H.264 VIDEO TRAFFIC MODELING VIA HIDDEN MARKOV PROCESS

Stefania Colonnese, Stefano Rinauro, Lorenzo Rossi, and Gaetano Scarano

Dip. INFOCOM, "Sapienza" Universita´ di Roma
via Eudossiana 18 00184 Roma, Italy
(colonnese, rinauro, rossi, scarano)@infocom.uniroma1.it

ABSTRACT

The H.264 video coding standard includes new syntactic structures that allow efficient drift-free switching among precoded sequences at different bit-rates, making H.264 standard suitable for video streaming in time-varying channels. In network design, a good modeling of the video source is desirable to achieve a good dimensioning. A model of the traffic generated by a bitstream switching source requires considering not only variations of the video sequence activity, as it occurs in modeling classic VBR sources but also global variations of the average bit-rate. In this work we study a synthetic source constituted by a Hidden Markov Process modeling a real bit-rate switching video source. Parameter estimation is performed by the Expectation-Maximization algorithm. Model accuracy is assessed by a comparison of the frame loss rate of a fixed size buffer filled with the synthetic source and with a real H.264 video source.

1. INTRODUCTION

In these years, one of the emerging application which advances is the video streaming service. The video streaming service consists in a server which stores pre-encoded video sequences and transmits them to users. The video content is represented by two or more pre-encoded bitstreams, characterized by different encoding parameters, thus the server is able to choose the appropriate bitstream to transmit according to the network conditions. In fact, a video streaming service can be employed in networks that show a non-stationary behaviour, such as wireless networks that usually have a time varying structure, *e.g.* channel characteristics. The video server shall adapt the bitstream characteristics to the network behaviour by switching dinamically among pre-encoded bitstreams having different features, *e.g.* bandwith, delay, etc., but representing the same video content. Hence, the bitstreams should provide a periodical access point. An Intra (I) coded frame can be inserted periodically in order to provide this feature. Since I frames are larger than predicted (P) or bidirectional predicted (B) frames, their impact on the bitrate is considerable.

The most recent video coding standard, namely ITU-T Rec. H.264 or ISO/IEC MPEG-4/Part 10-AVC [1], includes new features that make attractive its employment in many contexts, including video streaming applications. In fact, H.264 introduces a new syntactic structure named Switching Picture (SP), which can be employed as a virtual access point instead of a I frame. Two kind of SP frames exists: the primary SP frame, which provides a virtual access point to the bitstream and is transmitted during ordinary streaming of an assigned video bitstream, and the secondary SP frame, that is transmitted only when bitstream switching occurs [2]. The primary SP frame permits to reconstruct exactly the frame using as reference frames the previous pictures of the same bitstream; the secondary SP frame uses as reference frames the pictures that are already available to the decoder, *i.e.*, the previous pictures of the bitstream that was transmitted before the switching. Since SP frames employ motocompensation to encode the picture, their impact on the bitstream is not dramatic as for I frames, making SP frames actractive for video streaming applications.

A good modeling of the video streaming statistical characteristics is crucial in network design. In fact, video traffic suffers errors, delay and jitter and is expensive in bandwidth allocation. During the stage of network design all of these requirements and characteristics must be taken into account in order to guarantee the negotiated quality of service.

In this work, we model the output of a H.264 video source representing a streaming application as the output of a Markovian random process. Such model may be a good tool in network design since it may be employed to replace the real source with a synthetic source that generates video traffic with similar statistical characteristics to a real one. By only varying the parameters of the synthetic source it is possible representing several classes of real video sources and generate synthetic traffic as much as needed. Moreover, the process parameter estimation can be performed only observing short segments of real video traffic.

In literature there are many works on modeling video source in broadcasting contexts. In [3], there's a summary of several models usually employed, such as Markoviand models, or TES (Transform Expand Sample) models. In [4], an MPEG1 video source is synthesized by a Markov chain, representing different video activities, and three AR processes representing a different kind of frame (I, P or B). First works on H.264 modeling are [5, 6, 7] which study models employing respectively wavelets, gamma distributions and Markov chains.

In the area of video streaming, only recently the literature has began to analyze the dynamical behavior of the H.264 source performing bitstream switching. A first preliminary approach can be found in [8], in which a Markov chain models the whole frame sequence by representing a frame as a state in the chain. In [9] the authors exploit the Group Of Pictures (GOP) structure of the video traffic by modeling the video source as a switching autoregressive hidden Markov process whose states represent different kind of GOPs. Instead, in [10] a low order autoregressive process is used to model the correlation between the frames whereas a Markov chain governs the global averages at the GOP layer.

The aim of this work consists in modeling a H.264 video source performing a bitstream switching and then creating a synthetic video traffic having similar statistical characteristics of a real one. We employ a Hidden Markov Process (HMP) to model the video source: each state of the Markov chain represent a different kind of GOP. The synthetic traffic is created by modulating mean and covariance matrix of a multivariate white Gaussian process according to the state. Parameter estimation is carried out in the likelihood sense by the Expectation-Maximization (EM) algorithm [12], by observing only a segment of real video traffic. First and second order moments are calculated. With respect to [10] parameter estimation is performed without any knowledge of the bitstream transition probability and observing only a part of a real video traffic.

The model is validated by a comparison of the buffer loss rate for the synthetic source and the real source, and by calculating the autocorrelation functions of the two sequences at the frame layer.

The remainder of this paper is organized as follows: in Section 2, we will describe the analyzed Markovian model; in Section 3 we will introduce the model validation based on a network point of view and the main simulation results; Section 4 concludes the paper.

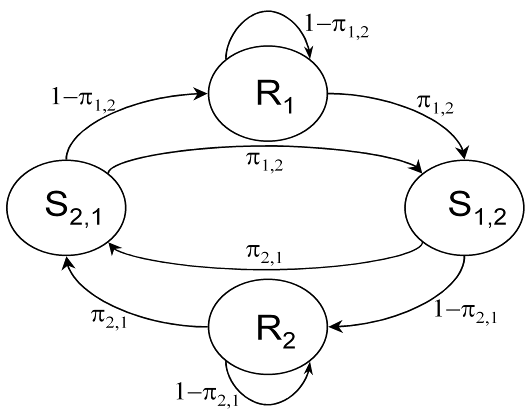

Figure 1: *H.264 video source Markov Model for L=2.*

2. MARKOV MODEL OF H.264 VIDEO SOURCE

2.1 Source characteristics

The video source transmits data extracted from L different bitstreams of the same video sequence, VBR encoded at different average bit-rate. A SP frame is inserted periodically to provide the virtual access point to the bitstream, and to allow, by sending a secondary SP frame, switching among the bitstreams. Hence, the GOP structure is constituted by a SP frame followed by $N_{GOP}-1$ non-switching frames. We denote with r_i the average bit-rate of the i-th bitstream, $i=1,\ldots,L$. Since during bitstream switching between rate r_i and rate r_j a secondary SP frame replaces the primary SP frame, each bitstream will present L different kind of GOPs. We denote the GOP with the primary SP frame with R_i and the GOPs with the secondary SP frame with S_{ji}, $i \neq j$, $i,j=1,\ldots,L$. Hence, the entire video source exhibits $N_s = L^2$ different kind of GOPs.

Due to the presence of secondary SP frames, GOPs can't be generated in arbitrary order. In fact, to switch from a bitstream to another, a switching GOP must be always generated, *i.e.* a R_i GOP can't never follow a R_j GOP. In other words, the kind of the GOP actually generated depends on the kind of the GOP previously generated.

Hence we model the GOP sequence as a first-order homogeneous Markov chain in which each state corresponds to a different kind of GOP. We denote the transition matrix, constituted by N_s^2 elements, with Π. The probability to switch from bit-rate r_i to bit-rate r_j is denoted with $\pi_{i,j}$. An example of the model is shown in Fig.1 with $L=2$, $N_s=4$.

In each state, the source emits a N_{GOP}-dimensional random variable representing the frame sizes of the GOP associated to the λ-th state, $\lambda = 1,\cdots,N_s$:

$$x[n] \stackrel{\text{def}}{=} [x_0[n],\cdots,x_{N_{GOP}-1}[n]]^{\mathrm{T}}$$

being $x_i[n]$ the size of the i-th frame of the n-th GOP of the coded video sequence.

2.2 The Hidden Markov Process

Hidden Markov Processes (HMP) are a well-known stochastic process family widely studied in the literature [11]. A HMP is a discrete-time finite-state homogeneous Markov chain observed through a discrete-time memoryless invariant channel. The state sequence is not observed directly at the destination. Various types of HMP are described in [11], where stationarity and ergodic conditions and algorithms for parameter estimation are also fully discussed.

Since $x[n]$, the observed variable, depends on the actual state of the Markov chain which models the GOP sequence and the GOP sequence is not directly observed, the proposed model is a HMP. By denoting the GOP sequence with $\{\lambda_n\}_{n=0}^{\infty}$, we model the random variable $x[n]$ as multivariate normal mixture whose averages are governed by the hidden Markov chain. Thus, $x[n]$ is written as:

$$x[n] = \mathbf{\Sigma}_\lambda e[n] + c_\lambda \tag{1}$$

where $e[n] = [e_0[n],\cdots,e_{N_{GOP}-1}[n]]^T$ is a standard normal random vector modeling the innovation of the sequence. The sequence $e[n]$ is supposed to be white, *i.e.* $E\{e[n]e[n-m]^{\mathrm{T}}\} = I\cdot\delta[m]$, being I the identity matrix and $\delta[m]$ the Kronecker delta. $\mathbf{\Sigma}_\lambda = \{\sigma_\lambda^i\}$, $i = 0,\ldots,N_{GOP}-1$ and $c_\lambda = [c_\lambda^0,\ldots,c_\lambda^{N_{GOP}-1}]^{\mathrm{T}}$, are respectively a matrix modeling the standard deviation of the frames in a GOP and the vector of the frames mean value. The Markov chain $\{\lambda\}_{n=0}^{\infty}$ and the innovation process $e[n]$ are supposed to be independent.

Ergodic conditions for the HMP reside only on the properties of the hidden Markov chain: if the chain is stationary, irreducible and aperiodic, the model is ergodic [11]. It is simply to show that the Markov chain modeling the GOP sequence has in fact the properties listed above.

Let us denote by $p_\lambda, \lambda = 1,\cdots,N_s$, the limit state probabilities of the Markov chain. The mean vector and the autocorrelation function for the variate $x[n]$ in (1) are defined as follows:

$$\begin{aligned} m_x &\stackrel{\text{def}}{=} E\{x[n]\} \\ \mathbb{R}_x[m] &\stackrel{\text{def}}{=} E\{x[n]x^T[n-m]\} \end{aligned} \tag{2}$$

and are proved to be:

$$m_x = \sum_{\lambda=1}^{N_s} p_\lambda c_\lambda \tag{3}$$

and:

$$\mathbb{R}_x[m] = \sum_{\lambda_1=1}^{N_s}\sum_{\lambda_2=1}^{N_s} p_{\lambda_1}\Pi^m(\lambda_1,\lambda_2)c_{\lambda_2}c_{\lambda_1}^T + \delta[m]\sum_{\lambda=1}^{N_s} p_\lambda \Sigma_\lambda \Sigma_\lambda^T. \tag{4}$$

2.3 Parameter Estimation

A crucial step in the modeling procedure is in estimating the correct parameters of the synthetic source in order to generate traffic statistically similar to the real video coded traffic. The parameter estimation is performed by observing a real video traffic and then extrapolating the parameters that permits to generate traffic similar, in the likelihood sense, to the observed traffic.

The Expectation-Maximization (EM) algorithm, originally developed by Dempster, Laird and Rubin [12] performs a local maximization of the log-likelihood function and it is frequently used in HMP parameter estimation. Let us denote $x_0^{N-1} \stackrel{\text{def}}{=} \{x[n]\}_{n=0}^{N-1}$ the observed video traffic and $\Theta \in \mathbf{\Theta}$ the model parameter, where $\mathbf{\Theta}$ is the parameter space. In detail, $\Theta = \{\Pi, \Sigma_1,\ldots,\Sigma_{N_s}, c_1,\ldots,c_{N_s}\}$. The EM algorithm iterates between two steps: the expectation step (E-step) which computes the auxiliary likelihood function $Q(\Theta,\Theta^{(k)}) = E\{\log(f(\Lambda,x|\Theta)|x,\Theta^{(k)}\}$ being $\Lambda \in \mathbf{\Lambda}$ a plausible state sequence and $\Theta^{(k)}$ the parameter estimated at the k-*th* step; the maximization step (M-step) which maximizes the Q function:

$$\Theta^{(k+1)} = \arg\max_{\Theta} Q(\Theta,\Theta^{(k)}). \tag{5}$$

Algorithm is stopped when the parameters quit changing, *i.e.* $\|\Theta^{(k)} - \Theta^{(k+1)}\| < \varepsilon$ for some ε and an appropriate measure distance.

The EM algorithm applied to HMP is illustrated in the following steps:

1. The probabilities $\alpha(\lambda|n) \stackrel{\text{def}}{=} P(\lambda_n = \lambda|x_0^n, \Theta^{(k)})$, are calculated iteratively through the following expression:

$$\alpha(\lambda|n) = \begin{cases} \frac{f_{\lambda 0}\pi_\lambda^{(k)}}{\sum_{\mu=1}^{N_s}\pi_\mu^{(k)} f_{\mu 0}} & n = 0 \\ \frac{\sum_{\mu=1}^{N_s}\alpha(\mu|n-1)\pi(\mu,\lambda)^{(k)} f_{\lambda n}}{\sum_{\mu=1}^{N_s} a(\mu|n-1) f_{\mu n}} & n > 0 \end{cases} \tag{6}$$

being

$$\begin{aligned} a(\mu|n-1) &\stackrel{\text{def}}{=} P(\lambda_n = \mu|x_0^{n-1}, \Theta^{(k)}) \\ &= \sum_{\delta=1}^{N_s} \pi(\delta,\mu)^{(k)}\alpha(\delta|n-1), \end{aligned}$$

and

$$f_{\lambda n} \stackrel{\text{def}}{=} f(x[n]|\lambda_n = \lambda, \Theta^{(k)}).$$

$\pi_\lambda^{(k)}$ and $\pi(\mu,\lambda)^{(k)}$ represent respectively the initial a-priori and the transition probabilities estimated ad the k-*th* step.

2. The a-posteriori probabilities

$$\begin{cases} \gamma_n(\lambda) \stackrel{\text{def}}{=} P(\lambda_n = \lambda|x_0^{N-1}, \Theta^{(k)}) \\ \xi_n(\mu,\lambda) \stackrel{\text{def}}{=} P(\lambda_n = \mu,\ \lambda_{n+1} = \lambda|x_0^{N-1}, \Theta^{(k)}) \end{cases}$$

are calculated through the backward iteration:

$$\begin{cases} \gamma_{N-1}(\lambda) = \alpha(\lambda|N-1) \\ \xi_n(\mu,\lambda) = \frac{\alpha(\mu|n)\pi(\mu,\lambda)^{(k)}}{a(\lambda|n)}\gamma_{n+1}(\lambda) \\ \gamma_n(\mu) = \sum_{\lambda=1}^{N_s}\xi_n(\mu,\lambda) \end{cases} \tag{7}$$

3. Finally the parameter $\Theta^{(k+1)}$ is calculated according to the following expressions:

$$\pi(\mu,\lambda)^{(k+1)} = \frac{\sum_{n=0}^{N-2}\xi_n(\mu,\lambda)}{\sum_{n=0}^{N-2}\gamma_n(\mu)} \tag{8}$$

$$c_\lambda^{(k+1)} = \frac{\sum_{n=0}^{N-1}\gamma_n(\lambda)x[n]}{\sum_{n=0}^{N-1}\gamma_n(\lambda)} \tag{9}$$

$$\sigma_\lambda^{i^{(k+1)}} = \sqrt{\frac{\sum_{n=0}^{N-1}\gamma_n(\lambda)(x_i[n] - c_\lambda^{i^{(k+1)}})^2}{\sum_{n=0}^{N-1}\gamma_n(\lambda)}}. \tag{10}$$

3. SIMULATIONS

In this Section we assess the goodness of the model, that is, how our source achieves the aim of generate traffic with the same characteristics of real video traffic. First, we evaluate the convergence of the EM algorithm by estimating the parameter of a HMP synthetic source. Next, following the approach in [4] for MPEG1 video sources, we compare the buffer load generated by the synthetic source and by a real video source, in terms of observed frame loss rate. Furthermore, the frame-level autocorrelation of the real video source is compared with the autocorrelation of the synthetic source. For the sake of simplicity, but without loss of generality, we will refer to the case of $L = 2$.

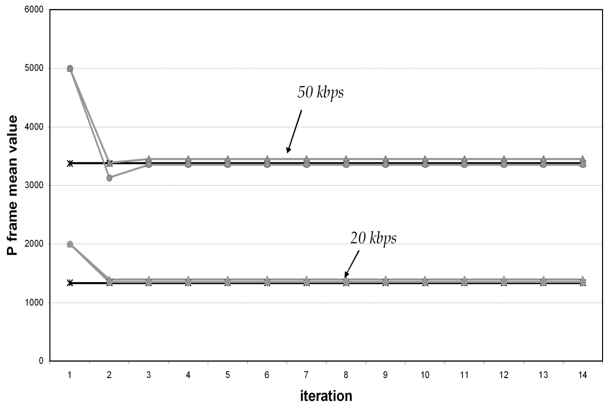

Figure 2: *P frame mean value: true value (black - star), estimation from non-switching state (gray - circle), estimation from switching state (gray - triangle).*

3.1 EM algorithm convergence

EM algorithm needs an initial estimation of the parameter to start the iterations: the choice of this initial estimation is crucial in order to reach the global maximum of the likelihood function [12]. To start sufficiently close to the global maximum, we perform an heuristic coarse estimation by exploiting the nominal bit-rate and the transition matrix structure. Specifically, we set the elements of the vector c_λ to the nominal frame size and the matrix Σ_λ to a diagonal matrix whose elements are the average of the nominal frame size:

$$c_\lambda^{i^{(0)}} = r_\lambda \cdot T_{GOP}/N_{GOP},\ i = 0,\ldots,N_{GOP}-1 \tag{11}$$

$$\sigma_\lambda^{i^{(0)}} = \frac{1}{L}\left(\sum_{\lambda=1}^{L} r_\lambda \cdot T_{GOP}/N_{GOP}\right),\ i = 0,\ldots,N_{GOP}-1 \tag{12}$$

where T_{GOP} is the GOP period. The transition matrix elements are set to the uniform distribution except for the transitions not allowed by the SP frame characteristics that are set to zero.

In the first test, we evaluate the goodness of the parameter estimation algorithm by generating video traffic from a synthetic source according to the model above and then estimating the parameters to verify the algorithm convergence. The synthetic source switches between $L = 2$ bitstreams where the frame averages are set to the same value for the same kind of frame of a fixed bitstream, *e.g.* all the P frames at the GOPs R_i and S_{ji} have the same mean value and standard deviation. The transition probabilities π_{12} and π_{21} are set to 0.4 and 0.7. The observed sequence is 25 GOPs long. We set the nominal bit-rates at 20 kbps and 50 kbps, since their closeness can put under stress EM algorithm in identifying the bitstreams. The algorithm estimates the entire c_λ vectors and Σ_λ matrices, so, to compare the estimation to the real values we average the estimations of mean values and standard deviations for all the frames of the same kind at each GOP. Figs.2-4 show the simulation results for the mean P and SP frame value and the transition probabilities: it's remarkable that after 3 iterations EM algorithm converges near the true parameter values, validating the parameter estimation for this kind of model. Moreover, P mean value estimated in the switching GOPs and in the non-switching GOPs are very similar.

After the assessment of the parameter estimation procedure, we finally compare a real video source with the synthetic source by first evaluating the frame loss rate on buffer load then comparing the autocorrelation function at the frame layer.

3.2 Comparison between a real source and the synthetic source

A real sequence is generated using the H.264 reference encoder JM v11.0 [13], extended profile. The Class A test sequence Bridge (far) in QCIF format is encoded at different bit-rates at ten frames per

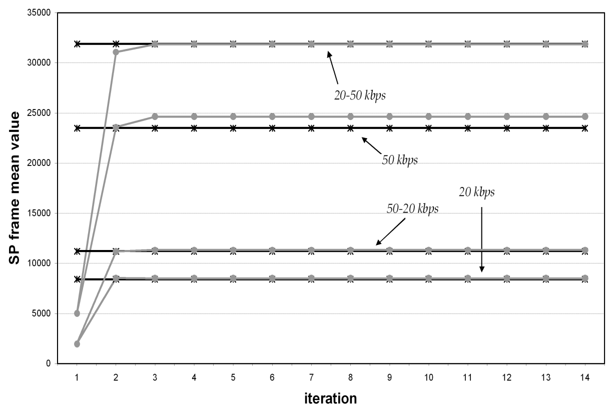

Figure 3: *SP frame mean value: true value (black - star), estimation (gray - circle).*

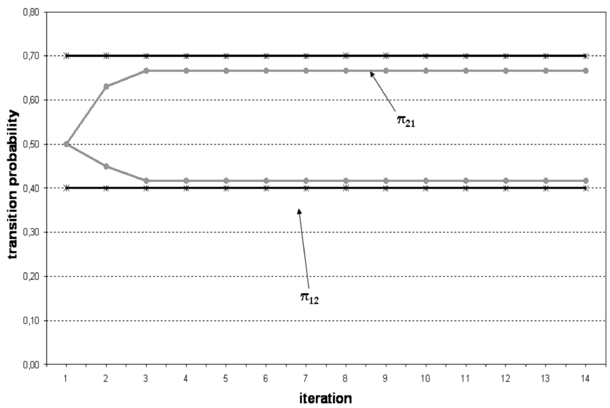

Figure 4: *Transition probability estimation: true value (black - star), estimation (gray - circle).*

second, $N_{GOP} = 10$, namely one SP picture is followed by nine P pictures. The sequence is 690 frames long, and parameter estimation is performed by observing only the first 25 GOP. Unlike from [8, 9, 10], the synthetic source is unaware of the transition probabilities so they must also be estimated. The bitstream switching probabilities $\pi_{1,2}$ and $\pi_{2,1}$ are respectively 0.4 and 0.7. The allowable bit-rates we considered are 20, 50 and 100, 320 kbps; the size of the buffer's cell is 384 bit (48 bytes being the size of an ATM cell without the overhead).

In order to assess the validity of the model we compare the frame loss rate of a finite-size buffer loaded with the real source with the frame loss rate of the same buffer loaded with the synthetic source. Frame loss events happen due to the finite size of the buffer; a synthetic source that shows similar statistical characteristics of a real one will exhibit also similar frame loss rate. The buffer output rate varies according to the initial and final average rates of the VBR source. The processes of buffer filling and depleting are shown in Fig.5. For what the depleting procedure is concerned the stepwise curve represents the written data and the straight lines the read data; the channel rate determines the straight lines slope. The results based on the described buffering schemes are reported in the following, together with the detailed simulation settings.

In Fig.6 the frame loss rates of the real source and the synthetic source are shown. Both get similar loss rates despite the transition probability estimation; in particular at low bit-rates the synthetic source loss rate is almost identical to the real source. Therefore we can assess that the model is able to reproduce a real video source characteristics in the sense of buffer allocation.

In Figs.7 and 8 the autocorrelation functions at bit-rates 20-50

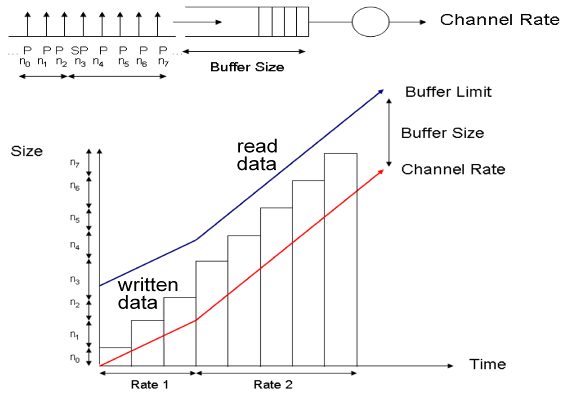

Figure 5: *Buffer filling and depleting in presence of bitstream switching.*

kbps and 100-320 kbps for the real and the synthetic source are shown for the two sets of transition probabilities. It's remarkable that the Markov model captures well the GOP periodicity by observing only a part of the real video sequence.

In order to test the model in presence of non stationarity we have concatenated the test sequence Akiyo, Bridge (close) and Bridge (far) and then we modelled this new sequence. Parameter estimation is performed by observing the whole sequence so as to include non stationarity not observed in the first GOPs of the sequence. The bitstream switching probabilities $\pi_{1,2}$ and $\pi_{2,1}$ are set to 0.4 and 0.7. Fig. 9 shows the frame loss rate of the two sources. We observe that the model achieves a good modeling (in the frame loss rate sense) also in presence of non stationarity.

4. CONCLUSION AND FURTHER WORK

The H.264 video coding standard introduces a compression tool for fast bitstream switching, based on the syntactic element Switching Pictures (SP). In this work a Hidden Markov Process is employed to model a H.264 source performing bit-rate adaptation using Switching Pictures. In the model each state represent the generation of an entire kind of GOP. Parameter estimation is performed by the Expectation-Maximization (EM) algorithm in order to find a local maximization of the likelihood function observing only a part of a real video sequence. We assessed the model performance at first by validating EM algorithm convergence through parameter estimation from a synthetic source which generates video traffic according to the model, then we compared a real video source to the synthetic source by examining their frame loss rate due to the transmission through a fixed size buffer and finally we compared the autocorrelation function of the two sequences. Numerical experiments show that the Markov GOP model provides a good approximation of the source behavior.

Future works will direct on the modeling of a more complex video sequence, exhibiting a larger amount of movement, scenes changes and including intra frames, by evaluating other Markov process, such as auto-regressive hidden Markov processes, to take into account interframe correlation. Moreover, an "on-line" parameter estimation in which the parameter is estimated iteratively at every new GOP observed will be studied, to manage non-stationarities of the video sequence due to, *e.g.*, a scene change.

REFERENCES

[1] T. Wiegand *et all.*, "Overview of the H.264 Video Coding Standard", *IEEE Trans. on Circ. and Sys. for Video Tech.*, Vol. 13, No. 7, pp. 560-576, July 2003.

[2] M. Karczewicz, R. Kureren, , "The SP and SI frames design for H.264/AVC", *IEEE Trans. on Circ. and Sys. for Video Tech.*, Vol. 13, No. 7, pp. 637-644 July 2003.

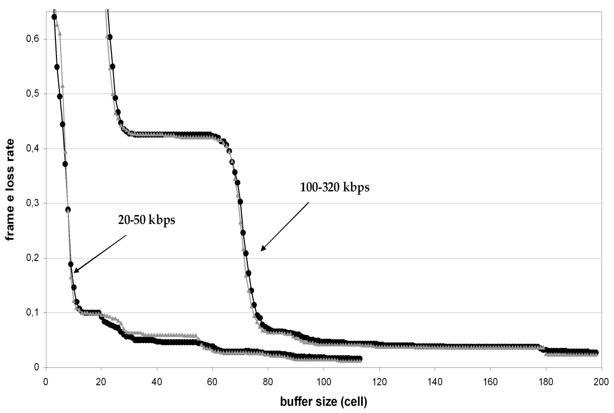

Figure 6: *Frame loss rate for the HMP (gray - triangle) and the real source Bridge (far) (black - circle),* $\pi_{12} = 0.4$, $\pi_{21} = 0.7$.

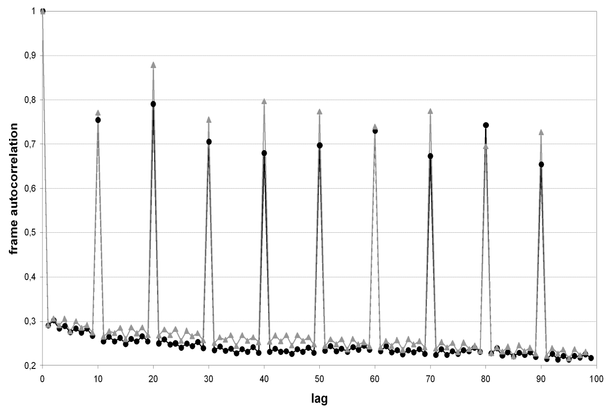

Figure 7: *Frame autocorrelation at 20-50 kbps for the HMP (gray - triangle) and the real source Bridge (far) (black - circle),* $\pi_{12} = 0.4$, $\pi_{21} = 0.7$.

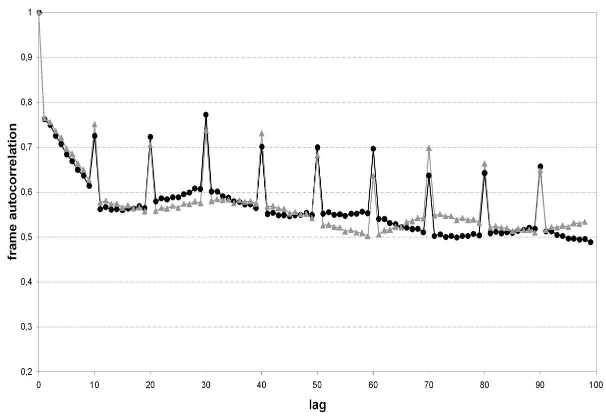

Figure 8: *Frame autocorrelation at 100-320 kbps for the HMP (gray - triangle) and the real source Bridge (far) (black - circle),* $\pi_{12} = 0.4$, $\pi_{21} = 0.7$.

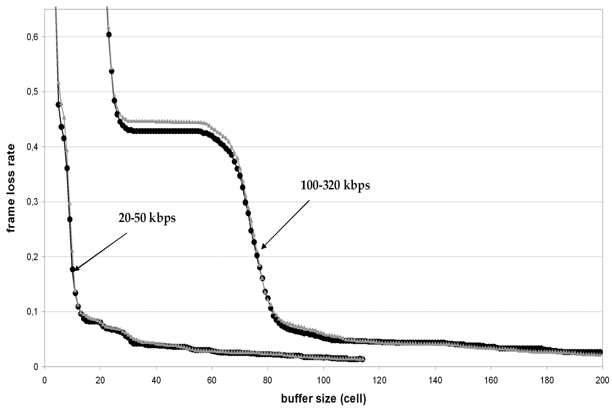

Figure 9: *Frame loss rate for the HMP (gray - triangle) and the composite source (black - circle),* $\pi_{12} = 0.4$, $\pi_{21} = 0.7$.

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

[4] N.D. Doulamis *et all.*, "Efficient modeling of VBR MPEG-1 coded video sources", *IEEE Trans. on Circ. and Sys. for Video Tech.*, Vol. 10, No. 1, pp. 93-112, February 2000.

[5] M. Dai, D. Louguinov, "Analysis and Modeling of MPEG-4 and H.264 Multi-Layer Video Traffic", INFOCOM-2005, Miami, Florida, U.S., March 13-17, 2005.

[6] H. Koumaras *et all.*, "A Markov Modified Model of H.264 VBR Video Traffic", IST Mobile Summit 2006, Mykonos, Greece.

[7] H. Koumaras *et all.*, "Analysis of H.264 Video Encoded Traffic", International Network Conference 2005, Samos, Greece.

[8] S. Colonnese, G. Panci, S. Rinauro, G. Scarano, "Modeling of H.264 Video Sources Performing Bitstream Switching", PCS-2007, Lisbon, Portugal, November 7-9, 2007.

[9] S. Colonnese, G. Panci, S. Rinauro, G. Scarano, "Markov Model of H.264 Video Sources Performing Bit-rate Switching", ICIP 2008, San Diego, USA, October 12-15, 2008.

[10] S. Colonnese, S. Rinauro, L. Rossi, G. Scarano, "Markov Model of H.264 Video Traffic",fat ISIVC 2008, Bilbao, Spain, July 9-11, 2008.

[11] Y. Ephraim and N. Merhav, "Hidden Markov processes", *IEEE Trans. Inform. Theory*, vol. 48, pp. 1518-1569, June 2002.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood From Incomplete Data via the EM Algorithm", *J. Roy. Statist. Soc. B*, vol. 39, no. 1, pp. 1-38, 1977.

[13] H.264 codec JM11.0 available at http://iphome.hhi.de/suehring/tml/