

Research Article

Statistical Time-Frequency Multiplexing of HD Video Traffic in DVB-T2

Mehdi Rezaei,¹ Imed Bouazizi,² and Moncef Gabbouj¹

¹Department of Signal Processing, Tampere University of Technology, 33720 Tampere, Finland

²Media Laboratory, Nokia Research Center, 33720 Tampere, Finland

Correspondence should be addressed to Mehdi Rezaei, mehdi.rezaei@ieee.org

Received 27 May 2008; Revised 25 August 2008; Accepted 21 October 2008

Recommended by Susanna Spinsante

Digital video broadcast-terrestrial 2 (DVB-T2) is the successor of DVB-T standard that allows a two-dimensional multiplexing of broadcast services in time and frequency domains. It introduces an optional time-frequency slicing (TFS) transmission scheme to increase the flexibility of service multiplexing. Utilizing statistical multiplexing (StatMux) in conjunction with TFS is expected to provide a high performance for the broadcast system in terms of resource utilization and quality of service. In this paper, a model for high-definition video (HDV) traffic is proposed. Then, utilizing the proposed model, the performance of StatMux of HDV broadcast services over DVB-T2 is evaluated. Results of the study show that implementation of StatMux in conjunction with the newly available features in DVB-T2 provides a high performance for the broadcast system.

Copyright © 2009 Mehdi Rezaei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

Digital video broadcast-terrestrial 2 (DVB-T2) is going to be a new European Telecommunications Standards Institute (ETSI) standard specification for digital terrestrial television. DVB-T2 is an upgrade of the DVB-T system designed to provide new high quality services. It utilizes advanced techniques that provide more flexibility for the broadcast system. Figure 1 shows an overview of the DVB-T2 system with its main components. The *generic stream encapsulation (GSE)* module encapsulates protocol data units in a protocol-independent manner into GSE packets, which are arranged into the so-called *baseband (BB)* frames by the *input stream processor* module. *Forward error correction (FEC)* encoding is performed at the *bit-interleaved coding* module using a *low-density parity code (LDPC)* concatenated to a BCH code. Subsequent interleaving and mapping to *physical layer (PL)* frames as well as OFDM symbol mapping is performed at the *frame mapper* module. The resulting PL frames are then passed to the *modulator* modules for modulation and transmission. The newly defined modulation modes 64 and 256 QAM and OFDM carrier modes significantly enhance the spectral efficiency; achieving bandwidths of up to 40 Mbps (not accounting for signaling overhead) thus,

enabling the broadcast of HDTV services over terrestrial networks.

As yet, another new feature, DVB-T2, utilizes an optional time-frequency slicing (TFS) scheme for data transmission that provides a great flexibility for system design so that a different range of services can be deployed in the system. In this approach, multiple radio frequency (RF) channels are combined into a coherent high-capacity channel to utilize advantages of statistical multiplexing (StatMux) across several high-definition (HD) services. It allows implementing a two-dimensional StatMux over the services to improve the performance of the broadcast system.

In digital communications, video signals are compressed in order to use transmission bandwidth efficiently. In video compression, a video sequence can be encoded at a constant bit rate (CBR) or a variable bit rate (VBR) bit stream. With a similar average bit rate, VBR bit streams consume more resources in terms of transmission bandwidth and delay than CBR bit streams. When encoding a CBR bit stream, a rate controller strictly controls the bit rate mainly by adjusting the quantization parameter (QP). Generally, a CBR can be achieved by large variations in QP and also in video quality. A VBR video bit stream can be produced by encoding a video sequence with or without a rate controller.

In uncontrolled VBR, a constant QP is used for encoding to provide a quasiconstant and better visual quality for compressed video. In controlled VBR, the QP is controlled by a soft rate controller to smooth the variations in the bit rate and also in the quality. Generally, in comparison with CBR, controlled VBR can provide a better visual quality at the expense of more variations in the bit rate. On the other hand, in comparison with uncontrolled VBR, controlled VBR can provide less variation in bit rate at the expense of more variations in the quality.

In video broadcasting, the video sources are encoded to VBR bit streams to provide a better average quality for broadcasted services. However, VBR services need more resources in terms of transmission bandwidth and delay than CBR services. When several VBR video services are broadcasted simultaneously, utilizing StatMux can improve the bandwidth efficiency and end-to-end delay of the broadcast system.

In StatMux, a fixed bandwidth communication channel is shared for transmitting several bit streams. The channel is virtually divided into several variable bandwidth channels that are adapted to the variations in the bit rate of the bit streams. The attempt is to distribute the channel capacity among the bit streams dynamically according to the required bandwidth by the bit streams such that a virtual variable bandwidth channel is allocated to each bit stream.

The performance of StatMux depends on the statistical properties of the multiplexed bit streams as well as the number of bit streams. The statistical properties of video bit streams depend on the encoding parameters such as bit rate, frame rate, and picture size as well as video content and the rate control method. On the other hand, the number of services depend on the service bit rates and the channel capacity. Consequently, the performance of StatMux is application dependent and it should be evaluated specifically for each application. The TFS, introduced in DVB-T2, that allows implementation of StatMux in two dimensions, makes this application more specific. The main goal of this research is to evaluate the performance of StatMux specifically in DVB-T2 by computer simulations. To obtain accurate evaluation results, multiplexing simulations should be repeated many times with different video bit streams. A huge amount of traffic is needed that can be provided synthetically by a video traffic model. The accuracy of the simulation results depends on the accuracy of the model. Therefore, the first attempt is to provide an accurate model for video traffic in this application. Studying statistical properties of HDV traffic, a model for VBR video traffics is proposed in this paper. Then, the proposed traffic model is used to generate synthetic traffic for evaluating the performance of StatMux in DVB-T2.

The rest of this paper is organized as follows. Background information about the VBR video traffic modeling is presented in Section 2. The proposed model for VBR video traffic is presented in Section 3. In Section 4, the performance of StatMux in DVB-T2 is evaluated. Some simulation results are presented in Section 5. The paper is closed with conclusions in Section 6.

2. VBR Video Traffic Modeling

Accurate modeling of VBR video traffic is important in this research. A good model predicts or provides a desired metric or a set of desired metrics for the modeled data similar to the original data. For example, if the packet loss probability is the desired metric, then a good model produces traffic that precisely provides this metric of interest in simulations.

Generally, the performance of a communication network in terms of delay, data drop rate, and bandwidth usage depends on the statistical properties of the traffic in the network. For example, the autocorrelation function (ACF) of the service traffic has a major impact on the performance of communication networks. VBR video traffic was found to exhibit self-similar characteristics [1]. In mathematics, a self-similar object is exactly or approximately similar to a part of itself, for example, the whole has the same shape as one or more of the parts. Self-similarity is a typical property of fractals. A fractal is a rough or fragmented geometric shape that can be subdivided in parts, each of which is (at least approx.) a reduced-size copy of the whole.

The main feature of self-similar processes is that they exhibit long range dependence (LRD), that is, their autocorrelation function $r(k)$ decays less than exponentially fast, and is nonsummable, that is, $r(k) \sim k^{-\beta}$, as $k \rightarrow \infty$, for $0 < \beta \leq 1$. The quantity $H = 1 - \beta/2$ is called *Hurst parameter* or *Hurst exponent*. The Hurst exponent was originally developed in hydrology [2]. It shows whether the data is a purely *random walk* or has underlying trends. The Hurst exponent is related to the *fractal dimension* and it is a measure of the smoothness of fractal time series based on the asymptotic behavior of the *rescaled range* of the process. The Hurst exponent is defined as

$$H = \frac{E[\log(R/S)]}{\log(T)}, \quad (1)$$

where T is the duration of the data sample and R/S is the corresponding value of the rescaled range, where S denotes the standard deviation of the sample data and R stands for the difference between the max and min of accumulated deviation from the mean value during the time period T . If $H = 0.5$, the behavior of the time series is similar to a random walk process and samples are uncorrelated. When $H > 0.5$, the time series covers more distance than a random walk. In this case, the process is, namely, *persistent* and samples are positively correlated. This means that if the time series is increasing, it is more probable that it will continue to increase. When $H < 0.5$, the time series covers less distance than a random walk, in which case the process is, namely, *antipersistent* and samples are negatively correlated. This means that if the time series is increasing, it is more probable that it will then decrease, and vice versa.

In communication networks, the Hurst exponent of traffic is relevant to the buffering requirements for traffic transmission. Considering the definition of R in (1), in fact it is equal to the minimum buffering space for perfect transmission of the data during the time period T by a channel with a bandwidth equal to the average of the bit rate. Therefore, the performance of communication

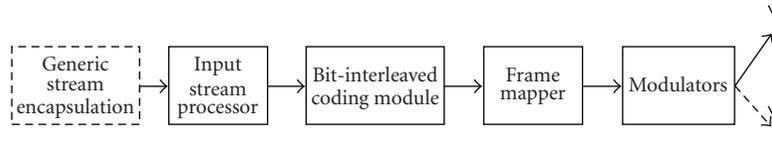


FIGURE 1: Overview of DVB-T2 system.

networks depends on the statistical properties of traffic such as self-similarity and smoothness. Many video traffic models attempt to capture these relevant statistics.

Several stochastic models for video traffic have been proposed in the past [3]. Maglaris et al. [4] used two models for a video source: a continuous-state autoregressive (AR) Markov model and a discrete-state continuous-time Markov process. Heyman et al. [5] and Lucantoni et al. [6] also used a Markov chain process to develop models for video traffic at the frame level. Grunenfelder et al. [7] used an autoregressive moving-average (ARMA) process to model video conference traffic at ATM cell level. Ramamurthy and Sengupta [8] proposed a hierarchical composite model which uses three processes: two AR processes and one Markov chain. The first AR process attempts to match ACF at short lags while the second attempts to match ACF at long lags. The Markov process captures the effects of scene changes. A combination of the three processes yields the final model. Another hierarchical model was proposed by Heyman and Lakshman in [9] that consists of three different stochastic processes for video scene length, size of the first frame in the scene, and the size of other frames in the scene, respectively. The scene change process was found to be uncorrelated and it was enough to match the distribution of scene length. It was found that the scene length distribution fits Weibull, Gamma, and Pareto distributions. It was also found that the number of ATM cells in a frame of a scene change fits Weibull and Gamma distributions. A Markov chain was used for the frame size within a video scene. Melamed et al. [10] developed a model for video traffic based on Transform-Expand-Sample (TES) process for the number of bits in one group of pictures (GOPs). TES processes are designed to fit simultaneously both the distribution and ACF of the empirical data. Lazar et al. [11] and Reiningner et al. [12] used a TES process for modeling of frame and or slice sizes. A process was used for each type of I, P, and B frames (or slices). The final model is composed according to a deterministic structure of the GOP. Garrett and Willinger [13] used a fractional autoregressive integrated moving average (F-ARIMA) process to provide a model for video traffic at the frame level. They used a hybrid distribution which consisted of a concatenation of a Gamma and a Pareto distributions for the frame size distribution. A background sequence is generated by an F-ARIMA process based on a desired value for Hurst exponent and the final sequence is generated by a transformation on the background sequence based on the parameters of the desired distribution. In a similar approach, Huang et al. [14] used an F-ARIMA process to generate background sequences for different frame types based on the value of Hurst exponent. The background sequences

were transformed by a weighted sum of exponentials to match the distribution. Krums and Tripathi [15] proposed a model in which the video scene length is generated by a geometric distribution. The size of I frames is modeled by the sum of two random components: a scene-related component and an AR-2 component that accounts for the fluctuation within the scene. The sizes of P and B frames are modeled by two processes of i.i.d. random variables with Lognormal distributions. The final model is obtained by combining the three submodels according to a given GOP pattern. Liu et al. [16] proposed a video traffic model in which a hybrid Gamma-Pareto distribution is used for all three types of frames and the autocorrelation structure is modeled using two second-order nested AR processes. One AR process is used to generate the mean frame size of the scenes to model the long-range dependence and the other is used to generate the fluctuations within the scene to model the short-range dependence. Sarkar et al. [17] proposed another model for VBR video traffic in which a video sequence is segmented using a classification based on size of three types of video frames. In each class, the frame sizes are produced by a shifted Gamma distribution. Markov renewal processes model video segment transitions. Dai et al. [18] presented a hybrid wavelet framework for modeling VBR video traffic. They modeled the size of I frames in the wave domain and the size of P and B frames based on the intra-GOP correlation. The reviewed models above are samples of different approaches. The review is not exhaustive and some related approaches are not reviewed in this paper.

The proposed models for VBR video traffic in earlier works attempt to fit some statistical properties such as frame size distribution, ACF, and Hurst exponent for sample video traffic data that are encoded for a special application (e.g., video conference) by a particular encoder (e.g., H.263, MPEG-4). Then, the proposed models have been validated based on some practical measures such as data drop rate and delay in buffering simulations.

There are some concerns about the previous proposed traffic models. The first concern is that most of the models have been built based on a limited number of sample real bit streams; therefore, the accuracy of these models is limited to special applications in terms of video content, encoding method, and encoding parameters. The second concern is that, although these models capture some statistical properties of the traffic that may be correlated with practical metrics of interest, the correlation may not be always accurate. For example, it is possible to find bit streams with different Hurst exponents and similar practical performance in terms of data drop rate and delay and also it is possible to find bit streams with similar statistical properties that have different

performances in terms of data drop rate and delay. Some examples are shown in Section 5. The other concern is that all practically possible traffics may not be covered by a model that captures only some statistical parameters. On the other hand, some synthetic bit streams may be generated by the models which is difficult to find a match for them among the real bit streams because some practical constraints that exist on real bit streams are not considered in the models. These concerns affect the accuracy of the simulation results, where synthetic traffics are used.

Considering the concerns above, in a new approach, a model for VBR video traffic is proposed in this paper. In the new approach, the first attempt is to capture the practical metrics of interest, such as buffering parameters, while some statistics are used. The new model is not limited to any special distribution, ACF, or range of Hurst exponent. The new modeling approach simulates the interaction between the video encoder and the video source to generate a synthetic video traffic. The interaction of the encoder with the video source is controlled by a rate controller. Unlike previous modeling approaches in which first statistical properties such ACF are captured to achieve practical properties such as buffering parameters, in the new approach, practical properties are captured directly. The model is tuned similar to a video encoder with a rate controller to generate traffics with desired buffering properties. The practical and statistical properties of a video traffic depend on the video content properties, encoding method, and rate control algorithm. Accordingly, the proposed model can generate various traffics according to the content, for example, sport, movie, news, and so on. Also, it can produce video traffics according to the encoding parameters such as bit rate, frame rate, picture size, and so on. Moreover, it can produce video traffics according to rate control parameters such as buffering delay. These features are beneficial in simulation tasks in which the effects of content properties and encoding parameters on the results of simulation are studied.

From a modeling point of view, the self-similarity properties of VBR video traffic depend on the degree of control that is imposed on the bit rate. While uncontrolled VBR bit streams usually have persistent behaviors with a large Hurst exponent, the controlled VBR bit streams, depending on the degree of control, tend toward the antipersistent case.

We proposed a model for antipersistent video traffic in [19]. A multi-Gamma model was proposed for video frame sizes in which a Gamma distribution is considered for each picture type (e.g., I, P, and B) in each video scene. The proposed model has many parameters to be determined. Considering the functionality of the video rate controller and assuming uniform distributions over some parameters of the model, the final model parameters are reduced to few parameters. Later on, statistics collected from a large video database showed that the assumed uniform distributions should be modified to Gamma distributions. Accordingly, a modified version of the model is presented in [20]. The modified parts of the model are used for the case in which synthetic bit streams are generated without any prototype bit streams. However, the results presented in [19, 20] do not show the effect of these modifications because the

models have been validated for the case in which they have been parameterized based on extracted parameters from a prototype bit stream not based on the provided statistics in the modified parts.

The proposed traffic model in this paper is a modified and a generalized form of our previous models. The previous models can generate only antipersistent traffics in which $H < 0.5$, while the new model proposed in this paper can be used for both persistent and antipersistent traffics, that is, the Hurst exponent can assume any value in the range of $0 < H < 1$. The previous models were targeted for controlled VBR with small variations in the bit rate while the new model can be used for a wider range of VBR video including controlled and uncontrolled bit streams with any level of variations in the bit rate. In the previous models, a constant average bit rate is assigned to all video scenes while in the new model video scenes may have different average bit rates that are defined according to a Gamma distribution and also according to the buffering constraint imposed on the bit stream. The self-similarity and LRD properties of video traffics are captured indirectly when the buffering constraint is imposed on the bit stream. Generating synthetic traffic by the proposed model is straightforward with a low degree of complexity.

3. Proposed Model for VBR Video Traffics

A video sequence includes several scenes and each scene includes a number of video frames from different types such as I, P, and B frames. According to the proposed model, a Gamma distribution is used for each frame type in each video scene. Note that at the sequence level, each frame type can have a PDF which may be very different from the scene level because the PDFs of video scenes are combined together at the sequence level. In the proposed model, the PDF of each frame type can have any distribution at the sequence level. Although other distributions such as Lognormal may be used at the scene level, the Gamma distribution has been used because it fits well enough the practical results and it simplifies the modeling approach. According to the model, a Gamma PDF for the size of frame (x) of type i in scene s is considered as

$$\text{Gamma}(x, k_{is}, \theta_{is}) = x^{k_{is}-1} \frac{e^{-x/\theta_{is}}}{\theta_{is}^{k_{is}} \Gamma(k_{is})}, \quad x > 0, \quad (2)$$

where $k_{is} > 0$ is the shape parameter and $\theta_{is} > 0$ is the scale parameter of Gamma distribution. i stands for I, P, or B frame type. $s = 1, \dots, S$ denotes the scene index.

To generate a synthetic video traffic by the proposed model, several parameters should be determined. The main parameters include the total number of frames in the video sequence (N), structure of GOP, that is, the number of P pictures (N_P) and B picture (N_B) in GOP, the length of video scenes as well as their parameters (k_{is}, θ_{is}), average bit rate (B), frame rate (F), and smoothing buffer size (S_B). To produce synthetic traffics, the main parameters such as N, N_P, N_B, B, F , and S_B are set directly by the user whereas the remaining parameters are determined as explained in the sequel.

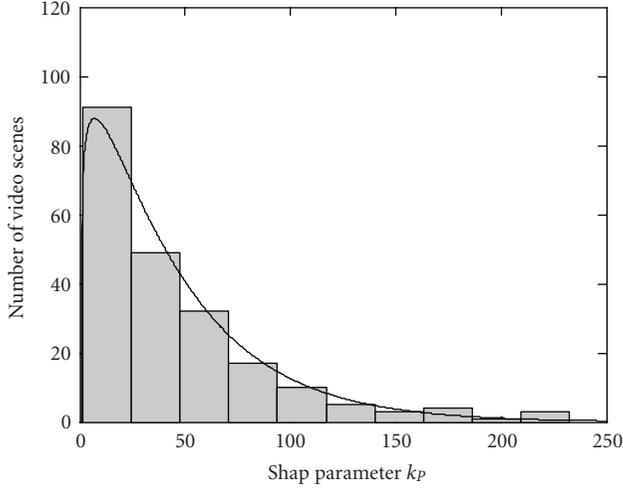


FIGURE 2: Histogram of shape parameter for Gamma distributions of P frames (k_p) over video scenes that is fitted to a Gamma PDF.

Statistics collected from a large video database show that a Gamma PDF can be considered over the length of video scenes as

$$P_{L_s} = \text{Gamma}(L_s, k_{L_s}, \theta_{L_s}), \quad (3)$$

where L_s denotes the length of a video scene s . This distribution is used to generate the length of video scenes. The shape and scale parameters (k_{L_s}, θ_{L_s}) are content dependent. Moreover, the statistics show that a Gamma PDF can be assumed also for the shape parameters k_{I_s}, k_{P_s} , and k_{B_s} used in (2) over the scenes as

$$\begin{aligned} P_{k_I} &= \text{Gamma}(k_I, k_{k_I}, \theta_{k_I}), \\ P_{k_P} &= \text{Gamma}(k_P, k_{k_P}, \theta_{k_P}), \\ P_{k_B} &= \text{Gamma}(k_B, k_{k_B}, \theta_{k_B}). \end{aligned} \quad (4)$$

These distributions are used to generate the shape parameters of the distributions used in (2). A sample histogram of k_p and its related Gamma PDF are depicted in Figure 2. More details about the collected statistics are presented in Section 5.

As a new measure, *relative coding complexity* is defined. This measure reflects the video content properties as well as the encoding parameters. The relative coding complexity is defined between two picture types. The relative complexity of I to P and I to B pictures in a scene s is defined as $X_{P_s} = \bar{I}_s/\bar{P}_s$ and $X_{B_s} = \bar{I}_s/\bar{B}_s$, respectively, where \bar{I}_s, \bar{P}_s , and \bar{B}_s denote the average size of I, P, and B pictures, respectively, in the scene s . The relative coding complexity is a known concept that is used in some control algorithms, for example, in [21, 22]. Experimental results show that the values of relative complexities are not only dependent on the properties of video content such as motion activities but they are also dependent on the encoding parameters, such as bit rate, frame rate, GOP structure, and picture dimensions. Moreover, they are affected by the rate control algorithm and the smoothing buffer size. Statistics from

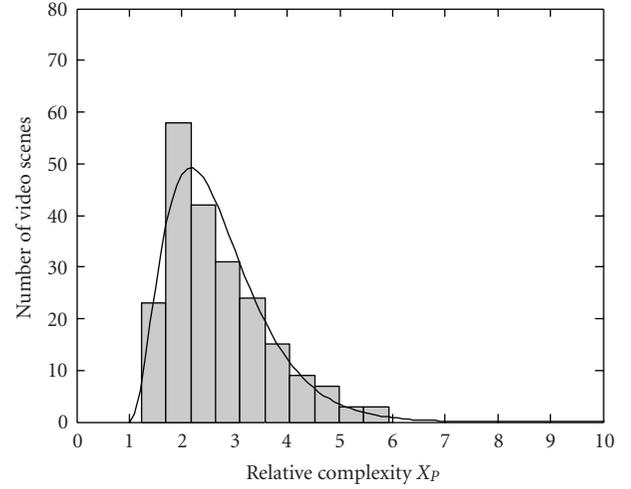


FIGURE 3: Histogram of the relative complexity X_p over video scenes that is fitted to a Gamma PDF.

different video sequences which are encoded with similar encoding parameters show that the values of $X_{P_s} - 1$ and $X_{B_s} - 1$ have distributions close to Gamma PDF over the scenes as

$$\begin{aligned} P_{X_{P_s}-1} &= \text{Gamma}(X_{P_s} - 1, k_{X_P}, \theta_{X_P}), \\ P_{X_{B_s}-1} &= \text{Gamma}(X_{B_s} - 1, k_{X_B}, \theta_{X_B}). \end{aligned} \quad (5)$$

These distributions are used to generate values for the relative complexities. A sample histogram of X_p collected from real traffics and related Gamma PDF are shown in Figure 3.

To find the remaining parameters, a long-term average bit rate is defined for the video sequence, while it may exhibit a large variation over the scenes. In our previous models presented in [19, 20] for antipersistent traffic, it was assumed that all video scenes in the sequence have a similar average bit rate. To generalize the model, it is assumed that video scenes can have different average bit rates B_s . However, some constraints over the average bit rates of a video scene are imposed to provide a buffering constraint over the final bit stream. Generally, a Gamma PDF can be assumed for the average bit rate of video scenes as follows:

$$P_{B_s} = \text{Gamma}(B_s, k_{B_s}, \theta_{B_s}), \quad (6)$$

where the distribution parameters depend on the control strength and smoothing buffer size. Using this distribution, preliminary values for the average bit rates are generated. The preliminary values are modified if the final bit stream should be constrained to a buffering constraint. Consider that the desired bit stream has an average bit rate B over all scenes and it preserves a buffering constraint with buffer size S_B . To achieve the buffering constraint, similar to a rate controller, the following constraints are imposed on the preliminary values of the scene bit rates:

$$0 < \left(\sum_{s=1}^n B_s \cdot L_s / F - B / F \cdot \sum_{s=1}^n L_s \right) < S_B, \quad n = 1, \dots, S, \quad (7)$$

where F denotes the video frame rate. The first term in the parenthesis corresponds to the expected value of the overall input to the buffer and the second term corresponds to the overall output from the buffer. Therefore, this condition can guarantee a kind of buffering constraint based on the expected values of scene bit rate. This condition is examined for all n from 1 to S . If it is not met for some values of n , then the value of B_n is corrected by a minimum change such that the condition is met. The resulting bit stream is constrained to an expected buffer size. However, the buffer constraint is not strict because it is imposed based on expected values of scene bit rates. To ensure a strict buffering constraint for the bit stream, margins are considered for the critical buffer conditions and formula (7) is rewritten as

$$M_L S_B F < \left(\sum_{s=1}^n B_s L_s - B \sum_{s=1}^n L_s \right) < M_H S_B F, \quad n = 1, \dots, S, \quad (8)$$

where M_L and M_H are two margins (e.g., 0.2 and 0.8) for low and high buffer fullness states, respectively.

For a GOP in a video scene, the average frame size can be estimated as

$$\bar{x} = \frac{\mu_{I_s} + N_P \mu_{P_s} + N_B \mu_{B_s}}{1 + N_P + N_B} = \frac{B_s}{F}. \quad (9)$$

From the definition of relative complexity, it is concluded that

$$\mu_{I_s} = \mu_{P_s} X_{P_s} = \mu_{B_s} X_{B_s}, \quad (10)$$

where μ_{is} denotes the mean frame size of type i in a video scene s . Combining (9) and (10), the values of μ_{I_s} , μ_{P_s} , and μ_{B_s} are obtained for each video scene. For a Gamma distribution, $\text{Mean} = k\theta$; and therefore, the scale parameters are obtained as

$$\theta_{I_s} = \frac{\mu_{I_s}}{k_{I_s}}, \quad \theta_{P_s} = \frac{\mu_{P_s}}{k_{P_s}}, \quad \theta_{B_s} = \frac{\mu_{B_s}}{k_{B_s}}. \quad (11)$$

The shape parameters have been already generated by (4). Now, all the required parameters for generating the video scenes and the desired bit stream are available.

There are only few parameters that are defined by the user for the model and still they can be reduced. Experimental results show that the model is not very sensitive to the shape parameters used in Gamma distributions (3), (4), and (5) for a relative wide range of bit streams. Therefore, it is enough to consider constant values for k_{S_L} , k_{k_I} , k_{k_P} , k_{k_B} , k_{X_P} , and k_{X_B} in the model. The user only defines the mean values for L_s , k_I , k_P , k_B , X_P , and X_B then the scale parameters are calculated according to the shape parameters and the mean values by $\text{Mean} = k\theta$. Typical values for k_{S_L} , k_{k_I} , k_{k_P} , k_{k_B} , k_{X_P} , and k_{X_B} are 1.5, 5, 3, 3, 2.5, and 2.5, respectively. The algorithm of generating synthetic video traffics is summarized as follows.

- (1) Define the desired encoding parameters including the number of frames (L), the average bit rate (B), the frame rate (F), and the GOP structure (N_P, N_B).
- (2) Define the mean values for L_s , k_I , k_P , k_B , X_P , and X_B according to the content and encoding parameters.

- (3) Using the mean values of L_s , k_I , k_P , k_B , X_P , and X_B , calculate the scale parameters θ_{L_s} , θ_{k_I} , θ_{k_P} , θ_{k_B} , θ_{X_P} , and θ_{X_B} according to $\text{Mean} = k\theta$.

- (4) Using (3), generate S scene length L_s , such that

$$\sum_{s=1}^S L_s \geq L. \quad (12)$$

- (5) Using (6) and (8), generate the scene bit rates.

- (6) Using (5), generate the relative complexities X_{P_s} and X_{B_s} .

- (7) Combine (9) and (10), calculate μ_{I_s} , μ_{P_s} , and μ_{B_s} for each video scene.

- (8) Using (4), generate k_{I_s} , k_{P_s} , k_{B_s} .

- (9) Using (11), calculate θ_{I_s} , θ_{P_s} , and θ_{B_s} for the scenes.

- (10) Using (2), generate the frame sizes for each video scene.

4. Performance of StatMux in DVB-T2

In this section, the performance of StatMux in DVB-T2 is evaluated by simulations. In the TFS transmission scheme as defined by DVB-T2, the service data is transmitted as time-frequency slices, that is, time-slice frames that are transmitted by parallel radio channels. The time slices have durations of about a few hundred milliseconds (typically 180 milliseconds) and a number of maximum 6 RF channels can be used for transmission of time-sliced data. Figure 4 shows an example of a TFS frame for 4 RF channels and 15 services. There is a time shift between the services in different RF channels to enable frequency hopping at the receiver. At the beginning of each frame, two synchronizing symbols are inserted (shown as P1 and P2 in the figure). The synchronization symbols allow a receiver to rapidly detect the presence of DVB-T2 signal, as well as to synchronize to the frame. Data related to a number of different services can be statistically multiplexed over the two dimensions of time and frequency. Performance of StatMux in DVB-T2 depends on the bandwidth of the coherent transmission channel, the number of multiplexed services, and the statistical properties of service traffics. A set of comprehensive simulations were performed to evaluate the performance of StatMux of HDV services over DVB-T2.

4.1. Simulations. To evaluate the performance of StatMux in DVB-T2, StatMux is compared with deterministic multiplexing (DetMux) in which a fixed bandwidth is allocated to each service. To provide accurate results, the multiplexing simulations were performed as close as possible to a real system. Service traffics were generated with parameters similar to typical real traffics and typical values were selected for simulation parameters. According to the simulation, for each service, video frames are packetized into protocol data unit (PDU) and then GSE packets [23]. BB frames are formed from the GSE packets and FEC parity check data with a code rate of 1/4 were added [24]. BB frames are

| RF 1 | RF 2 | RF 3 | RF 4 |
|-----------|-----------|-----------|-----------|
| 15 | 11 | 7 | Service 3 |
| 14 | 10 | 6 | Service 2 |
| 13 | 9 | 5 | Service 2 |
| 12 | 8 | 4 | Service 1 |
| 11 | 7 | Service 3 | Service 1 |
| 10 | 6 | Service 2 | Service 1 |
| 9 | 5 | Service 2 | 15 |
| 8 | 4 | Service 1 | 14 |
| 7 | Service 3 | Service 1 | 13 |
| 6 | Service 2 | Service 1 | 12 |
| 5 | Service 2 | 15 | 11 |
| 4 | Service 1 | 14 | 10 |
| Service 3 | Service 1 | 13 | 9 |
| Service 2 | Service 1 | 12 | 8 |
| Service 2 | 15 | 11 | 7 |
| Service 1 | 14 | 10 | 6 |
| Service 1 | 13 | 9 | 5 |
| Service 1 | 12 | 8 | 4 |
| P2 | P2 | P2 | P2 |
| P1 | P1 | P1 | P1 |

FIGURE 4: Example of a TFS frame for 4 RF channels and 15 services.

buffered in the service buffers. In a real system, convolutional interleaving is performed on BB frames, that is not essential for the multiplexing performance and, hence, it is not implemented in the simulations. Multiplexing simulations are performed over the BB frames stored in the service buffers. Detailed simulation parameters are presented in Section 5. Multiplexing algorithms are explained in the sequel.

4.2. Multiplexing Algorithms. In an ideal case of StatMux, the available bandwidth is distributed between the services proportional to their temporal required bandwidth. A multiplexing algorithm was used in the simulation that performs close to the ideal case. According to the method used, the TFS frames are formed such that the number of allocated BB frames to each service is proportional to the amount of stored BB frames in the service buffer. As a simple case, consider the case of N services being multiplexed that have similar average bit rates and each TFS frame carries B_{TFS} number of BB frames. When forming a TFS frame, if the service buffers contain B_1, B_2, \dots , and B_N number of BB frames, b_1, b_2, \dots , and b_N number of BB frames from services 1, 2, \dots , and N , respectively, are used for forming the TFS frame such that

$$b_i = \frac{B_{TFS} B_i}{\sum_{j=1}^N B_j}, \quad i = 1, 2, \dots, N. \quad (13)$$

In a general case in which the multiplexed services have different average bit rates, the buffer occupancies are normalized to the average service bit rates as

$$b_i = \frac{B_{TFS} B_i / R_i}{\sum_{j=1}^N B_j / R_j}, \quad i = 1, 2, \dots, N, \quad (14)$$

where R_i denotes the average bit rate of the i th service.

In the simulation of DetMux, TFS frames are formed such that a fixed number of BB frames is allocated to each

service in all TFS frames. Details of simulation parameters are presented in Section 5.

5. Simulation Results

Some simulation results are presented in this section that can be divided into two parts. The first part is related to the proposed video traffic modeling approach and the validation of the model. The second part of the results presents the performance of StatMux in DVB-T2.

To collect some statistics from real video bit streams, a comprehensive study on a large set (40 sequences) of long (about 2500 to 5000 frames per sequence) HDV sequences was performed. After a preliminary study, a number of 25 HDV sequences with a resolution of 1280×720 (720p) were selected from [25–27]. The selected video sequences, which were encoded with a bit rate higher than 6 MB/s, were decoded and used as source signals when they are again encoded at a bit rate of 6 MB/s in our simulations. The video sequences were encoded several times by the FFMPEG H.264/AVC encoder with different buffering constraints [28]. A VBR rate controller is implemented in FFMPEG encoder that was used in the simulations. Smoothing buffers with sizes corresponding to 0.5, 1, 2, 3, 4, and 10 seconds buffering delay were used for the rate control. Moreover, the sequences were encoded with constant QPs and without any buffering constraint. Various statistics related to the proposed traffic model were collected. These include video scene length, scene bit rate, relative complexity of picture types, shape and scale parameters of the Gamma PDFs, Hurst exponent, minimum buffering delay, variance, and mean of different picture types. The collected statistics formed a rich database that was used for building and parameterizing the proposed traffic model. Few hundred video scenes were used in the simulations. Due to space limitation, the results presented in this section constitute only a small part of collected results.

As sample results, Figure 5 to Figure 10 compare the results of encoding “*The Living Sea*” video sequence in two cases: uncontrolled VBR and controlled VBR cases. The other encoding parameters such as average bit rate, frame rate, and GOP structure are similar for both cases. The fullness of the decoder buffer (with zero buffering period) and the size of the video frames for the two cases are shown in Figures 5 and 8. Histograms of I and P frames are depicted in Figures 6 and 9 for the two cases. Figures 7 and 10 show the ACF of video frames size for the two cases. The figures show that the size of the video frames, the distribution of the video frame size, and the ACF are very different in the two cases. These sample results prove that the statistical properties of VBR video traffics depend on the encoding process. Therefore, the encoding process is considered in the proposed modeling approach.

To show the relation between the statistical properties and the practical parameters of real bit streams, the Hurst exponent and the minimum buffering delay for a number of encoded bit streams were measured and are depicted in Figure 11. This figure shows that some bit streams

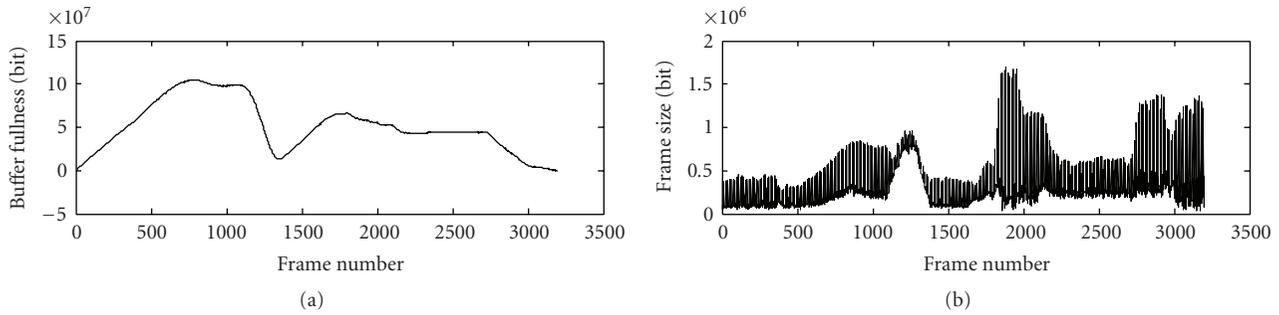


FIGURE 5: Buffer occupancy and frame size of “The Living Sea” video sequence encoded with a constant QP.

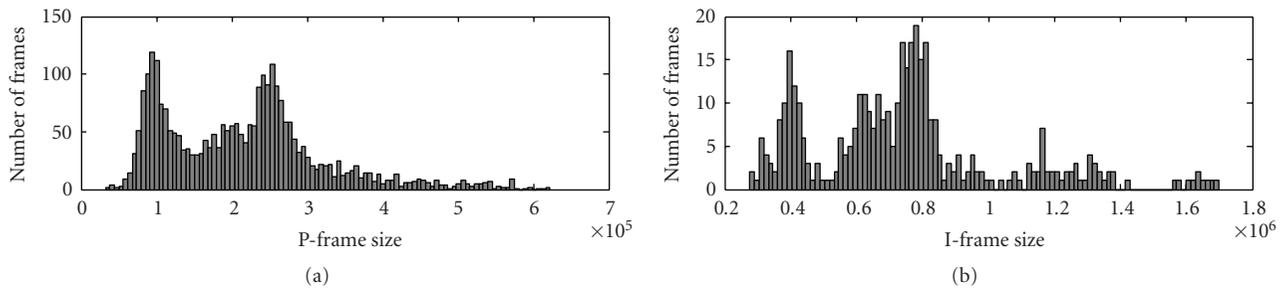


FIGURE 6: Histograms of P and I frame size in “The Living Sea” video sequence encoded with a constant QP.

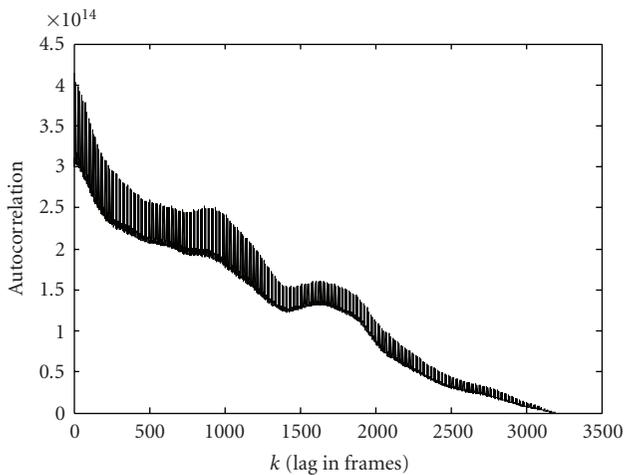


FIGURE 7: ACF of picture size in “The Living Sea” video sequence encoded with a constant QP.

with similar Hurst exponents have very different buffering requirements and also some bit streams with similar buffering requirements have very different Hurst exponents. Note that there is a tradeoff between buffering requirement and bandwidth in a communication network. Consequently, another important result is that statistical properties may not always reflect the practical parameters and thus previous models that rely only on capturing such statistical properties may not be accurate for estimating practical metrics of interest. The proposed model solves this problem by taking the practical parameters such as encoding parameters and

buffering constraints into consideration in the modeling approach.

The proposed traffic model is a modified version of our previous models that were validated successfully in [19, 20]. To validate the multi-Gamma video traffic model proposed in [20], we selected a set of known video sequences including *Foreman*, *Carphone*, *Silent*, *New York*, and *Football* sequences. We repeated and concatenated each of these sequences to provide longer sequences (900 frames) and then the resulting sequences were concatenated again to make a longer video sequence. The fact that the resulting video sequence has several different scenes was suitable for evaluating the model. The video sequence was encoded with a bit rate of 300 kb/s, a frame rate of 30 f/s, and a buffering delay of 0.4 second to produce a prototype video bit stream. The model parameters were extracted based on the prototype bit stream and a synthetic sequence was generated by the proposed model. The prototype and the synthetic traffics were compared by several measures including histogram, ACE, Hurst exponent, and buffering requirements. The simulation results presented in [20] show that the multi-Gamma model can generate synthetic bit streams close to the prototype real bit streams when they are parameterized according to the prototypes. The modifications of the model are related to the case in which the synthetic traffics are generated without the use of any prototype. Therefore, the validation results presented in [20] are not repeated in this paper and only the modified part of the model is validated. Generating the multi-Gamma model parameters is part of these modifications. The collected statistics from the real bit streams show that the Gamma distributions can be fitted to the shape and the scale parameters of the multi-Gamma

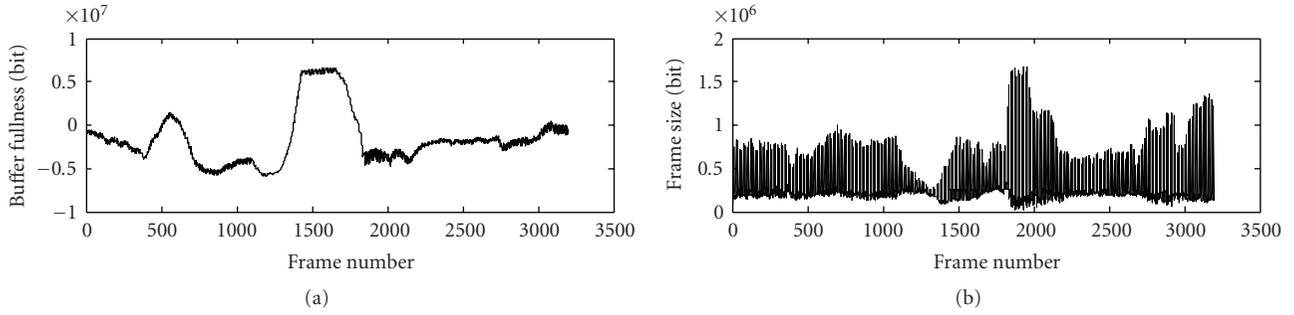


FIGURE 8: Buffer occupancy and frame size of “The Living Sea” video sequence encoded with FFMPEG VBR video rate controller.

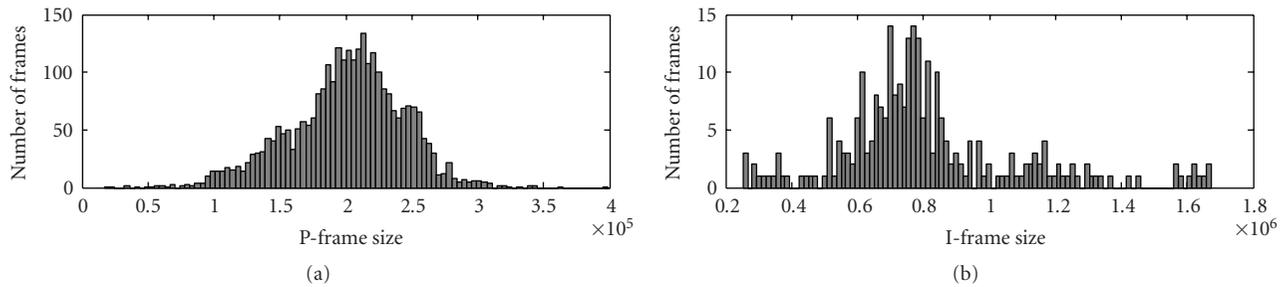


FIGURE 9: Histograms of P and I frame size in “The Living Sea” video sequence encoded by FFMPEG VBR video rate controller.

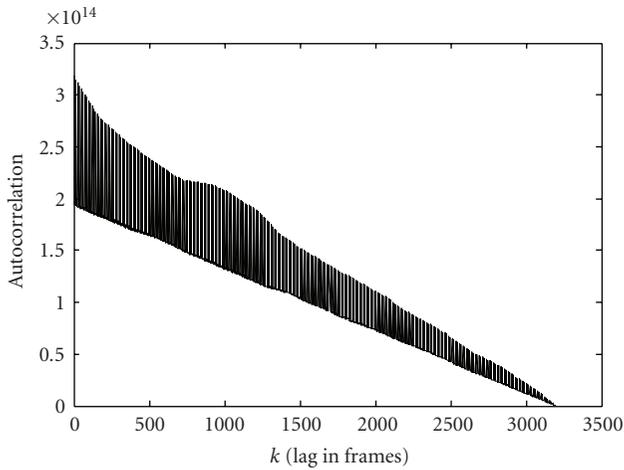


FIGURE 10: ACF of picture size in “The Living Sea” video sequence encoded with FFMPEG VBR video rate controller.

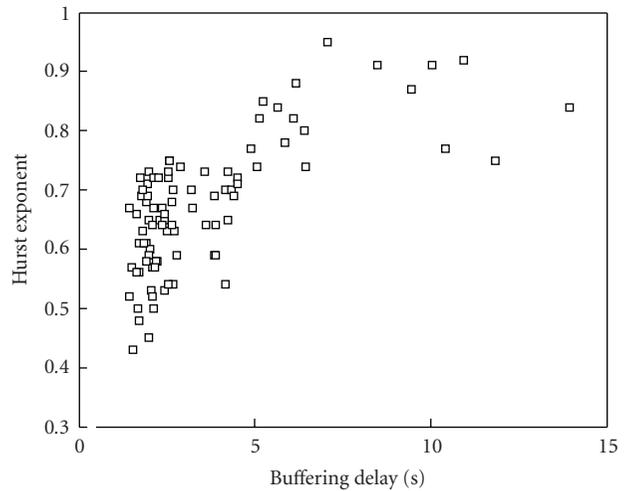


FIGURE 11: Hurst exponent versus buffering delay in real traffics.

model over the video scenes. Figure 2 shows the histogram of the shape parameter of the Gamma distributions of the P frames over the scenes (k_p) as a sample. As shown, a Gamma PDF is fitted to the histogram. Moreover, the statistics show that other Gamma distributions can be fitted to the relative complexities X_P and X_B over the video scenes. Figure 3 depicts the histogram of the relative complexity X_P over the scenes and also a Gamma PDF that is fitted to the histogram. The number of parameters that need to be determined for the multi-Gamma model is proportional to the number of

scenes in a video sequence. The above results are used to generate the parameters of the multi-Gamma model by only few other Gamma distributions each defined by only two parameters. In fact, we model the parameters of the multi-Gamma model to decrease the number of parameters that is required for generating synthetic traffic. Another part of the modification of the model is related to the range of operation that is validated below.

The model has been modified to generate bit streams with a wide range of statistics and practical metrics of

interest. To validate the proposed model in a wide range of operation, the model was parameterized to generate synthetic video bit streams with different buffering constraints. Buffer sizes corresponding to target maximum buffering delays of 1 to 15 seconds were used in the model. For each buffer size or target delay, a number of 20 bit streams, each including 3000 frames, with a bit rate of 6 MB/s were generated. Values of 100, 180, 46, 46, 3, and 4.3 were used for mean of L_s , k_I , k_P , k_B , X_P , and X_B , respectively, as user-defined parameters. Buffering simulations were performed on the bit streams and the minimum (over the frames) buffering delay for zero data drop rate was measured for each bit stream. The measured values have been compared with the target maximum buffering delay in Figure 12. As shown, the maximum (over 20 samples) delay obtained is close to the target maximum delay in different operating points or target delays. Moreover, delays obtained for 20 samples in each target delay have been distributed below and are close to the maximum values. This is very similar to real conditions in which the encoded bit streams by a rate controller may not use the whole available range of the buffer space. When generating the bit streams above, only the sizes of buffer S_B and k_B , were changed for different operating points and all other parameters were kept fixed. Simulation results show how well synthetic bit streams are in conformance with the desired practical constraints. Previous traffics models are usually validated by comparing the performance of real and modeled traffics in term of data drop rate in a buffering delay. In the simulation above, we consider the performance of modeled traffics in term of minimum delay for the zero data drop rate case which is a fixed practical reference point. This is beneficial from two points of view. First, when the model is used for simulation of StatMux in DVB-T2, we are interested in the zero drop rate case. Second, when in practice a video sequence is encoded, it is encoded with a buffering constraint for a zero data drop rate not for a target nonzero drop rate. However, the proposed model can be easily tuned by k_B , for a target nonzero data drop rate and a given delay. Therefore, the proposed model can be tuned similar to a video encoder and a video rate controller. This is a great advantage of the proposed model.

The above results show that the proposed model can provide practical metrics of interest for the synthetic traffics with a wide range of statistics. To assess the range of statistics of the metrics, for the generated bit streams explained above, the Hurst exponents were computed and are depicted in Figure 13. An approximate exponential function between the buffering delay and Hurst exponent can be considered over the results. This is in conformance with collected statistics from real bit streams depicted in Figure 11. Using the approximate exponential function, the model can be tuned to generate bit streams with a target Hurst exponent in the whole range. Note that our previous models are valid only for $H < 0.5$ while the new model is valid for $0 < H < 1$.

To evaluate the performance of StatMux over video broadcast services in DVB-T2, the proposed traffic model was parameterized to generate synthetic traffics corresponding to HDV contents with 3000 frames, 6 MB/s, and with a GOP structure as "I B P B P B P B P B P B".

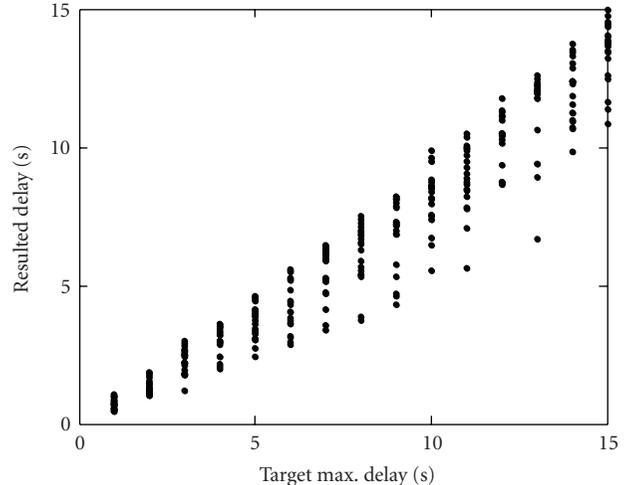


FIGURE 12: Resulted delay for modeled traffics versus target max delay.

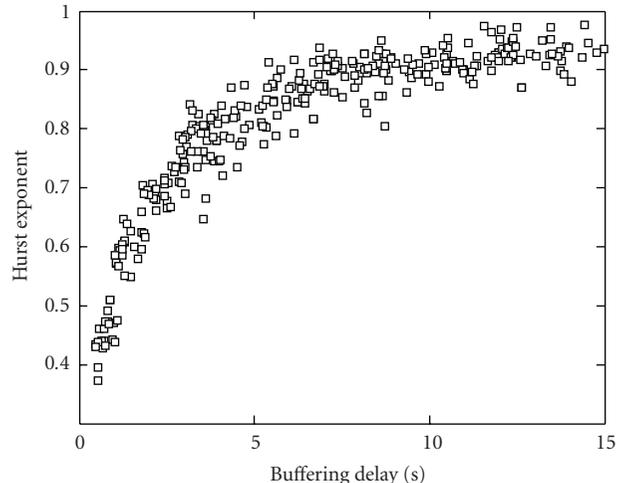


FIGURE 13: Hurst exponent versus buffering delay in modeled traffics.

TABLE 1: Gain of StatMux in different buffering delays when the bit streams are constrained to a buffering delay of 3 Seconds.

| Delay "ms" | Bandwidth utilization | | Gain in BW % |
|------------|-----------------------|----------------|--------------|
| | <i>DetMux</i> | <i>StatMux</i> | |
| 26 | 50.6 | 80.0 | 58.10 |
| 42 | 54.1 | 85.0 | 57.11 |
| 61 | 57.2 | 90.0 | 57.34 |
| 105 | 62.6 | 95.0 | 52.73 |
| 200 | 67.2 | 95.6 | 42.26 |

The model was tuned to generate traffics with different buffering constraints including 4, 5, and 7 seconds target buffering delays. Multiplexing simulations were performed over the synthetic video bit streams as explained in Section 4. 6 RF channels were considered in the simulations. The

TABLE 2: Gain of StatMux in different buffering delays when the bit streams are constrained to a buffering delay of 5 Seconds.

| Delay “ms” | Bandwidth utilization | | Gain in BW % |
|------------|-----------------------|----------------|--------------|
| | <i>DetMux</i> | <i>StatMux</i> | |
| 33 | 47.1 | 80.0 | 69.85 |
| 57 | 51.2 | 85.0 | 66.02 |
| 86 | 54.7 | 90.0 | 64.53 |
| 105 | 58.6 | 92.2 | 57.34 |
| 232 | 61.8 | 95.0 | 53.72 |

TABLE 3: Gain of StatMux in different buffering delays when the bit streams are constrained to a buffering delay of 7 Seconds.

| Delay “ms” | Bandwidth utilization | | Gain in BW % |
|------------|-----------------------|----------------|--------------|
| | <i>DetMux</i> | <i>StatMux</i> | |
| 44 | 43.1 | 80.0 | 85.61 |
| 86 | 46.7 | 85.0 | 82.01 |
| 170 | 50.6 | 90.0 | 77.87 |
| 250 | 52.9 | 91.2 | 72.40 |
| 501 | 58.0 | 95.0 | 63.79 |

performance of StatMux was compared with the performance of DetMux at several operating points in a two-dimensional space of bandwidth utilization and delay. For each operating point, two simulations corresponding to StatMux and DetMux were performed on a number of 14 to 20 bit streams. To provide different operating points, the number of multiplexed services has been changed from 14 to 20 while the transmission bandwidth was kept constant. To get statistically acceptable results, for each operating point, the simulations were repeated 5 times. The whole procedure above was repeated 3 times to get 3 performance curves corresponding to the bit streams with 3 different buffering constraints. The performance curves are depicted in Figures 14 and 15. The bandwidth utilization is depicted as a function of buffering delay in StatMux and DetMux for three different groups of video bit streams. The groups have different buffering constraints corresponding to 3, 5, 7 seconds. D3, D5, and D7 in the figures correspond to DetMux while S3, S5, and S7 correspond to StatMux. Figure 15 is a zoomed version of Figure 14 in a low-delay practical operating area. The high delay end points on the curves of DetMux in Figure 14 are very close to the target delays (3, 5, 7 seconds) used for generating traffics in the model. This closeness shows that the model performs accurately in different operating points. Moreover, it proves the accuracy of the multiplexing simulations. Sample results from the curves shown in Figure 15 are presented in Tables 1–3. Moreover, the gain of StatMux is presented for 5 operating points in the tables. The gain of StatMux was computed in term of percentage of bandwidth increase with respect to DetMux. According to Table 1, when the bit streams are constrained to a buffering delay of 3 seconds, for a buffering delay between 26 to 200 milliseconds, a gain of 42–58% increase in bandwidth is expected. Table 2 shows that when the bit streams are constrained to a buffering

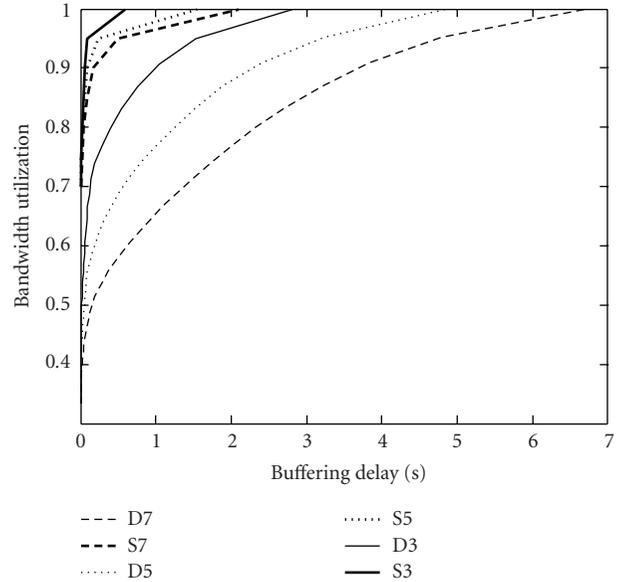


FIGURE 14: Bandwidth utilization versus buffering delay for StatMux and DetMux.

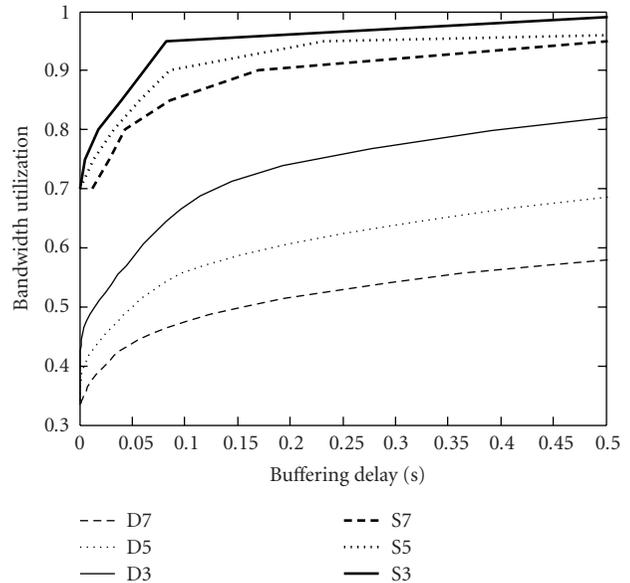


FIGURE 15: Bandwidth utilization versus buffering delay for StatMux and DetMux in low-delay operating area.

delay of 5 seconds, for a buffering delay between 33 to 232 milliseconds, a gain of 54–70% increase in bandwidth is expected. According to Table 3, when the bit streams are constrained to a buffering delay of 7 seconds, for a buffering delay between 44 to 501 milliseconds, a gain of 64–86% increase in bandwidth is expected. Simulation results show that using StatMux in DVB-T2 can considerably improve the bandwidth efficiency and end-to-end delay of a broadcast system.

From a video quality point of view, a typical buffering constraint about 5 seconds is large enough to allow a

quasiconstant quality for an encoded video. According to the results presented in Table 2 for such high-quality bit streams, a bandwidth efficiency of 95% can be achieved with only a buffering delay of 0.23 second by StatMux.

6. Conclusions

A model for variable bit rate video traffics was proposed that can generate a wide range of synthetic video bit streams with practical and statistical metrics of interest. The proposed model was validated successfully and was used to study the performance of statistical multiplexing of HDV services in a DVB-T2 broadcast system by computer simulations. Simulation results showed that the TFS introduced in DVB-T2 in conjunction with StatMux can provide a high performance in terms of bandwidth efficiency, end-to-end delay, and video quality for the broadcast system.

Acknowledgment

This work was partially supported by Nokia and the Academy of Finland, Project no. 213462 (Finnish Centre of Excellence program 2006–2011).

References

- [1] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger, "Long-range dependence in variable-bit-rate video traffic," *IEEE Transactions on Communications*, vol. 43, no. 234, pp. 1566–1579, 1995.
- [2] H. E. Hurst, R. P. Black, and Y. M. Simaika, *Long-Term Storage: An Experimental Study*, Constable, London, UK, 1965.
- [3] M. R. Izquierdo and D. S. Reeves, "Survey of statistical source models for variable-bit-rate compressed video," *Multimedia Systems*, vol. 7, no. 3, pp. 199–213, 1999.
- [4] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J. D. Robbins, "Performance models of statistical multiplexing in packet video communications," *IEEE Transactions on Communications*, vol. 36, no. 7, pp. 834–844, 1988.
- [5] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 2, no. 1, pp. 49–59, 1992.
- [6] D. M. Lucantoni, M. F. Neuts, and A. R. Reibman, "Methods for performance evaluation of VBR video traffic models," *IEEE/ACM Transactions on Networking*, vol. 2, no. 2, pp. 176–180, 1994.
- [7] R. Grunenfelder, J. P. Cosmas, S. Manthorpe, and A. Odinma-Okafor, "Characterization of video codecs as autoregressive moving average processes and related queueing system performance," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 3, pp. 284–293, 1991.
- [8] G. Ramamurthy and B. Sengupta, "Modeling and analysis of a variable bit rate video multiplexer," in *Proceedings of the 11th IEEE Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '92)*, vol. 2, pp. 817–827, Florence, Italy, May 1992.
- [9] D. P. Heyman and T. V. Lakshman, "Source models for VBR broadcast-video traffic," *IEEE/ACM Transactions on Networking*, vol. 4, no. 1, pp. 40–48, 1996.
- [10] B. Melamed, D. Raychaudhuri, B. Sengupta, and J. Zdepki, "TES-based video source modeling for performance evaluation of integrated networks," *IEEE Transactions on Communications*, vol. 42, no. 10, pp. 2773–2777, 1994.
- [11] A. A. Lazar, G. Pacifici, and D. E. Pendarakis, "Modeling video sources for real-time scheduling," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '93)*, vol. 2, pp. 835–839, Houston, Tex, USA, November 1993.
- [12] D. Reininger, B. Melamed, and D. Raychaudhuri, "Variable bit-rate MPEG video: characteristics, modeling and multiplexing," in *Proceedings of the 14th International Teletraffic Congress (ITC)*, pp. 295–306, Antibes Juan-les-Pins, France, June 1994.
- [13] M. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 4, pp. 269–280, 1994.
- [14] C. Huang, M. Devetsikiotis, I. Lambadaris, and R. Kaye, "Modeling and simulation of self-similar variable bit-rate compressed video: a unified approach," *ACM SIGCOMM Computer Communication Review*, vol. 25, no. 4, pp. 114–125, 1995.
- [15] M. Krunz and S. K. Tripathi, "On the characterization of VBR MPEG streams," in *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, vol. 25, pp. 192–202, Seattle, Wash, USA, June 1997.
- [16] D. Liu, E. I. Sára, and W. Sun, "Nested auto-regressive processes for MPEG-encoded video traffic modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 2, pp. 169–183, 2001.
- [17] U. K. Sarkar, S. Ramakrishnan, and D. Sarkar, "Modeling full-length video using Markov-modulated gamma-based framework," *IEEE/ACM Transactions on Networking*, vol. 11, no. 4, pp. 638–649, 2003.
- [18] M. Dai, D. Loguinov, and H. Radha, "A hybrid wavelet framework for modeling VBR video traffic," in *Proceedings of the International Conference on Image Processing (ICIP '04)*, vol. 5, pp. 3125–3128, Singapore, October 2004.
- [19] M. Rezaei, I. Bouazizi, and M. Gabbouj, "A model for controlled VBR video traffic," in *Proceedings of the IEEE International Conference on Signal Processing and Communications (ICSPC '07)*, Dubai, UAE, November 2007.
- [20] M. Rezaei, I. Bouazizi, and M. Gabbouj, "Generating antipersistent VBR video traffic," in *Proceedings of the Picture Coding Symposium (PCS '07)*, p. 6, Lisbon, Portugal, November 2007.
- [21] ISO/IEC JTC/SC29/WG11/N0400 MPEG93/457, Test Model 5, TM5, April 1993.
- [22] M. Rezaei, M. M. Hannuksela, and M. Gabbouj, "Semi-fuzzy rate controller for variable bit rate video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 5, pp. 633–644, 2008.
- [23] ETSI, "Generic Stream Encapsulation (GSE) Protocol," *ETSI standard*, DVD Document A116, May 2007.
- [24] ETSI, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)," *ETSI standard*, EN 302 307, V1.1.2, Jun 2006.
- [25] Microsoft, <http://www.microsoft.com/windows/windowsmedia/musicandvideo/hdvideo/contentshowcase.aspx>.
- [26] HD-Channel, <http://www.hd-channel.com>.
- [27] <http://www.highdefforum.com/showthread.php?t=6537>.
- [28] <http://ffmpeg.mplayerhq.hu>.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

