clip is a sequence of consecutive similar sized GOPs. For instance, if the following k *GOPs* $G_{i+1}, G_{i+2}, ..., G_{i+k}$ are in the same clip, starting from the ith *GOP*, then G_{i+k+1} belongs to the same clip if its size is less than the average clip size or equal to the average clip size plus a threshold. Subsequently, the clips are organized into shot classes. A shot class of length k is a union of k distinct but not necessarily consecutive clips. Each clip belongs to only one shot class. Shot classes are obtained by partitioning the entire range of GOP sizes into n sub-intervals. For this study, $n = 7$. Finally, each shot is sub-partitioned into three groups one per type of frame. That is, the first group contains all the I frames in the shot, the second group all the P frames and the third group all the B frames. A shifted gamma distribution was fitted to the frame sizes of each group. Thus, for the 7 shot classes there are 7 different gamma distributions for each frame type I, B and P. Subsequently a Markov chain was constructed consisting of 7 states, one for each shot class. The authors provided two methods to compute the transition matrix and call the resulting matrices P_A and P_B, respectively. The matrix P_A supports self-transitions but P_B excludes self-transitions.

The algorithms for computing P_A and P_B are quite similar. In both cases the transition probabilities were computed by calculating the transitions among shot classes as one sequentially traverses all GOPs in the original video. For P_A, $p_{ij} = f_{ij}/f_i$, where f_{ij} is the total number of transitions from shot S_i to S_j, and f_i is the total number of transitions out of S_i. The transition matrix P_B was computed in a similar manner except that all self-transitions are ignored. Next, two models, A and B, for the generation of video frame-size sequences were presented. Both models use Markov renewal processes. Model A uses only the matrix P_A for inter- and intra-state transitions. The synthetic trace was generated by starting from an initial random state and sampling frame sizes from the gamma distribution for the current state. After generating all frames of a GOP in the current state, the next state was chosen using the transition probability matrix. The process is repeated until the desired total number of frames was generated. Model B used gamma-distributed random variables for lengths of video segments and matrix P_B for inter-state transitions. A video segment is the maximum consecutive number of clips is a shot class. The length of a segment is the number of GOPs it contains. For model B, the number of GOPs generated while in a state is modeled by a gamma distribution of the segment length for this state. The parameters for these distributions for all states were estimated from the segment lengths of the original video. After the frames corresponding to a segment in a shot class are generated, the transition matrix P_B is used to determine the next shot class or state.

Only one GOP structure was used throughout this study and results for two full-length movies Crocodile Dundee and ET are reported. Q-Q plots were presented to show visually the similarity of the model generated VBR video data set with the original data set. Model A overestimated the sizes of the I, P and B frames and model B underestimated them. However, these deviations were very small and the frame-length distributions of synthetic and original videos are almost indistinguishable for both models. Next a leaky-bucket simulation was done to compare the data loss rate of original and synthetic traces. The data was passed through a generic buffer with capacity c and drain rate factor d. For this study, the buffer capacity was expressed in terms of the mean frame size of the VBR source and was independent of d. The results validated the model and revealed that even a low loss rate could affect a large fraction of I frames causing a significant degradation of the quality of transmitted video.

5) Other Markov-modulated Models: In [50], Lombardo et al used a Switched Batch Bernoulli Process (SBBP) to model MPEG video traffic. The model was defined in the discrete-time domain to make it more suitable for ATM networks, which are typically represented by discrete-time models due to the fixed size ATM packets requiring a fixed time to be transmitted out. A Switched Batch Bernoulli Process (SBBP) is another name for a Markov-modulated Bernoulli chain. It consists of N states, and in each state the inter-arrival time is geometrically distributed with a parameter that depends on the state of the Markov modulated chain. A GOP structure of IBBPBBPBBPBB was assumed and the distribution of the I, B, and P frames was approximated by a gamma function. The authors first obtained an SBBP for the I frames based on the frame size distribution and the autocorrelation function using a genetic algorithm. Subsequently, they obtain an SBBP for the entire trace by attaching a Markov chain to each state of SBBP for the I frames, whereby each state represents a B or P frame in a GOP. The states of the attached Markov chain are visited sequentially following the GOP pattern of B and P frames. In order to capture the intra-GOP correlation, the pdfs for the B and P frames are made dependent on the I frame belonging to the same GOP. The dependencies are obtained by fitting linear functions to the actual data for the mean and variance, from which a gamma distribution can be determined. The authors subsequently analyzed numerically a single finite capacity queue with an SBBP input and a constant service time. The video model was not validated against an actual trace.

In [51], Sun and Daigle proposed a scene-based Markov modulated model with feedback control of the frame size distribution. A state space of frame sizes is determined from the real trace and Markov transition matrices are estimated. When in a particular state, the frames sizes are generated according to a uniform distribution whose range is adjusted using some feedback. However, it is not clear from the paper how the scene changes are determined from the actual trace and modeled in order to generate the synthetic trace.

Zhao et al [52], [53] proposed a traffic model for slow motion MPEG-4 video which is based on a Markov modulated process with correlated batch arrivals. They considered traffic with two classes of arrivals, class 1 and 2. Class 1 represents critical data like I frames and class 2 represents less critical data like P frames. The video sequence considered had two types of frames, I and P. The frame size distribution of I frames was observed to be normal while for P frames it was lognormal. The authors modeled the video traffic as a discrete time batch Markov arrival process (DBMAP) with marked transitions. The underlying Markov chain of the video arrival process is given by $D = D0 + D1 + D2$, in which $D0$ corresponds to state transitions of the Markov chain without arrivals, $D1$ corresponds to state transitions with arrivals of a class 1 video frames, and $D2$ corresponds to state transitions with arrivals of a class 2 video frames.

The main purpose of the paper was to study the performance of transmitting MPEG-4 video over the uplink of an unreliable wireless channel. The authors showed that the transmission time of a burst over an arbitrary radio link control follows a Phase Type (PH) distribution, and consequently they modeled the uplink as a DBMAP/PH/1 priority queue. They presented a computational algorithm for the analysis of the queue and relevant numerical results. However, no results were given to validate the accuracy of the synthetic video generated by the DBMAP model.

C. Summary

A wide variety of Markovian models exist in the literature for different types of video and for different video coding standards. These Markovian models appear to be more accurate than the AR models. There are two main factors that distinguish these models from each other. First is the modulated process which is dependent on the frame size distribution of the video traffic. Authors have used many different types of frame size distributions, i.e., gamma, AR, Bernoulli, normal, lognormal, geometric and uniform. The second factor that differentiates these models is how the state space of the Markov process is determined and how the transition probabilities are calculated. As mentioned at the beginning of this section, some models use bit-rate or frame-size ranges for determining the number of states, while others use scene activity levels. In all cases the states are dependent on the bit-rate variation in the video traffic. However, the same transition probabilities cannot be used for all types of video. The transition probabilities are very much dependent on the type of video traffic. Table II summarizes the Markovian models along with some of their attributes.

V. Self-similar Models

In this section, we review the self-similar based video models that have been proposed in the literature.

A. Introduction

A process is said to be self-similar if the observations for that process appears 'similar' regardless of the duration of sampling interval. A self-similar phenomenon behaves the same when viewed at different degrees of magnification, or different scales on a dimension (space or time). Self-similar processes are long-range dependent. Long-range dependence (LRD) is the phenomenon where observations of an empirical record are significantly correlated to observations that are far removed in time. LRD is quantified by a single parameter, H, named after H. E. Hurst who studied long-term storage in water reservoirs [54]. H is related to the rate of decay β of the autocorrelation coefficients and to the parameter α that characterizes the power law behavior of the spectral density around the origin. Once H is estimated, a process such as

TABLE II
SUMMARY OF MARKOVIAN MODELS FOR VIDEO TRAFFIC

Model Type	Video Coding	Level	Scene Changes	Sources	Publication Date
Motion-classified AR/MC [43]	DPCM	Frame	No	Single	1993
MC [45]	MPEG	Frame/Slice	No	Single	1993
MRP [46]	DPCM	Frame	No	Single	1994
Markov-modulated AR [47]	DPCM	Frame	Yes	Multiple	1996
Simple MC [44]	MPEG	I/B/P Frames	Yes	Single	1997
Scene-based MC [48]	MPEG	I/B/P Frames	Yes	Single	1998
SBBP [50]	MPEG	I/B/P Frames	No	Single	1998
Markov-Modulated punctured AR [49]	H.263	I/B/P Frames	Yes	Single	2002
MMG [16]	H.264	I/B/P Frames	Yes	Single	2003
DBMAP [52], [53]	MPEG-4	I/P Frames	No	Single	2004
Scene-based Markov-Modulated [51]	MPEG-4	I/B/P Frames	Yes	Single	2005

fractional ARIMA, or Fast Fractional Gaussian Noise (*ff*Gn) is used to create a background sequence. This background sequence is then used to generate the foreground sequence using the desired empirical marginal bit-rate distribution.

B. A Survey of Self-similar Models for Video Traffic

In this section, we present the literature survey of self-similar video traffic models.

1) FARIMA models: Garrett and Willinger [55] analyzed a 2-hour long sequence of DCT encoded Star Wars movie and observed that the autocorrelation of the VBR video sequence decays hyperbolically (equivalent to long-range dependence) and can be modeled using self-similar processes. They proposed a Fractional Autoregressive Integrated Moving Average (FARIMA) model to replicate the LRD properties of the video sequence. The tail behavior of the marginal bit-rate distribution was described using a hybrid gamma/Pareto model. However, they did not provide an explicit model for the SRD structure of video traffic.

Huang et al [56] developed a model for an MPEG-1 encoded movie sequence, which contains I, B and P frames. They presented a unified approach that models the marginal distribution of empirical video records and also models directly both the short and the long-term empirical autocorrelation structures. The Hurst parameter was used to generate the LRD part of the autocorrelation function, and a weighted sum of exponentials was used to match the SRD part. The Hurst parameter was determined using a rescaled adjusted range statistic and variance-time analysis and the SRD parameters were determined using regression analysis. The results show that the autocorrelation function matched well for both short and long lags. However in [31], the authors claim that this model does not capture the multi-timescale variations observed in real traces.

Liu et al [57] proposed to model MPEG compressed video sequences by Markov modulated self-similar processes to capture the LRD characteristics of the video ACF. The MPEG video sequence is decomposed into three parts according to different scene complexity. Each part is described by a self-similar process. The FARIMA method is used to generate three self-similar processes. A beta distribution is used to characterize the marginal CDF of the video traffic. To model the whole data set, a Markov chain is used as a dominating process to govern the transitions among these three self-similar processes. The simulations on MPEG encoded Star Wars movie demonstrated that this model can capture the LRD of ACF and the marginal CDF of the frame sizes very well. Based on this model, the same authors proposed a simple composite model for modeling MPEG video consisting of I, B and P frames [58]. The only difference is that the video is decomposed according to the type of frames instead of scene activity. There are three self-similar processes for I, B and P frames. The results showed that it performed better than their previous model.

2) Discrete-time Statistically Self-similar System: Narasimha and Rao [59] proposed a Discrete-time Statistically Self-similar System (DTSS) for modeling variable-bit rate video traces. It was shown that with heavy-tailed inputs, it is possible to model both scene density time-series (number of frames/scene versus scene change index) autocorrelation as well as its marginal distribution with this DTSS model. The ACF of the synthetic trace generated by the DTSS was compared with that of the actual Star Wars movie trace and the ACF fits for LRD Model [55], M/G/∞ model and Markovian model. The performance of DTSS was very close to the M/G/∞ model [60].

C. Summary

The ACF structure of VBR video traffic exhibit both LRD and SRD properties. If the video is highly correlated for a large number of lags, then a model that captures LRD like self-similar process may be a good option. In this section we discussed the various LRD models that have been proposed in the literature. Table III summarizes the self-similar models with some of their attributes. The main drawback of self-similar models is the computational complexity. Also, these models fail to capture the SRD properties of video traffic.

VI. Wavelet-based Models

In this section we survey video traffic models based on wavelets.

A. Introduction

Recently, techniques using the wavelet transform have been used to model video traffic. Wavelet analysis is typically based on a decomposition of the signal using a family of basis functions. This includes a high-pass wavelet function that generates the detailed coefficients and a low-pass scaling filter which produces the approximation coefficients of

TABLE III
SUMMARY OF SELF-SIMILAR MODELS

Model Type	Video Coding	Level	Scene Changes	Sources	Publication Date
FARIMA [55]	DPCM	Frame	Yes	Single	1994
Unified-FARIMA [56]	MPEG	I/B/P Frames	Yes	Single	1995
Markov-Modulated FARIMA [57], [58]	MPEG	I/B/P Frames	Yes	Single	1999, 2002
DTSS [59]	MPEG	Frame	Yes	Single	2003

the original signal. The wavelet basis functions absorb the long-range and short-range dependencies by differencing the averages at all time scales, hence the wavelet coefficients are short-range dependent. This makes it possible to model wavelet coefficients as independent (or low-order Markov dependent) random variables without losing much information [61]. Wavelet models for generating synthetic traffic have two parts. First, the coefficients are obtained by applying the wavelet transform to the trace and parameters are estimated for the wavelet correlation model. In the second step, the coefficients are generated using the correlation model and an inverse wavelet transform is applied to get the synthetic trace.

B. Survey of Wavelet-based Video Traffic Models

We now proceed to review the relevant literature of wavelet-based models.

Wang et al [62] proposed a wavelet decomposition approach for modeling MPEG-1 broadcast video traffic. The proposed method decomposes video traffic into two parts via a wavelet transformation and models each part separately. The first part, which is modeled by an AR(1) process, captures the long-term trend of the traffic. The second part uses vector quantization to addresses the short-term behavior of the traffic. To generate the synthetic traffic, the authors specify a long-term traffic profile according to the derived AR(1) parameters, and a sequence of short-term local dynamics patterns according to the empirical transition probability of the Markov chain with 64 states that represent group of vectors obtained through vector quantization. Then, the inverse wavelet transform is performed to synthesize the traffic. The results show that this model performs better when compared with the DAR(1) [23] and a Markov chain model presented in [44].

Ma and Ji [61] investigated the independent wavelet model which is the simplest wavelet model. They showed that wavelets are capable of characterizing both long- and short-range dependent processes through variances of wavelet coefficients at different time scales. They also developed Markov wavelet models which capture the dependence among wavelet coefficients. A JPEG-coded Star Wars movie trace was used for this study. The results indicate that independent wavelet models are sufficiently accurate and Markov wavelet models only improve the performance marginally. The JPEG still image coding standard [63] is the most widely employed compression algorithm today for still color images. JPEG video coding uses a lossy compression algorithm based on DCT transform coding of image blocks of size 8x8. The transform coding is followed by quantization of the DCT coefficients and variable length coding [64].

Arifler and Evans [65] showed that self-similar scaling is present in video traffic and the compression ratio does not change this behavior. They have not presented a model to generate synthetic traces but suggested that a wavelet-based model can be used for this purpose.

Dai et al [14], [66] proposed a frame-based hybrid framework for modeling MPEG-4 and H.264 AVC and SVC video traffic. They used Haar wavelets to model the distribution of the I frame size and a simple time-domain model for the P and B frame sizes. The detailed coefficients are estimated using a mixture-Laplacian distribution while the coarsest approximation coefficients are modeled as dependent random variables with marginal gamma distribution. Using the estimated approximation and detailed coefficients, the inverse wavelet transform is performed to generate synthetic I frame sizes. It was noted that there is a strong correlation between the P/B frame sizes and the I frame size belonging to the same GOP, called intra-GOP correlation. A linear model that uses this intra-GOP correlation was proposed to generate the P and B frames. According to the authors, this model accurately captures both the SRD and LRD properties of video traffic. Simulation results showed that this model preserves the temporal burstiness and effectively captures the statistical features like the autocorrelation function and the frame-size distribution of the original traffic.

A comparative study between the GBAR model, a FARIMA-based model, a wavelet-based model and an empirical video trace (a JPEG coded Star Wars movie) was reported in [67]. The performance was measured in terms of goodness of fit for the second-order statistics (auto-correlation function and pdf of frame sizes). The statistical parameters of real data and source model data were compared to verify their similarity. The mean cell loss rate and mean buffer occupancy was also obtained by simulating a buffer with the synthetic trace obtained by each of the three models and also with the empirical video trace as input. It was shown that the FARIMA and wavelet domain methods are more appropriate for modeling the co-existence of SRD and LRD behavior in video traffic. The wavelet-based model performs best as a single source descriptor. It is also computationally efficient and analytically tractable compared to the self-similar FARIMA model.

C. Summary

It appears that wavelet-based models are good for modeling the co-existence of SRD and LRD behavior in video traffic. These results show that a key advantage of using wavelets is their ability to reduce the complex temporal dependence so significantly that the wavelet coefficients only possess the short-range dependence. However, the wavelet-based model requires understanding of digital signal processing tools and techniques. Also, it requires several trials in order to determine the optimal number of levels of decomposition for a particular type of video. A summary of the three wavelet based models discussed above is presented in table IV.

VII. Other Approaches

In this category we include other video models which do not appear to be part of a class of models, such as, the Seasonal ARIMA model [13], M/G/∞ process [60] and Transform-Expand-Sample (TES) based models [68]-[69].

A. The M/G/∞ Process

Krunz and Makowski [60] proposed a different approach for characterizing VBR video streams based on the $M/G/\infty$ process. The authors argued that the autocorrelation function $\rho(k)$, k =1,2,..., of a compressed-video sequence is better captured using the expression $\rho(k) = e^{\beta\sqrt{k}}$ than $\rho(k) = e^{\beta log k}$ associated with long-range dependences or $\rho(k) = e^{\beta k}$ associated with Markovian models. A video model with such a correlation structure can be constructed using the so-called $M/G/\infty$ input processes. In essence, the $M/G/\infty$ process is the stationary probability distribution of the number of busy servers in an $M/G/\infty$ queue. By varying the service time distribution G many forms of time dependence can be displayed, which makes the class of $M/G/\infty$ input models a good candidate for modeling many types of correlated traffic in computer networks.

For video traffic, they derived the appropriate G that gives the desired correlation function $\rho(k) = e^{\beta\sqrt{k}}$. Though not Markovian, this model is shown to exhibit short-range dependence. Poisson variates of the $M/G/\infty$ model were appropriately transformed to capture the marginal distribution of a video sequence. Using the performance of a real video stream as a reference, they studied via simulation the queueing performance under three video models: the proposed $M/G/\infty$ model, the fractional ARIMA model [55] which exhibits LRD, and the DAR(1) model which exhibits a Markovian structure. The results were obtained using JPEG and MPEG-2 I frame sequences. The results indicated that only the $M/G/\infty$ model is capable of consistently providing acceptable predictions of the actual queueing performance. Furthermore, only $O(n)$ computations are required to generate an $M/G/\infty$ trace of length n compared to $O(n^2)$ for an FARIMA trace.

B. The Seasonal ARIMA Model

Al Tamimi et al [13], [70], [71] have recently used a seasonal ARIMA model for modeling MPEG-4 part2, MPEG-4 AVC and SVC coding standards. ARIMA is a process in which auto-regression analysis, differencing, and moving average methods are used to fit time series data. ARIMA has three main parts: Autoregressive (AR), Integrated or differencing (I), and Moving Average (MA). ARIMA models are typically represented as ARIMA(p, d, q), where p is the order of the autoregressive part, d is the order of differencing part, and q is the order of moving average part. ARIMA models can be implemented using simple equations. For example, ARIMA(1,1,1) can be described as:

$$y(t) = w(t)+y(t-1)+f(y(t-1)-y(t-2))-qw(t-1) \quad (6)$$

where $w(t)$ is the error term at time t, f is the coefficient of the AR part and q is the coefficient of the MA part of the ARIMA model.

The seasonal ARIMA (SARIMA) [72] is an extension of the ARIMA model used for series that exhibit periodic or seasonal behavior. Seasonal ARIMA is described as ARIMA $(p, d, q) \times (P, D, Q)^s$. P, D, and Q represent the order of seasonal AR (SAR) part, the order of seasonal differencing part, and the order of the seasonal MA (SMA) part, respectively. s represents the seasonality of the series (e.g., the month seasonality in a year is 12). A statistical model to represent MPEG4-Part2 video traces using the SARIMA model class, called Simplified Seasonal ARIMA (SAM) was proposed in [13]. SAM views mobile video traffic as a time-series of frames clustered in GOPs. Using ARIMA representation, the simplified seasonal ARIMA or SAM can be written as

$$SAM = (1, 0, 1)(1, 1, 1)^s \quad (7)$$

This equation indicates that SAM has one autoregressive (AR) coefficient, no differencing, and one MA coefficient (1,0,1). The seasonal behavior includes one seasonal autoregressive coefficient, one seasonal differencing coefficient, and one seasonal moving average coefficient (1,1,1). In MPEG4-Part2, the seasonal period s is equal to the GOP size. An important difference between MPEG4-Part2 and AVC encoded videos is the multiple-frame reference feature in AVC. This feature results in the change of the seasonality period from s to $2\times s$. The authors used the public-domain statistical package R [73] to analyze the traces. They also implemented a SAM trace generator using R. Full trace frame size and ACF comparisons are provided in the paper. The authors claim that SAM allows easy adjustments of traffic parameters required for resource allocation studies.

C. Transform-Expand-Sample (TES) based Models

The TEStool is an interactive software environment for modeling auto-correlated time series using a class of stochastic processes called Transform-Expand-Sample (TES) [68]. TES processes are designed to fit simultaneously both the marginal distribution and the autocorrelation function of the empirical data. TES defines a method for generating an auxiliary background process, $\{U_n\}$, which allows one to vary the nature of dependence among the target random variables $\{X_n\}$. The process $\{X_n\}$, referred to as the foreground process, is generated from $\{U_n\}$using a suitable transformation.

In [74], Melamed et al developed a model for the number of bits per GOB for H.261-encoded video using the TES method. The H.261 pictures are divided into regions of 11 $\times$ 3 macroblocks, each of which is called a GOB. The model produced an autocorrelation which matched its empirical data counterpart up to a lag of 100 frames. The authors compared the TES model with a frame-oriented AR model and showed that TES is more accurate for modeling video traffic. Reininger et al [75] developed a frame-layer model using Composite-TES (C-TES) processes for MPEG sequences containing I, B and P frames. The scene change process was not taken into account in this model. The model requires nine parameters, three for each frame type.

TABLE IV
SUMMARY OF WAVELET-BASED MODEL FOR VIDEO TRAFFIC

Model Type	Video Coding	Level	Scene Changes	Sources	Publication Date
Wavelet/AR(1) [62]	MPEG	I/B/P Frames	Yes	Single	1997
Simple Wavelet [61]	JPEG	Frame	Yes	Single	2001
Hybrid Wavelet/Time domain [14], [66]	MPEG-4, H.264 AVC and SVC	I/B/P Frames	Yes	Single	2005, 2009

In another paper, Melamed and Pendarakis [76] presented a frame layer TES based model, that considers scene changes, for the Star Wars movie under a JPEG-like compression. Scene boundaries were detected by measuring the sustained absolute difference of the bit rates between a series of successive frames. A clustering technique was used to categorize the video sequence into four scene classes. A TES model was then created for each scene class, where each class was mapped to a state of a Markov chain, giving it the name of Markov Renewal Modulated TES (MRMT) process. This model produced better matches for the autocorrelation function for long lags, since the Markov chain captures the longer term scene change behavior. Two statistical models were developed using TES for MPEG video having two levels of priority in [77]. The first model is matched with the empirical frame size histogram and autocorrelation function of each frame type (I, P and B). The second model is created with the assumption that the number of bits in each frame type is gamma distributed.

Matrawy et al [69] also used the TEStool to model MPEG-4 VBR traffic. They modeled I, P and B frames using three different TES models and then used interleaving to generate the original sequence of frames for MPEG4. However, they have not specified which traces were used and how have they modeled the scene changes. The main drawback of all TES based models is that they require access to the TEStool and they have high computational complexity.

D. Summary

Table V summarizes the models discussed in this section along with some of their attributes.

VIII. AN OVERVIEW OF H.264 SVC, 3D AND HD VIDEO

In this section, we present an overview of the latest development in H.264 SVC, 3D and HD video. We also briefly discuss the traffic models that have been proposed to generate these types of video traffic.

A. H.264 SVC

Modern video transmission systems and networks support a wide range of connection qualities, network protocols and receiving devices, ranging from smart phones and tablets to high-definition televisions. The varying connection quality is due to the adaptive resource sharing mechanisms of the underlying network that responds to the varying data throughput requirements of a varying number of users. Scalable video coding is an attractive solution to the problems due to the distinctive characteristics of these transmission systems. A video bit stream is called scalable when parts of the stream can be removed in a way that the resulting sub-stream forms another valid bit stream for some target decoder. This sub-stream represents the source content with a reconstructed quality that is less than that of the complete original bit stream [10].

H.264 SVC supports layer-scalable coding. A layer-scalable encoding consists of a base layer and one or several enhancement layers identified by increasing layer identifiers. H.264 SVC provides three types of scalability, i.e., temporal scalability, spatial scalability, and quality (SNR) scalability [10]. A bit stream provides temporal scalability when the set of corresponding frames can be partitioned into a temporal base layer and one or more temporal enhancement layers with the following property. Let the temporal layers be identified by a temporal layer starting from 0 for the base layer. Then for temporal layer identifier k, the bit stream that is obtained by removing all frames of all temporal layers with a layer identifier T greater than k forms another valid bit stream for the given decode [10]. Spatial scalability provides different spatial frame resolutions. It provides a mechanism for reusing an encoded lower resolution version of an image sequence for the coding of a corresponding higher resolution sequence. Each spatial layer employs motion compensated prediction and intra-prediction [78]. Quality scalability can be considered as a special case of spatial scalability with identical picture sizes for base and enhancement layer. It is also referred to as coarse-grain quality scalable coding (CGS). H.264 SVC CGS employs same inter-layer prediction mechanisms as for spatial scalable coding, but without using the corresponding upsampling operations and the inter-layer deblocking for intra-coded reference layer macroblocks [10].

Various traffic models have been presented in the literature for modeling layer scalable video traffic. Dai et al presented a video traffic model for H.264 SVC video in [14] based on wavelets that exploits the cross-correlation between base and enhancement layers. It is quite similar to the wavelet model for H.264 AVC video discussed in section VI-B. Zhou et al used a two-state Markov chain to model the MPEG-4 2-layer spatial scalable video in [79]. In [80], the authors proposed a model based on the Markov arrival process (MAP) for modeling layered spatial scalable video data with no or few scene changes. Fiems et al [81] have proposed multi-class DBMAPs to characterize H.264/SVC scalable video traces at the sub-GoP level. They showed that a genetic algorithm yields Markov models with limited state space that accurately capture the characteristics of the video traces.

B. Three-Dimensional video

Three-dimensional (3D) video has become very popular during the last few years due to the advancement in display

TABLE V
SUMMARY OF OTHER VBR VIDEO TRAFFIC MODELS

Model Type	Video Coding	Level	Scene Changes	Sources	Publication Date
M/G/Infinity [60]	JPEG, MPEG-2	Frame	Yes	Single	1998
Seasonal ARIMA [13], [70], [71]	MPEG-4, H.264 AVC and SVC	I/B/P Frames	Yes	Single	2005, 2008, 2010
TES [74]	H.261	GOB	No	Single	1994
Composite-TES [75]	MPEG	I/B/P Frames	No	Single	1994
MRMT [76]	JPEG	Frame	Yes	Single	1994
TES [77]	MPEG	I/B/P Frames	Yes	Single	1995
Frame-based TES [69]	MPEG-4	I/B/P Frames	-	Single	2002

technology, signal processing, circuit design and networking infrastructure. A central issue in the storage and transmission of 3D content is the representation format and the compression technology. A number of factors must be considered in the selection of a distribution format. These factors include available storage capacity, bandwidth, player and receiver capabilities, backward compatibility, minimum acceptable quality, and provisioning for future services [82].

Multiview video, with two different views of a given scene, gives viewers the perception of depth. It is commonly referred to as three-dimensional (3D) video [83], [84]. Providing 3D video services over transport networks requires efficient video coding techniques and transport mechanisms to accommodate the large amount of video data from the two views especially on transmission links with limited bandwidth.

Vetro et al reviewed the 3D video representation and compression formats and their applicability in different storage and transmission systems in [82]. In [83], the authors analyzed the basic traffic and statistical multiplexing characteristics of 3D videos encoded with the three main representation formats: multiview (MV) representation, frame sequential (FS) representation and side-by-side (SBS) representation. With MV representation, each view has the same frame rate as the underlying temporal video format. The FS representation merges the two views to form a single sequence with twice the frame rate and applies conventional single-view encoding. Lastly, the SBS representation halves the horizontal resolution of the views and combines them to form a single frame sequence for single-view encoding. The results showed that the MV representation achieves the most efficient encoding, but generates high traffic variability, which makes statistical multiplexing more challenging.

Rossi et al provided a non-stationary Hidden Markov Model (HMM) for multi-view video in [85]. They have also provided a numerically stable version of the Expectation-Maximization (EM) algorithm for estimating the parameters of a non-stationary HMM.

C. High Definition Video

High-definition video or HD video is any video system of higher resolution than standard-definition (SD) video. The most commonly used display resolutions are 1,280×720 pixels (720p) or 1,920×1,080 pixels (1080i/1080p). HDTV and HD video streaming is very popular now a days. The difference with SD video is that HD takes more bandwidth and storage due to the large frame sizes. In [78], the authors have studied the video traffic characteristics of H.264 encoded HD videos. They have discussed how the different H.264 video encoding types affect video traffic and quality statistics. Al Tamimi et al proposed a seasonal ARIMA (SAM) model for HD videos encoded with H.264 AVC format in [86].

IX. Comparison of Four Video Traffic Models

A large number of video traffic models have been proposed in the literature in the last twenty years, focusing on different types of videos and encoding formats and using different modeling techniques. Attributes such as type of video and encoding standard are important in selecting an appropriate model. For example, for video conference, we can use a simple model like DAR(1) which has few parameters, but if the video is very dynamic having many scene changes then a more complex model like a Markov-modulated model with a larger number of parameters is more suitable. Also, if the video is MPEG with I, B and P frames, then we need a model with a separate process for each frame type, as they have quite different characteristics and marginal distributions of frame sizes. If the video is highly correlated for a large number of lags, than a model that captures LRD like a self-similar model or a wavelet-based model may be a good option. However, the drawback of self-similar models is their computational complexity. Similarly, models based on TES require the special TEStool software and they too have high computational complexity. A summary of the different categories of video traffic models is given in table VI.

The most common way to use the video traffic models in network performance evaluation is to generate traces and subsequently use them in simulation studies. It is technically feasible to include a video model in a simulator, but it would increase its execution time. However, these models cannot be used in a queueing model because they would increase its dimensionality significantly, thus rendering it impractical. One way that a video model can be used in a queueing model is to use it to produce a packet trace and then use queuing-theoretic solutions that use a trace as an input [87].

In this section, we present a comparison of four different models selected from the papers reviewed above. We applied these models to the most widely used encoding standard H.264 AVC. The video traces were downloaded from Arizona State University's video traces library [26], [27] (http://trace.eas.asu.edu/tracemain.html). We chose two models from the autoregressive group of models, a model from the Markov-based models and a model from the wavelet-based models. Specifically, we implemented and compared the following four models:

1) The Markov-modulated Gamma model (MMG) [16] discussed in section IV-B4

2) The Discrete Autoregressive model (DAR) [25] discussed in section III-B2
3) The Wavelet model [14], [66] discussed in section VI-B
4) The AR(2) model [30], [31] discussed in section III-B3

We chose these models because they are different from each other and also because they are based on MPEG-4/H.264 video with a GOP pattern that contains I, B and P frames. The first three models are fairly recent and are based on H.264 AVC. The DAR model is for video conference applications only and is not suitable for other applications like IPTV or video streaming with many scene changes. Therefore, we selected an additional model, i.e., the AR(2) model, from the category of autoregressive models that also incorporates scene changes.

Although the AR(2) model was developed for H.261 video, this is the closest to H.264 video as it has three types of frames and a similar GOP pattern. We applied it to the H.264 traces in order to compare it with the other models. We also chose the MMG model because it appears to be the best from the category of Markov-based models. It was developed for H.264 video coding standard and is the most recent model from this category. Moreover, it appears that the gamma distribution fits the frame size distribution of H.264 video better compared to other distributions such as normal and log-normal. This was also observed in [14], [66]. Lastly, the wavelet model that we chose, is also the most recent from the category of wavelet-based models. It models both inter-GOP and intra-GOP correlation in the VBR video and appears to perform better than the other models in this category. We did not implement any self-similar model because they capture the LRD properties of the video traffic but not the SRD properties [14], [17], [66]. We did not observe long-range dependencies in the autocorrelation function of the H.264 video traces. Also we did not implement a TES-based model because the TEStool is no longer available.

None of the survey papers [12], [17], [18], [19] presented earlier provide a comparison between these chosen four models. This is because the H.264 standard did not exist when the surveys were conducted and the models that are compared in this paper were not proposed yet. Two of these papers [17], [18] survey the models falling in the AR, Markovian, and self-similar categories. However, since these papers were published in 1999, they do not cover the VBR traffic models proposed in the last thirteen years. The survey paper by Al Heraish [12] reviews the AR models for video-conference type traffic only and it also does not cover many of the latest models. Similarly, in [19] the author surveys the AR models for full motion videos only. Lastly, the comparison results based on model implementation are not presented in [12], [17] and [19].

A. Model Implementation

In this section, we describe the implementation details of the four selected models.

1) Markov Modulated Gamma Model: We used the algorithm presented in [16] for the MMG model. First, a two-pass algorithm is used to partition the video into clips. A clip is a sequence of consecutive k similar sized GOPs. The clips are then organized into shot classes. A shot class of length k is a union of k distinct but not necessarily consecutive clips. A predefined number of shots n is used for generating the synthetic trace. We used $n = 7$ as it was the optimal number of shots based on the results in [16]. The GOP sizes are partitioned into these 7 shots. The successive partitioning boundaries of these shots increase in a geometric progression with a as the first term and $b = ar^n$, as the $(n+1)^{th}$ term where $r = e^{(\ln b - \ln a)/n}$, a and b are the GOP sizes corresponding to the 1 and 99 percentile points.

A transition probability matrix is formed for the transition probabilities between the different shot classes. These shot classes are the states of the underlying Markov chain. The transition probabilities are computed from normalized relative frequency of transitions among shot classes as one sequentially traverses all GOPs in the original video i.e.,

P_{ij} =Prob[The next GOP belongs to shot j the current GOP belongs to shot i]

All frames are partitioned into $3n$ data sets as each shot is sub-partitioned based on type of frame I, B or P. Each of the $3n$ data sets fits an axis-shifted Gamma distribution, whose parameters are estimated form the data set it models. Therefore, for 7 shot classes there are 21 gamma distributions for each frame type I, B and P. For the shift we ignore 1% of the data points (frame sizes) and set the value of the shift at the one percentile value.

For the generation of a synthetic trace, we start from any state randomly selected, and generate a GOP. The I, P and B frame sizes are sampled from a gamma distribution with their respective mean and variance. After generating all the frames in the GOP, we determine the next state using the state transition matrix. The process is repeated until the desired number of frames is generated. The computational complexity of this algorithm to generate a video trace of length N is $O(N)$.

2) DAR Model: A discrete autoregressive model of order p, denoted as DAR (p), generates a stationary sequence of discrete random variables with an arbitrary probability distribution and with an autocorrelation structure similar to that of an Autoregressive model. A DAR(1) process is a Markov chain with discrete state space S and a transition matrix:

$$P = \rho I + (1 - \rho)Q \tag{8}$$

where ρ is the lag-1 autocorrelation coefficient, I is the identity matrix and the Q matrix consist of the Pearson type V probabilities $\{f_0, f_1, \ldots, f_k, F_K\}$, where $F_k = \Sigma_{k>K} f_k$ and K is the peak rate [25]. Each k, for $k < K$, corresponds to possible source rates less than the peak rate of K.

In this model, the frame sizes are expressed in terms of the number of ATM cells/frame. We begin by estimating the minimum number of cells per frame, the maximum number of cells per frame, the mean and the variance for each type of frame, i.e., I, B and P, from the video trace. Using these and the pdf of the Pearson V distribution with parameters (α,β) given below, we obtain the rows of the Q matrix:

$$f(x) = \frac{x^{-(\alpha+1)} e^{-\beta/x}}{\beta^{-\alpha} \gamma(\alpha)} \tag{9}$$

where

$$\text{Mean} = \frac{\beta}{\alpha - 1} \tag{10}$$

TABLE VI
SUMMARY OF VIDEO TRAFFIC MODELS

Model Type	Video Coding	Level	Scene Changes	Sources	Strong Points	Limitations
Autoregressive models	DPCM, MPEG, H.261, H.264	I/B/P Frames	Mostly No, Yes with some models	Single/ Multiple	Simple to understand and implement	There is no single AR model that can model different statistical characteristics
Markov process based models	DPCM, MPEG, MPEG-4, H.263, H.264	I/B/P Frames	Mostly Yes	Single	Accurate compared to AR models, and can be used to model different types of video traffic	It is difficult to accurately define and segment video sources into different states in the time domain due to the dynamic nature of video traffic
Self-similar models	DPCM, MPEG	I/B/P Frames	Yes	Single	Accurately capture the LRD in video traffic	High computational complexity, fail to capture SRD in video traffic
Wavelet based models	MPEG, JPEG, H.264	I/B/P Frames	Yes	Single	Accurately model both SRD and LRD in video traffic	Difficult to implement, and determine how many levels of decomposition are needed
M/G/∞ process based model	JPEG, MPEG-2	Frame	Yes	Single	Shown to be more accurate than AR and self-similar models	Good for SRD traffic only
Seasonal ARIMA model	MPEG-4, H.264 AVC and SVC	I/B/P Frames	Yes	Single	Can be used to model different types of video traffic	-
TES based models	JPEG, MPEG, H.261, MPEG-4	GOB, I/B/P Frames	Yes/No	Single	TES method is non-parametric and can generate any marginal distribution or an arbitrarily close approximation	Require TES tool which is not publically available, high computational complexity

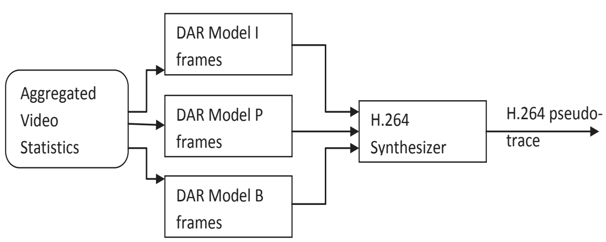

Fig. 3. Frame based DAR model [25]

and

$$\text{Variance} = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)} \tag{11}$$

The P matrix is then generated using the Q matrix and lag-1 autocorrelation coefficient ρ. From the transition matrix it is evident that if the current frame has i cells, then the next frame will have i cells with probability $\rho+(1\text{-}\rho)f_i$, and will have k cells, $k \neq i$, with probability $(1\text{-}\rho)f_k$.

The synthetic trace is generated by starting from a randomly selected state and generating frame sizes while traversing the transition probability matrix until the required number of frames is generated. The I, P and B frames are generated separately using their respective transition probability matrices and then multiplexed according to the required GOP format as shown in figure 3. The computational complexity of this algorithm to generate a video trace of length N is $O(N)$.

3) AR(2) Model: We used the frame-based AR model by Krunz and Tripathi [30], [31], described in section III-B3. It was shown that the frame size distribution of all three types of frames is lognormal while the scene length distribution can be approximated by a geometric distribution. We begin with a trace that contains a sequence of I, B and P frames, and separate all the frame types. Scene changes are determined using the I-frames only as follows. Let $Z_I(n)$, n =1,2,3... be the sequence of I frames. Suppose we are in the ith scene which started with kth I frame. The next I frame i.e. the $(n+k+1)$th frame belongs to $(i+1)$th scene if

$$\frac{|Z_I(n+k+1) - Z_I(n+k)|}{(\sum_{j=k}^{n+k} Z_I(j))/n} > T_1 \tag{12}$$

and

$$\frac{|Z_I(n+k+2) - Z_I(n+k)|}{(\sum_{j=k}^{n+k} Z_I(j))/n} > T_2 \tag{13}$$

Where T_1 and T_2 are two thresholds. We set T_1=0.05 and T_2=0.1, based on the suggestions in [31]. The scene length is modeled by a geometric distribution, whose parameters are estimated from the trace.

For generating the synthetic trace, we obtain the desired number of scenes from the original trace using the method described above. For each scene, we generate the number of I frames using the above geometric distribution. Then, we sample the first I frame size from a lognormal distribution, and subsequently the P and B frames are generated according to the GOP format, with frame sizes drawn from a lognormal distribution. The parameters for the three lognormal distributions are determined from the trace.

The next I frame size $X_I(i+1)$ for the current scene is obtained using the AR(2) process

$$X_I(i+1) = X_I(i) + \Delta_I(i) \tag{14}$$

where

$$\Delta_I(i) = a_1\Delta_I(i-1) + a_2\Delta_I(i-2) + \varepsilon(i) \tag{15}$$

and $\varepsilon(i)$ is a sequence of i.i.d. random variables sampled from normal distribution. a_1 and a_2 are estimated from the auto-correlation coefficients of Δ_I at lag 1 and lag 2. $\Delta_I(i)$ is the empirical sequence of differences between the average I frame size per scene and the actual I frame size for the entire trace. The computational complexity of this algorithm to generate a video trace of length N is $O(N)$.

4) Wavelet Model: As discussed in section VI-B, the wavelet analysis is based on a decomposition of the signal using a family of basis functions. This includes a high-pass wavelet function that generates the detailed coefficients and a low-pass scaling filter which produces the approximation coefficients of the original signal. It was observed in [14], that the detailed coefficients can be estimated using a mixture-Laplacian distribution. It was also noted that the approximation coefficients are non-negligibly correlated and are not i.i.d. To preserve the correlation of the approximation coefficients and achieve the expected distribution in the synthetic coefficients, the approximation coefficients are modeled as dependent random variables with a marginal Gamma distribution.

For implementing the wavelet based model, we used the Haar wavelet transform and the following algorithm proposed in [14]:

1) Generate the I-trace
   a) Perform J levels of decomposition on the original I trace
   b) For $i = 1$ to J
      i) Estimate the mixture-Laplacian parameters from the original detailed coefficients;
      ii) Generate synthetic detailed coefficients using the estimated parameters.
   c) At level J:
      i) Estimate the gamma distribution parameters from the original approximation coefficients;
      ii) Use copula [1] to generate correlated synthetic approximation coefficients.
2) Generate P-traces:
   a) Estimate the parameters of the generalized gamma distribution from the original residual process;
   b) Generate synthetic P-trace based on synthetic I-trace
3) Generate B traces: repeat step 2 using B frames.

The computational complexity of this algorithm to generate a video trace of length N is $O(N)$.

5) Performance Metrics: The video traces that we used are shown in table VII. The GOP size used is 16 with 3 B frames between I and P frames with the following pattern: IBBBPBBBPBBBPBBB. The quantization parameter (QP) for all three traces is 28. We picked these particular traces because we wanted to represent both video conference and IPTV. The NBC news sequence is closer to a video conference while the Star wars movie and Tokyo Olympics are similar to IPTV programs with many scene changes. We used the following performance metrics to validate and determine the accuracy of the models: a Q-Q plot of frame sizes and the frame size autocorrelation function. Both of these metrics are used in several papers discussed earlier such as [14] and [16].

[1]Copulas are functions that describe dependencies among variables, and provide a way to create distributions to model correlated multivariate data [88]

TABLE VII
VIDEO TRACES USED FOR COMPARISON

Movie	Frame Rate	Total number of frames	Movie length
Star Wars IV [G16,B3,QP28]	30 frames/sec	54,000	30 min
Tokyo Olympics [G16,B3,QP28]	30 frames/sec	133,128	60 min
NBC 12 News [G16,B3,QP28]	30 frames/sec	49,523	30 min

A Q-Q plot (Q stands for quantile) is a graphical method for comparing two probability distributions by plotting their quantiles against each other [15]. For instance, let us assume that we have two sets of observations for two random variables X and Y. Then, for a given percentile we calculate the corresponding percentile values of X and Y, i.e., P[$X \leq \gamma$] $= x$ and P[$Y \leq \gamma$]$= y$. The Q-Q plot is a graph of set of pairs (x, y) for different values of γ. If the two data sets are identical, the Q-Q plot is a straight line, that is, $x = y$ for all γ values. Otherwise, the closer to the line, the better the match between the two random variables X and Y.

Q-Q plots can be used to compare collections of data, or theoretical distributions. A Q-Q plot is a better approach than comparing histograms of the two samples, but requires more skill to interpret. Q-Q plots are commonly used to compare a data set to a theoretical model. A Q-Q plot depicts global similarity of two datasets [16]. However, it does not reveal any information about temporal ordering and burstiness of the frames. For example, one dataset may have all the large data values together and another dataset may have these large and small data values interleaved, and yet both may show identical Q-Q plots.

In the literature, the temporal ordering and burstiness of the frames sizes generated by a model is validated by using the generated trace and the original trace in a single server queue, so that to observe and compare the loss rate. This type of validation was influenced by studies in the 90s where the video was transmitted over an ATM network. The main performance criterion for congestion control and provision in ATM networks was the cell loss rate. However, currently, video is transmitted over the IP network, and in addition to packet loss, the one-way end-to-end delay and jitter are also important QoS metrics. Consequently, in order to study the temporal ordering of frame sizes, we believe that this should be done within the context of a tandem queueing network depicting the path of an end-to-end video flow with a view to measuring the above three QoS metrics. We hope to report our results on the temporal ordering of the four models compared in this paper in a follow-up paper.

B. Q-Q Plots for the Four Models

In figures 4 to 6 we present the Q-Q plots of the frame sizes (in bits) predicted by the Markov-modulated Gamma (MMG) model for the three traces. We can see that the Q-Q plot for the

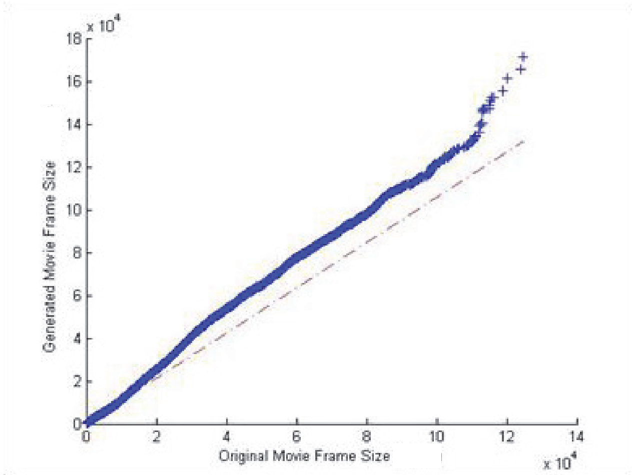

Fig. 4. MMG Model, Star Wars Trace

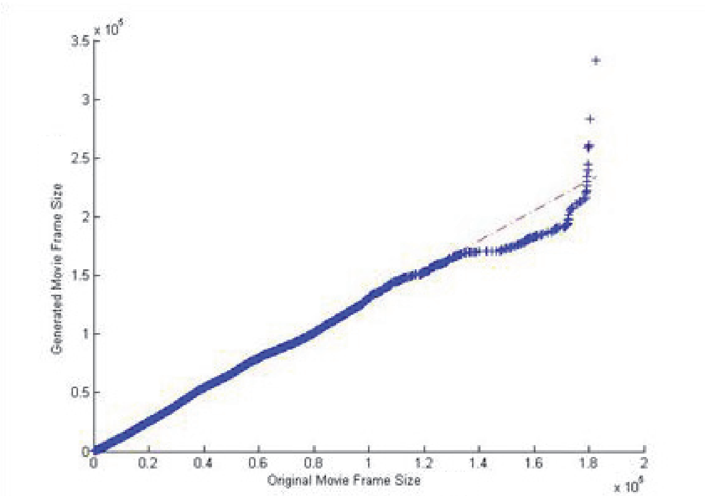

Fig. 5. MMG Model, NBC News Trace

NBC News is the best; this is because of the smaller variation in frame sizes as compared to the other two video sequences. For the other two traces, this model over-estimates the frame sizes.

Next, in figures 7 to 9 we present the Q-Q plots predicted by the Discrete Autoregressive (DAR) model when applied to the three different traces. We can see that the DAR model performs better for the NBC News trace as it is similar to a videoconference. This is because the authors proposed this model for the videoconferences only. In figures 10 to 12 the Q-Q plots for the AR(2) model are given.

Lastly, in figures 13 to 15, we present the results for the Wavelet based model. For the generation of I frames using the wavelet model, we used four levels of decomposition.

From the Q-Q plots, it is quite evident that the MMG model and the wavelet models perform better than the other two on all the three types of videos. However, both MMG and wavelet models tend to overestimate the larger frame sizes. The DAR model is good for the NBC News trace only. In [25] the authors mention that DAR model performs better for multiplexed video traces. Since the other three models are

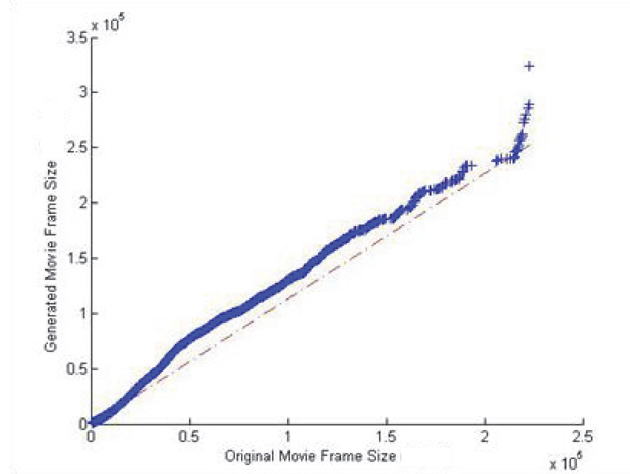

Fig. 6. MMG Model, Tokyo Olympics Trace

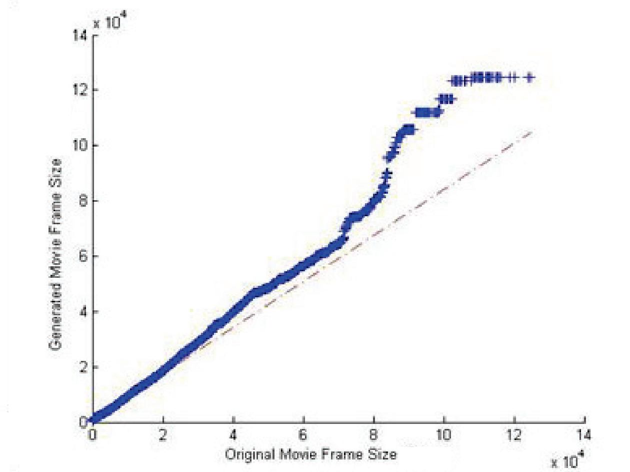

Fig. 7. DAR Model, Star Wars Trace

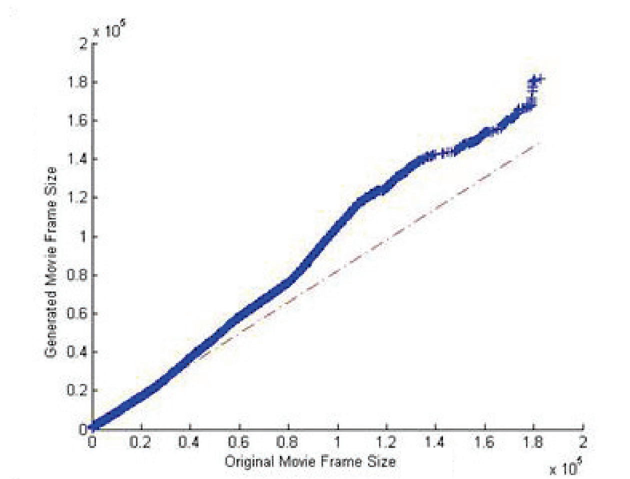

Fig. 8. DAR Model, NBC News Trace

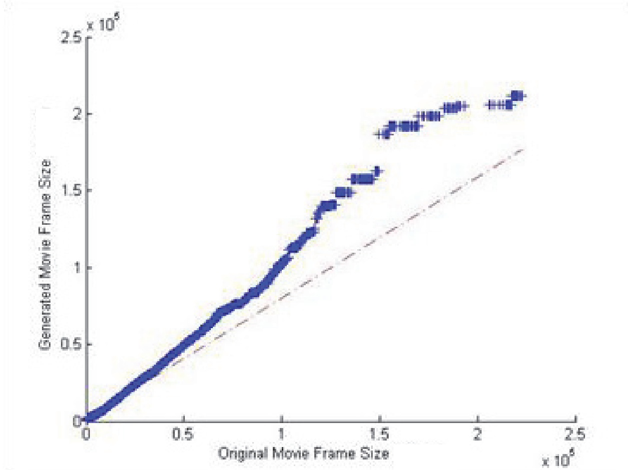

Fig. 9. DAR Model, Tokyo Olympics Trace

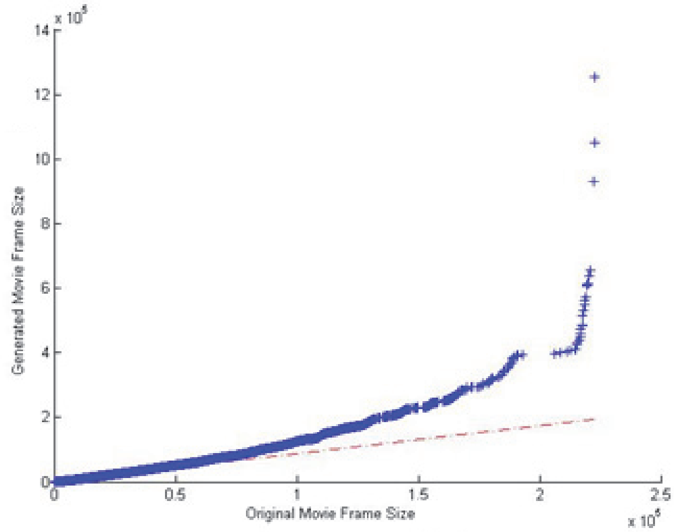

Fig. 12. AR2 Model, Tokyo Olympics Trace

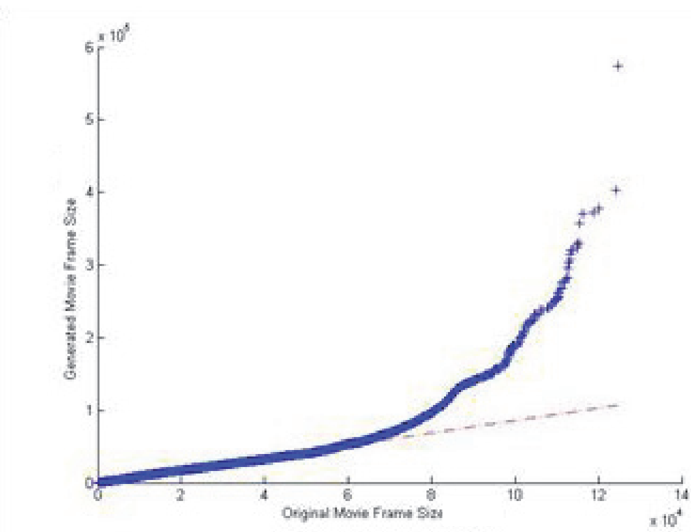

Fig. 10. AR2 Model, Star Wars Trace

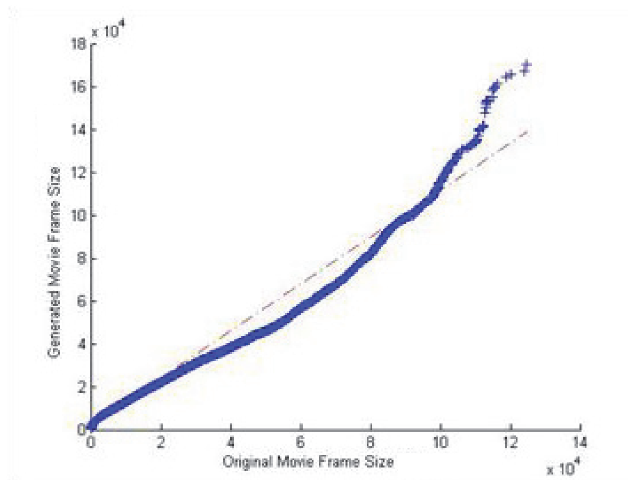

Fig. 13. Wavelet Model, Star Wars Trace

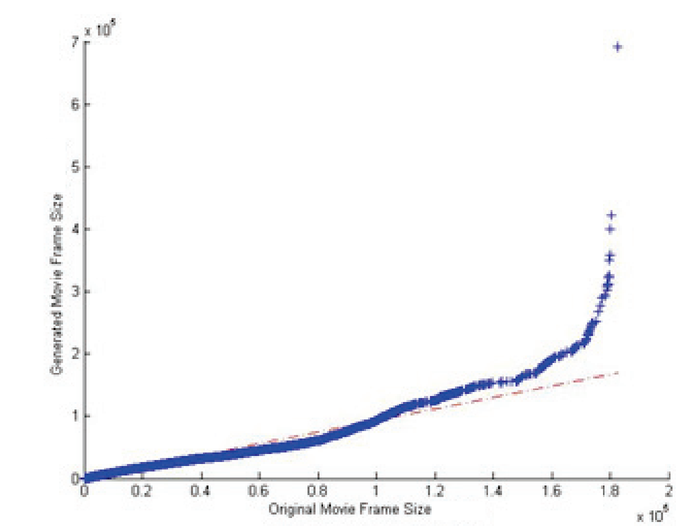

Fig. 11. AR2 Model, NBC News Trace

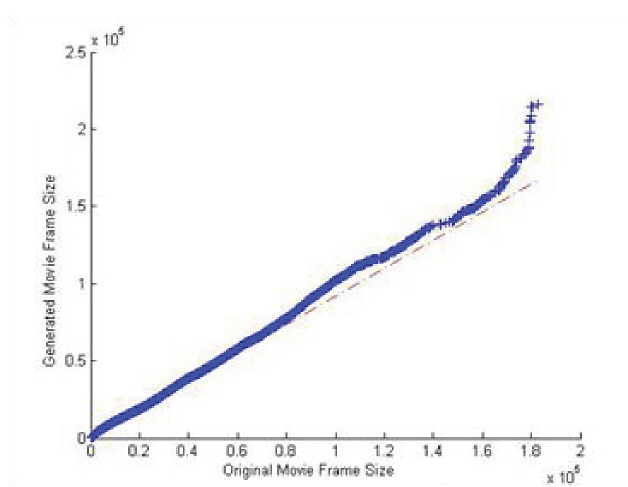

Fig. 14. Wavelet Model, NBC News Trace

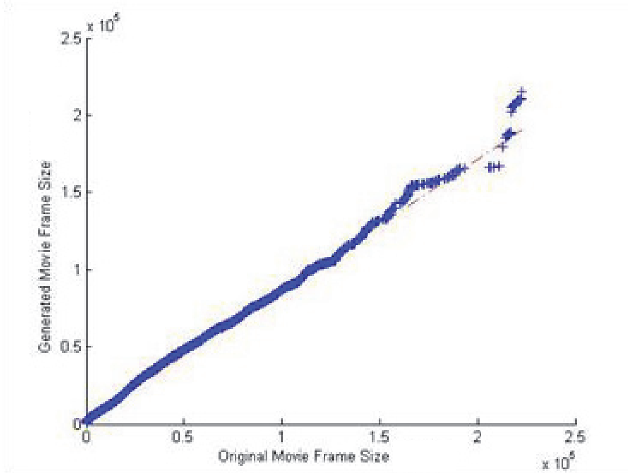

Fig. 15. Wavelet Model, Tokyo Olympics Trace

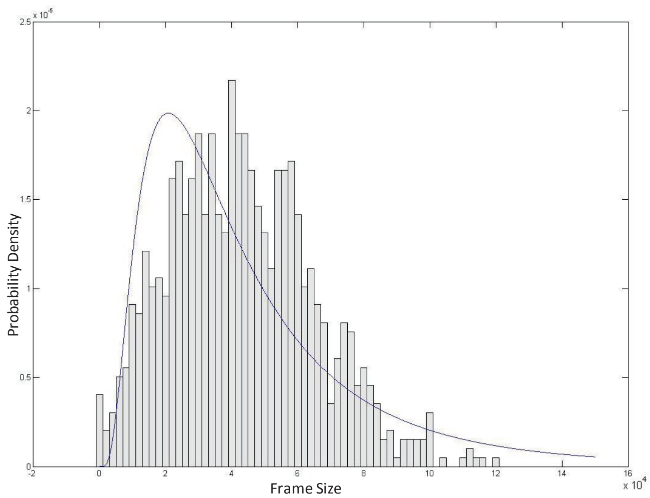

Fig. 16. Lognormal fit for I frames, AR(2) Model

for single source, the comparison results are provided here for single trace. The Q-Q plots for the AR(2) model suggest that it performs the worst in generating the synthetic traffic as it highly over-estimates the frame sizes for all traces. This is because the AR(2) model samples randomly the size of I frames from the lognormal distribution without considering the fact that some scenes have larger I frames than others. In contrast the MMG and DAR models keep track of how the sizes of I frames vary because they use the probability transition matrices. Also the lognormal fit to the frame size distribution is not as accurate for the H.264 traces compared to the H.261 traces. The lognormal distribution fit to the I (averaged over scenes), P and B frames are shown in figures 16 to 18 for the AR(2) model.

C. Frame Size Autocorrelation Function

The ACF of the frame sizes for all the empirical traces is a "comb of spikes" superimposed on a slowly decaying curve [26]. The existence of strong autocorrelation coefficients is due to the periodic recurrence of I, B and P frames. The larger

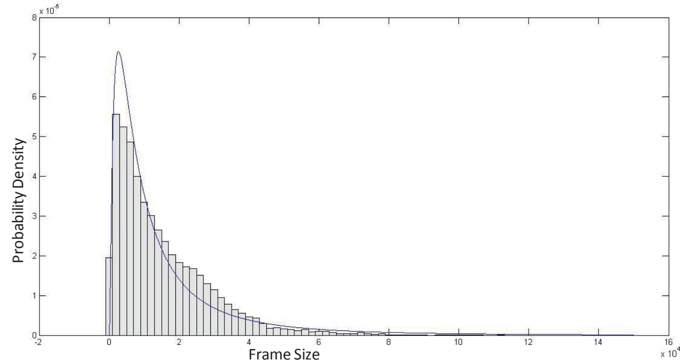

Fig. 17. Lognormal fit for P frames, AR(2) Model

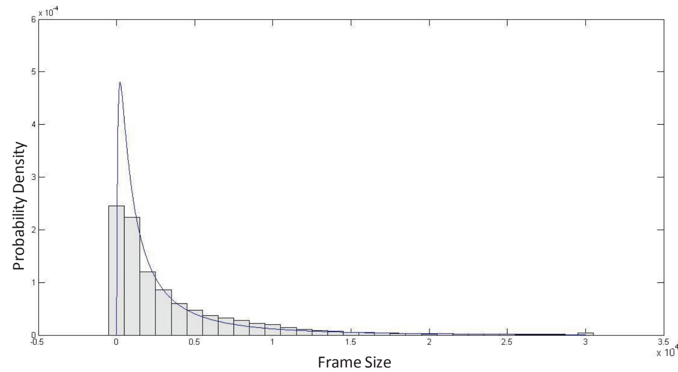

Fig. 18. Lognormal fit for B frames, AR(2) Model

peaks occur for lags that are multiples of 16, i.e., the I frame period, and it is a result of the correlation of the large I frames with each other. The three smaller peaks in between the larger peaks are the result of the correlation between the I and the P frames. For other lag values, the I or P frames are correlated with the B frames but the autocorrelation is relatively small.

In figures 19 to 22, we plot the autocorrelation coefficient of the frame sizes as a function of the lag in frames for the four models and compare it with the auto-correlation of the frame sizes in the actual traces. The results presented here are for the Star Wars movie trace. Rest of the results for the other traces are presented in Appendix A.

The autocorrelation plot for all the three models is similar in shape to the actual trace because of the periodic nature of the GOP pattern. It is obvious from the figures that the MMG model and the wavelet model closely follow the ACF of the actual traces compared to the DAR and AR(2) models. Again, we can see that the AR(2) model performs the worst. The spikes for the correlation between I frames are much smaller than the actual trace. As mentioned earlier, this is because the AR(2) model samples randomly the size of I frames from the lognormal distribution without considering the fact that some scenes have larger I frames than other.

X. Conclusion

Accurate video traffic models are necessary for evaluating the performance of a new network design, test the performance of an existing network, and evaluate call admission

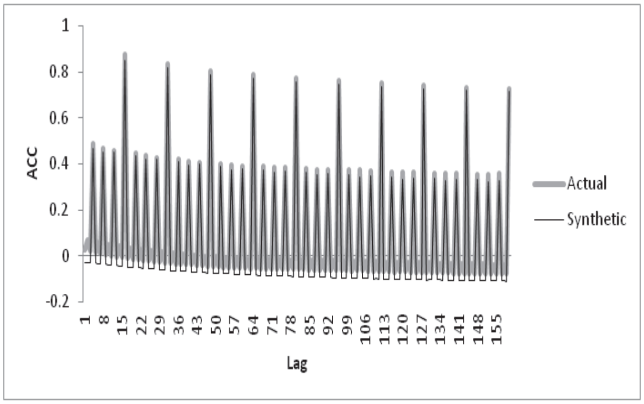

Fig. 19. MMG Model, Star Wars Trace

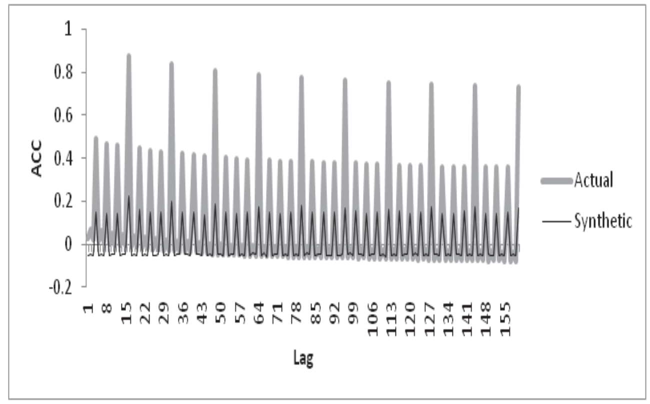

Fig. 21. AR2 Model, Star Wars Trace

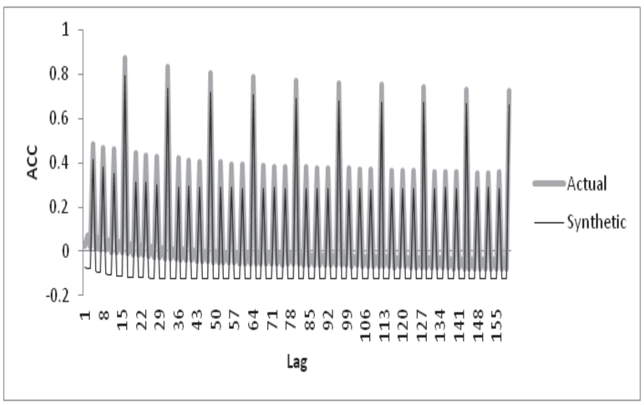

Fig. 20. DAR Model, Star Wars Trace

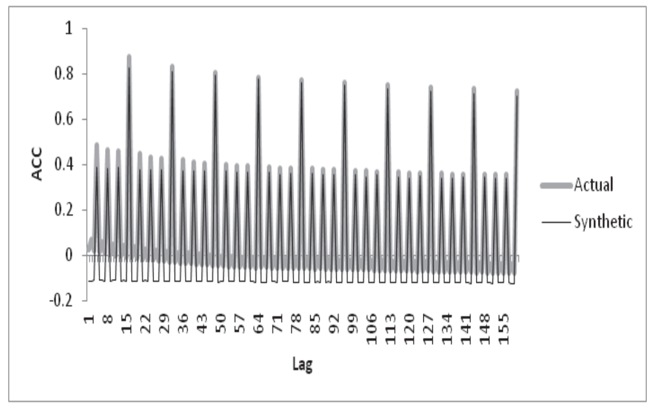

Fig. 22. Wavelet Model, Star Wars Trace

control and bandwidth allocation schemes for video streams. In this paper, we surveyed the VBR video traffic models proposed over the last two decades for different type of videos and encoding formats. We classified these models into the following five categories based on the methodology used: AR models, Markov process/chain models, self-similar and fractional ARIMA models, wavelet models, and other. Within each class, we surveyed the relevant video models and summarized their salient features in a table. AR models, in general, appear to capture the autocorrelation behavior of compressed video, and it is easy to estimate their coefficients from empirical data. There is no single AR model that is suitable for all video sequences and all purposes. Markovian models are more complex than AR models but they appear to be more accurate and have a wider applicability. Self-similar models are good for a video sequence which has a high correlation for a large number of lags. However, they are computationally complex and they do not capture the SRD properties of video traffic. Wavelet-based models are good for modeling both the SRD and LRD behavior in video traffic. A key advantage of using wavelets is their ability to reduce the complex temporal dependence so significantly that the wavelet coefficients only possess the short-range dependence. Finally, TES is a good modeling tool for video traffic, but it is computationally complex and it requires access to the TEStool.

We also implemented and compared four representative models on three different traces representing different types of videos. From the results, we can conclude that the MMG model and the wavelet model are good for both types of videos, i.e., IPTV and videoconference. The frame sizes sequences generated by both these models are very close to the actual traces. However, in the MMG model it is difficult to segment accurately the video sources into different states of a Markov chain due to the dynamic nature of video traffic. Wavelet model requires several trials in order to determine the optimal number of levels of decomposition for a particular type of video. We also conclude that the Discrete Autoregressive (DAR) model has an acceptable performance for videoconference type videos only. The AR(2) model generates videos which are not close to any of the actual video sequence. This is due to the fact that this model was developed for H.261 video and it uses a log-normal distribution fit for frame sizes. A gamma distribution would be a better fit for the frame size distribution and might improve the performance of this model.

In this paper, we have also provided a brief overview of H.264 SVC, 3D and HD videos. Few traffic models have been proposed for SVC video. SVC is quite different from AVC because it has multiple layers of video traffic and these layers are cross-correlated. A good model needs to capture this cross-correlation between the multiple layers of SVC video. Similarly, only one model has been proposed in the literature

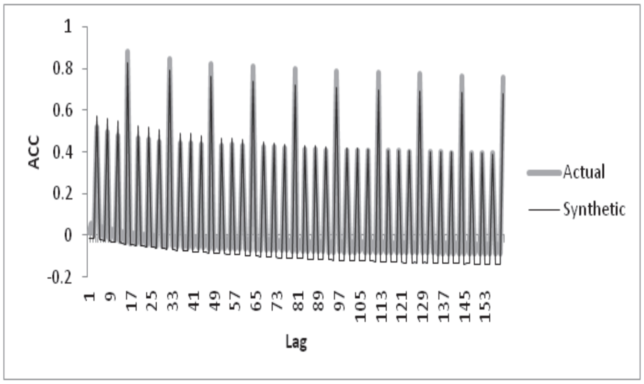

Fig. 23. MMG Model, NBC News Trace

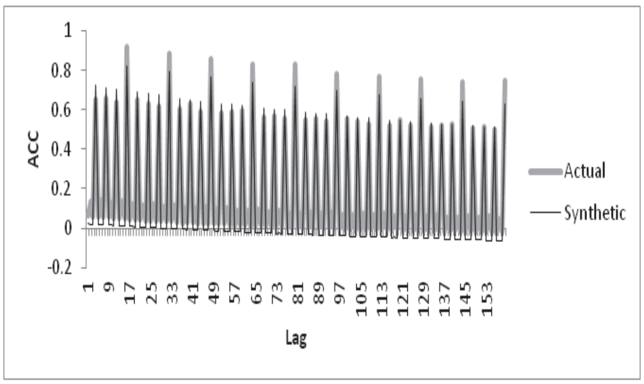

Fig. 24. MMG Model, Tokyo Olympics Trace

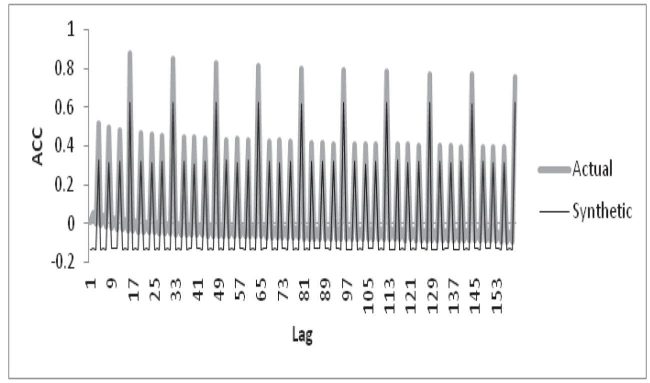

Fig. 25. DAR Model, NBC News Trace

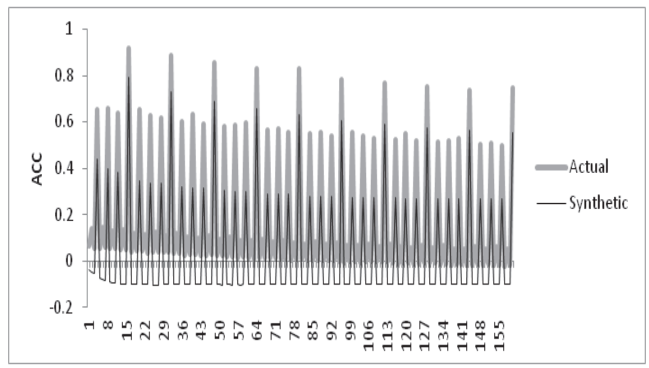

Fig. 26. DAR Model, Tokyo Olympics Trace

for modeling HD video traffic. Another important emerging type of video is 3D video. This type of video typically involves the transmission of two views, i.e., a left view and a right view, for each video frame. Multi-View Coding (MVC), i.e., the encoding of the left and right views, has attracted significant interest in the video compression research community. For future work, SVC, HD, and 3D video models need to be further investigated and compared.

APPENDIX A
FRAME SIZE AUTOCORRELATION FUNCTION

In figures 23 to 30, we plot the autocorrelation coefficient of the frame sizes as a function of the lag in frames for the four models and compare it with the auto-correlation of the frame sizes in the actual traces.

REFERENCES

[1] Cisco telepresence. [Online]. Available: http://www.cisco.com/web/telepresence/index.html

[2] Cisco webex. [Online]. Available: http://www.webex.com/

[3] I. R. Richardson, *The H.264 advanced video compression standard*, 2nd ed. Wiley, 2010.

[4] J. Hwang, *Multimedia networking from theory to practice*. Cambridge, 2009.

[5] K. Rijkse, "H.261 : Video coding for low-bit-rate communication," *IEEE Commun. Mag.*, vol. 34, no. 12, pp. 42–45, December 1996.

[6] "H.263 : Video codec for audiovisual services at p x 64 kbit/s," ITU-T Recommendation, 1993.

[7] J. Watkinson, *The MPEG Handbook: MPEG-1, MPEG-2, MPEG-4*, 2nd ed. Elsevier, 2004.

[8] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, p. 560576, July 2003.

[9] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T.Wedi, "Video coding with H.264/AVC: Tools, performance and complexity," *IEEE Circuits Syst. Mag.*, vol. 4, no. 1, pp. 7–28, 2004.

[10] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, September 2007.

[11] G. V. der Auwera, P. T. David, and M. Reisslein, "Traffic and quality characterization of single-layer video streams encoded with H.264/AVC advanced video coding standard and scalable video coding extension," *IEEE Trans. Broadcast.*, vol. 54, no. 3, pp. 698–718, September 2008.

[12] A. A. Alheraish, "Autoregressive video conference models," *Int. J. Network Management*, vol. 14, no. 5, pp. 329–337, 2004.

[13] A. Tamimi, R. Jain, and C. So-In, "SAM: A simplified Seasonal ARIMA model for mobile video over wireless broadband networks," in *Tenth IEEE International Symposium on Multimedia (ISM2008)*, Berkeley, California, USA, December 2008, pp. 178–183.

[14] M. Dai, Y. Zhang, and D. Loguinov, "A unified traffic model for MPEG-4 and H.264 video traces," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 1010 – 1023, August 2009.

[15] Online statistics education: An interactive multimedia course of study. developed by Rice University and University of Houston Clear Lake and Tufts University. [Online]. Available: http://onlinestatbook.com/index.html

[16] U. K. Sarkar, S. Ramakrishnan, and D. Sarkar, "Modeling full-length video using Markov-modulated Gamma-based framework," *IEEE/ACM Trans. Netw.*, vol. 11, no. 4, pp. 638–649, 2003.

[17] M. R. Izquierdo and D. S. Reeves, "A survey of statistical source models for variable-bit-rate compressed video," *Multimedia Systems*, vol. 7, no. 3, pp. 199–213, 1999.

[18] E. Casilari, M. Reyes, A. D. Estrella, and F. Sandoval, "Classification and comparison of modelling strategies for VBR video traffic," in *ITC-16*, 1999, p. 817826.

[19] A. Alheraish, "A comparison of AR full motion video traffic models

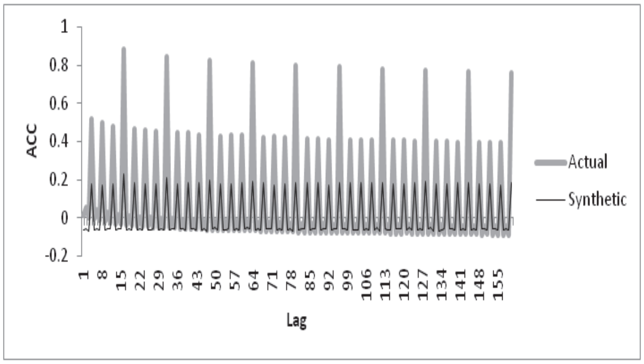

Fig. 27. AR2 Model, NBC News Trace

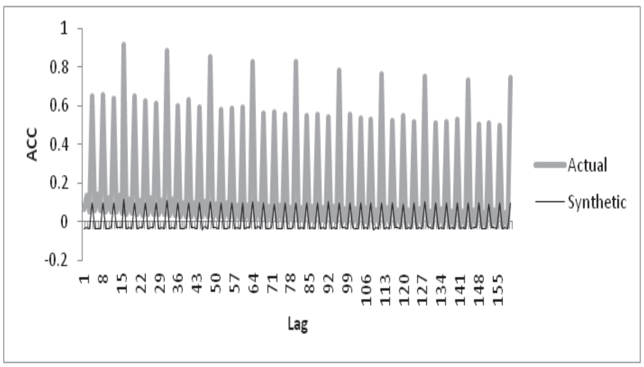

Fig. 28. AR2 Model, Tokyo Olympics Trace

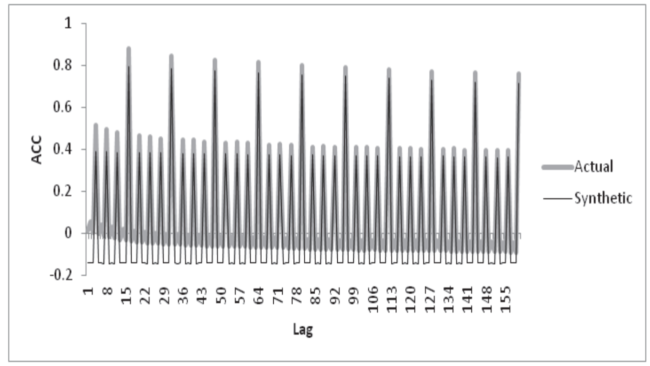

Fig. 29. Wavelet Model, NBC News Trace

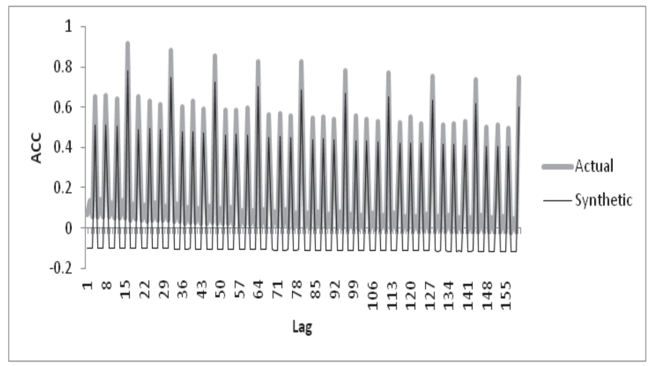

Fig. 30. Wavelet Model, Tokyo Olympics Trace

in B-ISDN," *Computers and Electrical Engineering Journal*, vol. 31, no. 1, pp. 1–22, January 2005.

[20] M. Nomura, T. Fujii, and N. Ohta, "Basic characteristics of variable rate video coding in ATM environment," *IEEE J. Sel. Areas Commun.*, vol. 7, no. 5, pp. 752–760, 1989.

[21] D. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 3, pp. 49–59, 1992.

[22] P. A. Jacobs and P. A. W. Lewis, "Time series generated by mixtures," *J. Time Series Analysis*, vol. 4, no. 1, pp. 19–36, 1983.

[23] D. P. Heyman and T. V. Lakshman, "Source models for VBR broadcast-video traffic," *IEEE/ACM Trans. Netw. (TON)*, vol. 4, no. 1, pp. 40–48, February 1996.

[24] J. S. Chitode, *Digital Communications*. Technical Publications, 2009.

[25] A. Lazaris and P. Koutsakis, "Modeling multiplexed traffic from H.264/AVC videoconference streams," *Computer Communications*, vol. 33, no. 10, pp. 1235–1242, 2010.

[26] G. V. der Auwera, P. T. David, and M. Reisslein, "Traffic and quality characterization of single-layer video streams encoded with H.264/MPEG-4 advanced video coding standard and scalable video coding extension," *IEEE Trans. Broadcast.*, vol. 54, no. 3, pp. 698–718, September 2008.

[27] P. Seeling, M. Reisslein, and B. Kulapala, "Network performance evaluation with frame size and quality traces of single-layer and two-layer video: A tutorial," *IEEE Commun. Surveys Tutorials*, vol. 6, no. 3, pp. 58–78, 2004.

[28] E. S. Pearson, *The Selected Papers of E. S. Pearson*. University of California press, 1966.

[29] S. Xu, Z. Huang, and Y. Yao, "An analytically tractable model for video conference traffic," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 2, pp. 63–67, February 2000.

[30] M. Krunz and H. Hughes, "A traffic model for MPEG-coded VBR streams," in *ACM SIGMETRICS*, 1995, pp. 47–55.

[31] M. Krunz and S. K. Tripathi, "On the characterization of VBR MPEG streams," in *SIGMETRICS*, 1997, pp. 192–202.

[32] H. Koumaras, C. Skianis, and A. Kourtis, "Analysis and modeling of H.264 unconstrained VBR video traffic," *International Journal of Mobile Computing and Multimedia Communication, IGI-Global*, vol. 1, no. 4, pp. 4–31, July 2009.

[33] H. Koumaras, C. Skianis, G. Gardikis, and A. Kourtis, "Analysis of h.264 video encoded traffic," in *INC 2005 Fifth International Network Conference*, Samos Island, Greece, July 2005.

[34] D. Liu, E. Sara, and W. Sun, "Nested autoregressive processes for MPEG encoded video traffic modeling," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 2, pp. 169–183, February 2001.

[35] S. Xu and Z. Huang, "A novel VBR video model on ATM networks," in *International Conference on Communications Technology*, vol. 2, Beijing, China, May 1996, pp. 1045–1048.

[36] S. Xu and Z. Huang, "A gamma autoregressive video model on ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 4, pp. 138–142, 1998.

[37] D. P. Heyman, "The GBAR source model for VBR videoconferences," *IEEE/ACM Trans. Netw. (TON)*, vol. 5, no. 4, pp. 554–560, August 1997.

[38] M. Frey and Nguyen-Quang, "A gamma-based framework for modeling variable bit rate MPEG video sources: the GOP GBAR model," *IEEE/ACM Trans. Netw.*, vol. 8, no. 6, pp. 710–719, December 2000.

[39] Q. T. Zhang, "A general AR-based technique for the generation of arbitrary Gamma VBR video traffic in ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 7, pp. 1130 – 1137, 1999.

[40] A. Alheraish, S. A. Alshebeili, and T. Alamri, "A GACS modeling approach for MPEG broadcast video," *IEEE Trans. Broadcast.*, vol. 50, no. 2, pp. 132 – 141, 2004.

[41] J.-L. C. Wu, Y.-W. Chen, and C.-C. Shiu, "Traffic modeling and bandwidth allocation for MPEG video sources in ATM networks," in *GLOBECOM '95*, vol. 3, 1995, pp. 2237 – 2241.

[42] E. Casilari, M. Lorente, A. Reyes, A. D. Estrella, and F. Sandoval, "Scene oriented model for VBR video," *Electronics Letters*, vol. 34, no. 2, pp. 166–168, 1998.

[43] F. Yegenoglu, B. Jabbari, and Y. Zhang, "Motion-classified autoregressive modeling of variable bit-rate video," *Circuits Systems Video Technology*, vol. 3, no. 1, pp. 42–53, 1993.

[44] O. Rose, "Simple and efficient models for variable bit rate mpeg video traffic," *Performance Evaluation Journal*, vol. 30, no. 1, pp. 69–85, 1997.

[45] P. Pancha and M. E. Zarki, "Bandwidth-allocation schemes for variable-bit-rate mpeg sources in atm networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, no. 3, pp. 190–198, June 1993.

[46] D. M. Lucantoni, M. F. Neuts, and A. R. Reibman, "Methods for

performance evaluation of VBR video traffic models," *IEEE/ACM Trans. Netw.*, vol. 2, no. 2, pp. 176–180, April 1994.

[47] G. Ramamurthy and B. Sengupta, "An analysis of a variable bit rate multiplexer using loss priorities," *Computer Networks and ISDN Systems*, vol. 28, no. 3, pp. 411–423, January 1996.

[48] G. Chiruvolu, T. K. Das, R. Sankar, and N. Ranganathan, "A scene-based generalized markov chain model for VBR video traffic," *ICC 98*, vol. 1, pp. 554–558, June 1998.

[49] T. P.-C. Chen and T. Chen, "Markov modulated punctured autoregressive processes for video traffic and wireless channel modeling," in *Packet Video*, April 2002.

[50] A. Lombardo, G. Morabito, and G. Schembra, "An accurate and treatable markov model of MPEG-video traffic," in *IEEE INFOCOM '98*, vol. 1, March 1998, pp. 217–224.

[51] Y. Sun and J. Daigle, "A source model of video traffic based on full-length VBR MPEG4 video traces," in *IEEE Global Telecommunications Conference GLOBECOM 05*, vol. 2, 28 November-2 December 2005.

[52] J.-A. Zhao, B. Li, C.-W. Kok, and I. Ahmad, "MPEG-4 video transmission over wireless networks: A link level performance study," *Wireless Networks*, vol. 10, no. 2, pp. 133–146, 2004.

[53] J.-A. Zhao, B. Li, C.-W. Kok, and I. Ahmad, "Priority scheduling with ARQ control: performance model for packet video in wireless networks," in *2002 International Symposium on Performance Evaluation of Computer and Telecommunication Systems, SPECTS 2002*, San Diego, USA, July 2002.

[54] H. E. Hurst, "Long-term storage capacity of reservoirs," *Transactions of the American Society of Civil Engineers*, vol. 116, pp. 770–799, 1951.

[55] M. W. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar vbr video traffic," in *ACM SIGCOMM*, London, UK, August 1994, pp. 269–280.

[56] C. Huang, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye, "Modeling and simulation of self-similar variable bit rate compressed video: A unified approach," in *ACM SIGCOMM*, August 1995, pp. 114–125.

[57] H. Liu, N. Ansari, and Y. Q. Shi, "Modeling VBR video traffic by markov-modulated self-similar processes," in *IEEE 3rd Workshop on Multimedia Signal Processing*, Copenhagen , Denmark, September 1999, pp. 363–368.

[58] N. Ansari, H. Liu, Y. Q. Shi, and H. Zhao, "On modeling MPEG video traffics," *IEEE Trans. Broadcast.*, vol. 48, no. 4, pp. 337–347, December 2002.

[59] R. Narasimha and R. M. Rao, "Discrete-Time Self-Similar Systems and Stable Distributions: Applications to VBR video modeling," *IEEE Signal Process. Lett.*, vol. 10, no. 3, pp. 65–68, March 2003.

[60] M. Krunz and A. Makowski, "Modeling video traffic using m/g/infinity input processes: A compromise between Markovian and LRD models," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 5, pp. 733–748, June 1998.

[61] S. Ma and C. Ji, "Modeling heterogeneous network traffic in wavelet domain," *IEEE/ACM Trans. Netw.*, vol. 9, no. 5, pp. 634–649, October 2001.

[62] X. Wang, S. Jung, and J. S. Meditch, "VBR broadcast video traffic modeling- a wavelet decomposition approach," in *IEEE Globecom '97*, vol. 2, November 1997, pp. 1052–1056.

[63] G. K. Wallace, "JPEG still picture compression standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, April 1991.

[64] T. Sikora, "Trends and perspectives in image and video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 6–17, January 2005.

[65] D. Arifler and B. L. Evans, "Modeling the self-similar behavior of packetized MPEG-4 video using wavelet-based methods," in *International Conference on Image Processing*, vol. 1, Rochester, NY USA, September 2002, pp. 848–851.

[66] M. Dai and D. Loguinov, "Analysis and modeling of MPEG-4 and H.264 multi-layer video traffic," in *IEEE INFOCOM*, vol. 4, March 2005, pp. 2257–2267.

[67] O. Lazaro, D. Girma, and J. Dunlop, "Statistical analysis and evaluation of modelling techniques for self-similar video source traffic," in *The 11th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications 2000*, vol. 2, London, UK, September 2000, pp. 1540–1544.

[68] J. R. Hill and B. Melamed, "TEStool: A visual interactive environment for modeling autocorrelated time series," *Performance Evaluation*, vol. 24, no. 1-2, pp. 3–22, 1995.

[69] A. Matrawy, I. Lambadaris, and C. Huang, "MPEG4 traffic modeling using the transform expand sample methodology," in *4th IEEE International Workshop on Networked Appliances (IWNA4)*, Gaithersburg, MD, 2002, pp. 249–256.

[70] A. Al-Tamimi, C. So-In, and R. Jain, "Modeling and resource allocation for mobile video over WiMAX broadband wireless networks," *IEEE J. Sel. Areas Commun., Special Issue on Wireless Video Transmission*, vol. 28, no. 3, pp. 354–365, April 2010.

[71] A. A. Tamimi, R. Jain, and C. So-In, "Modeling and generation of AVC and SVC-TS mobile video traces for broadband access networks," in *First annual ACM SIGMM conference on Multimedia systems (MMSys '10)*, Phoenix, Arizona, USA, February 2010, pp. 89–98.

[72] C. Chatfield, *The Analysis of Time Series: An Introduction*, 6th ed. Chapman and Hall/CRC, 2003.

[73] (2009) The project R of statistical computing. [Online]. Available: http://www.r-project.org/

[74] B. Melamed, D. Raychaudhuri, B. Sengupta, and J. Zdepski, "TES-Based video source modeling for performance evaluation of integrated networks," *IEEE Trans. Commun.*, vol. 42, no. 10, pp. 2773–2777, 1994.

[75] D. Reininger, B. Melamed, and D. Raychaudhuri, "Variable bit rate MPEG video: Characterization, modeling and multiplexing," in *Proc. 14th International Teletraffic Congress*, vol. 1a, Antibes Juanles- Pins, France, June 1994, pp. 295–306.

[76] B. Melamed and D. Pendarakis, "A TES-Based Model for compressed star wars video," in *IEEE GLOBECOM*, 1994, pp. 120–126.

[77] M. R. Ismail, I. E. Lambadaris, M. Devetsikiotis, and A. R. Kaye, "Modelling prioritized MPEG video using TES and a frame spreading strategy for transmission in ATM networks," in *INFOCOM '95 Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communication Societies*, vol. 2, Boston, MA, 2-6 April 1995, pp. 762–770.

[78] P. Seeling and M. Reisslein, "Video transport evaluation with H.264 video traces," *IEEE Commun. Surveys Tutorials*, vol. 14, no. 4, pp. 1142–1165, 2012.

[79] W. Zhou, D. Sarkar, , and S. Ramakrishnan, "Traffic models for MPEG-4 spatial scalable video," in *Globecom 2005*, 28 November - 2 December 2005, pp. 256–260.

[80] J. Zhao, B. Li, and I. Ahmad, "Traffic model for layered video: An approach on markovian arrival process," in *Packet Video 2003*, Nantes, France, April 2003.

[81] D. Fiems, B. Steyaert, and H. Bruneel, "A genetic approach to markovian characterisation of H.264/SVC scalable video," *Multimedia Tools and Applications*, vol. 58, no. 1, pp. 125–146, 2012.

[82] A. Vetro, A. M. Tourapis, K. Muller, and T. Chen, "3D-TV content storage and transmission," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 384–394, 2011.

[83] A. Pulipaka, P. Seeling, M. Reisslein, and L. J. Karam, "Traffic and statistical multiplexing characterization of 3d video representation formats," Submitted, 2012.

[84] P. Merkle, K. Muller, and T. Wiegand, "3D video: acquisition, coding, and display," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 946–950, May 2010.

[85] L. Rossi, J. Chakareskia, P. Frossard, and S. Colonnese, "A non-stationary hidden markov model of multiview video traffic," in *IEEE Int. Conf. on Image Processing (ICIP)*, 2010, pp. 2921–2924.

[86] A. A. Tamimi, R. Jain, and C. So-In, "Modeling and prediction of high definition video traffic:a real-world case study," in *Second International Conferences on Advances in Multimedia*, 2010, pp. 168–173.

[87] B. Anjum, "Bandwidth allocation under end-to-end percentile delay bounds," Ph.D. dissertation, Department of Computer Science, North Carolina State University, Raleigh, NC, April 2012.

[88] Matlab tutorial: Simulating dependent random variables using copulas. [Online]. Available: webpage: http://www.mathworks.com/products/statistics/examples.html?-file=/products/demos/shipping/stats/copulademo.html

Savera Tanwir received the BE degree in Computer Software Engineering from College of Signals, National University of Science and Technology (NUST), Pakistan in 2004 and Masters in Computer Networking from North Carolina State University (NCSU), USA in 2007. She was a lecturer at NUST School of Electrical Engineering and Computer Science from 2007 to 2010. She is currently a PhD student in the department of Computer Science at North Carolina State University. Her primary areas of research include computer networks, network resource scheduling and management, video traffic modeling and performance monitoring of computer networks.

Harry G. Perros is an IEEE Fellow, a Professor of Computer Science, an Alumni Distinguished Graduate Professor, and the Program Coordinator of the Master of Science degree in Computer Networks at NC State University.

He received the B.Sc. degree in Mathematics in 1970 from Athens University, Greece, the M.Sc. degree in Operational Research with Computing from Leeds University, England, in 1971, and the Ph.D. degree in Operations Research from Trinity College Dublin, Ireland, in 1975. He has held visiting faculty positions at INRIA, Rocquencourt, France (1979), NORTEL, Research Triangle Park, North Carolina (1988-89 and 1995-96), University of Paris 6, France (1995-96, 2000, 2002, and 2012), University of Paris 13, France (2005-2006), and Victoria University, Wellington, New Zealand (2006).

He has published extensively in the area of performance modelling of computer and communication systems, and he has organized several national and international conferences. He has also published three print books: Queueing Networks with Blocking: Exact and Approximate Solutions, Oxford University Press 1994, An Introduction to ATM Networks, Wiley 2001, Connection-Oriented Networks, Wiley 2005, and an e-book Computer Simulation Techniques The Definitive Introduction, 2002. He is currently completing a book on Networking Services.

In 1995 he founded the IFIP Working Group 6.3 on the Performance of Communication Systems, and he was the chairman from 1995 to 2002. He is also an associate Editor of several Journals including the Performance Evaluation Journal, and the Telecommunications Systems Journal.His current research interests are in the areas of next generation networks, multi-domain routing, resource allocation under QoS, and queueing theory.