Source modeling for Distributed Video Coding

Velotiaray Toto-Zarasoa, Aline Roumy, Christine Guillemot

Source modeling for Distributed Video Coding

Velotiaray Toto-Zarasoa, *Member, IEEE,* Aline Roumy, *Member, IEEE*
and Christine Guillemot, *Senior Member, IEEE,*

***Abstract*—This paper studies source and correlation models for Distributed Video Coding (DVC). It first considers a two-states HMM, i.e. a Gilbert-Elliott process, to model the bit-planes produced by DVC schemes. A statistical analysis shows that this model allows us to accurately capture the memory present in the video bit-planes. The achievable rate bounds are derived for these ergodic sources, first assuming an additive binary symmetric correlation channel between the two sources. These bounds show that a rate gain can be achieved by exploiting the sources memory with the *additive* BSC model. A Slepian-Wolf (SW) decoding algorithm which jointly estimates the sources and the source model parameters is then described. Simulation results show that the additive correlation model does not always fit well with the correlation between the actual video bit-planes. This has led us to consider a second correlation model (the *predictive* model). The rate bounds are then derived for the predictive correlation model in the case of memory sources, showing that exploiting the source memory does not bring any rate gain and that the noise statistic is a sufficient statistic for the MAP decoder. We also evaluate the rate loss when the correlation model assumed by the decoder is not matched to the true one. An *a posteriori* estimation of the correlation channel has hence been added to the decoder in order to use the most appropriate correlation model for each bit-plane. The new decoding algorithm has been integrated in a DVC decoder, leading to a rate saving of up to 10.14% for the same PSNR, with respect to the case where the bit-planes are assumed to be memoryless uniform sources correlated with the SI via an additive channel model.**

Index terms — Distributed Source Coding, Hidden Markov Process, Parameter Estimation, Distributed Video Coding.

I. Introduction

Distributed Source Coding (DSC) refers to the problem of separate encoding and joint decoding of correlated sources. From the Shannon's theorem [1], the minimum achievable rate for the lossless compression of two statistically dependent memoryless sources X and Y is given by their joint entropy $H(X,Y)$. To approach this rate bound, traditional systems encode and decode the two sources jointly. In 1973, Slepian and Wolf (SW) have established [2] that, for two dependent binary sources X and Y, this lossless compression rate can be achieved by a separate encoding of the two sources, provided that the respective rates of X and Y are greater than their conditional entropies, $H(X|Y)$ and $H(Y|X)$, and joint decoding is performed. The lossy equivalent of the SW theorem for two correlated continuous-valued sources has been formulated by Wyner and Ziv (WZ) [3]. First practical DSC solutions were based on channel codes, e.g., convolutional codes, turbo codes [4], [5] or Low-Density-Parity-Check (LDPC) codes [6]. In the sequel, we consider a SW encoder based on LDPC codes, whose performance is very close to the SW bound [6], and whose decoding complexity is linear with the code length.

This research was partially supported by the French European Commission in the framework of the FP7 Network of Excellence in Wireless COMmunications NEWCOM++ (contract n.216715).

Video compression has been recast into a DSC framework, leading to the emergence of Distributed Video Coding (DVC) systems. The video sequence is structured into Groups Of Pictures (GOP) in which *key frames* are intra-coded and intermediate frames (also called "*WZ frames*") are WZ-coded. Each WZ frame is encoded independently of the other frames. The WZ data is transformed and quantized, then the quantized coefficients are binarized. Each resulting *bit-plane* is encoded by a channel encoder to yield syndrome bits. The DVC decoder constructs the side information (SI) via a motion-compensated interpolation of the previously decoded key frames. The encoder first sends a subset of syndrome bits for each bit-plane. If the decoder cannot properly decode the current bit-plane, more syndrome bits are requested from the encoder.

Most DVC practical systems assume the SW encoded bit-planes to be memoryless sources. Here, we instead model the video bit-planes as two-state Hidden Markov sources, with the help of the Gilbert-Elliott (GE) process. The probability of a given symbol is hence dependent only on the current state. This two-state model has already been considered to model a communication channel in [7], [8], [9] as well as the correlation channel in DSC [10], [11], [12], [13], [14], assuming the input sources to be uniform. However, this GE process is used here to model the sources and not the correlation channel. One can thus model a source with infinite memory with few parameters. Raptor codes [15] have been used in [16] for distributed coding of hidden Markov sources. However, the approach is validated with synthetic sources and the parameters of the source are assumed to be known to the decoder, which is not the case in practical DVC setup where the source parameters differ from bit-plane to bit-plane and must be estimated on-line. A statistical analysis of the burst lengths in the bit-planes has shown in [17] that the two-state GE model was reliably capturing the source memory; the correlation between the two sources was assumed to follow an additive model.

This paper first derives the achievable entropy-rate for ergodic correlated sources, when the correlation channel is additive, showing a theoretic rate gain with respect to memoryless sources. The paper then describes a joint estimation-decoding algorithm based on the *Expectation Maximization* (EM) algorithm [18], which jointly estimates the source parameters and the source symbols. When the model parameters need to be estimated by the decoder as in actual DVC systems, the initialization of the estimator of the source model parameters can make use of the knowledge of the SI, which is not the case when using this model for the correlation channel. We derive properties of the source model (Lemma 1) which allows us to efficiently initialize the EM algorithm. More precisely,

a Baum-Welch algorithm is performed in order to estimate the parameters and the state realizations of the SI bit planes, which are then used to initialize the parameters and the state realizations of the SW-encoded source.

The first simulation results have shown that, although the GE model was well capturing the bit-planes memory, exploiting the source memory while decoding the bit-planes in the DVC decoder was not always improving the rate-distortion (RD) performances of the system. This is explained by the fact that the *additive* correlation model does not always fit well with the correlation between the actual video bit-planes. This has led us to then consider another correlation model, the *predictive* correlation model together with HMM memory sources. Achievable entropy-rates have thus been also derived for ergodic correlated sources, when the correlation channel is *predictive*. It is shown, similarly to the case of non-uniform Bernoulli sources, in [19], that, if the correlation channel is additive, then the compression rate can be reduced by exploiting the source memory. However, for the predictive correlation model, exploiting the source memory does not reduce the compression rate. Actually, we show that the noise statistics are sufficient for the MAP estimation in the predictive case and that the source statistics do not play any role in the performance bounds. We further study the impact on the DVC decoding performance of a possible mismatch between the assumed correlation model and the actual one. One extra step, i.e., an *a posteriori* estimation of the type of correlation channel, has thus been added to the SW decoder. These results extend those presented in [19] obtained for an *i.i.d* binary source model. We show that the joint source estimation-decoding algorithm together with the estimation of the channel correlation type (predictive or additive) per bit-plane leads to significant rate saving compared with classical DVC schemes.

This paper is structured as follows. Section II, after reviewing the theory behind DSC, presents the GE source model, and defines its parameters. Section III describes the LDPC-based *estimation-decoding* EM algorithm which takes into account the statistics of the source. Its performance is first assessed with synthetic sources. Section IV demonstrates the accuracy of the GE source model for DVC bit-planes, and presents how the EM algorithm can be used in a DVC codec. Finally, Section V introduces the predictive correlation model, the corresponding rate bounds with ergodic sources as well as the way the decoder operates to estimate the correlation model to be used in the decoding process. The RD performances obtained with the modified DVC codec are then presented by exploiting the complete HMM model as well as a simplified version which amounts to exploiting only the non-uniformity of the sources.

II. Distributed Source coding of correlated binary hidden Markov sources

This Section first revisits the principle of asymmetric DSC with the classical binary symmetric correlation model, referred to as *additive* BSC. The rate bounds are then derived in the case where the correlated sources have infinite memory.

In the sequel, a source is modeled as a random process. Uppercase sequences $\{X_n\}_{n\geq 1}$, in short $(\{X_n\},\{Y_n\},\{Z_n\},\{\Sigma_n\})$, refer to random stochastic processes, uppercase variables $(X_n, Y_n, Z_n, \Sigma_n)$ refer to random variables at instant n, and lowercase variables $(x_n, y_n, z_n, \sigma_n)$ refer to their realizations; bold uppercase variables $(\mathbf{X} = X_1^N, \mathbf{Y} = Y_1^N, \mathbf{Z} = Z_1^N, \mathbf{\Sigma}_{=}\Sigma_1^N)$ refer to vectors of random variables, and bold lowercase variables $(\mathbf{x} = x_1^N, \mathbf{y} = y_1^N, \mathbf{z} = z_1^N, \sigma = \sigma_1^N)$ refer to vectors of their realizations. By abuse of notation, uppercase variables (X, Y, Z, Σ) refers to i.i.d. processes. The symbol "$\oplus$" stands for the modulo-two addition of binary symbols. The bold $\mathbf{H}$ stands for the parity-check matrix of channel code. Finally, $H(X)$ stands for the entropy of the i.i.d. source, and $H(\mathcal{X})$ [20] stands for the entropy-rate of the ergodic source.

A. Asymmetric DSC: source coding with side information at the decoder only

DSC refers to the separate compression of two correlated sources assuming the decoding is performed jointly. In the asymmetric DSC setup, one of the sources (say $\{X_n\}$) has to be compressed, while the other (say $\{Y_n\}$) is available at the decoder. $\{Y_n\}$ therefore serves as a *side-information* (SI) for the decoding of $\{X_n\}$. When the sources $\{X_n\}$ and $\{Y_n\}$ are i.i.d. with finite alphabet, [2] shows that the source $\{X_n\}$ can be compressed at the conditional entropy: $H(X|Y)$, such that knowing the SI at the encoder does not help reducing the compression rate. In the case of ergodic random processes $\{X_n\}$ and $\{Y_n\}$ with discrete (in the sense of infinite countably) alphabets, [21] shows that the minimum compression rate for the source $\{X_n\}$ is the conditional entropy-rate $H(\mathcal{X}|\mathcal{Y})$.

Let us now consider two binary correlated sources: $\{X_n\},\{Y_n\}$. The correlation can be modeled by a virtual channel as defined below.

Definition 1. *An (X, Y, p) additive BSC is a channel with binary ergodic input $\{X_n\}$, binary ergodic output $\{Y_n\}$. The noise is an i.i.d. binary process $Z \sim \mathcal{B}(p)$. Z is independent of the channel input, and the channel output is obtained by $Y_n = X_n \oplus Z_n, \forall n$.*

If the correlated binary sources $\{X_n\},\{Y_n\}$ are i.i.d., [22] shows that the compression rate for $\{X_n\}$, $H(X|Y)$ [2], can be achieved by use of channel codes. This scheme is called *the syndrome approach*. More precisely, the (N, K) linear code $\mathcal{C}$, defined by its $(N-K)\times N$ parity check matrix $\mathbf{H}$, defines a partition of the N-long source sequences into cosets, where all the sequences in the same coset $\mathcal{C}_\mathbf{s}$ share the same syndrome $\mathbf{s}$, i.e. $\mathcal{C}_\mathbf{s} = \{\mathbf{x} : \mathbf{Hx} = \mathbf{s}\}$. To encode a particular vector $\mathbf{x}$, the encoder transmits its syndrome $\mathbf{s_x} = \mathbf{Hx}$, achieving a compression ratio of $N : (N-K)$; the side-information $\mathbf{y}$ is available at the decoder. The decoder's estimation $\hat{\mathbf{x}}$ consists in finding the closest sequence to $\mathbf{y}$ having syndrome $\mathbf{s_x}$. [22] shows that if the code defined by the parity check matrix $\mathbf{H}$ achieves the capacity of the (X, Y, p) additive BSC, then the syndrome approach with the same code achieves the compression rate for $\{X_n\}$ i.e. $lim_{N\to+\infty}(N-K)/N = H(X|Y)$.

B. Binary memory sources: Gilbert-Elliott modeling

We now consider binary sources with memory, modeled as a two-state Hidden Markov Model (HMM) (the Gilbert Elliott (GE) process). We will see in the sequel that the GE process accurately models the video bit-planes which are SW-encoded in a practical DVC system. This model is well suited to our problem since it builds infinite memory sources with very few parameters. This section states a property which will be useful in the model parameters estimation to be performed by the decoder.

Let $\{\Sigma_n\}$ be a finite Markovian process with memory of order one, having two realizations s and d, where s stands for *"sparse source"*, and d stands for *"dense source"*. The GE process $\{X_n\}$ is a binary source which depends on the Markov process $\{\Sigma_n\}$. More precisely, X_n only depends on Σ_n. Therefore, Σ_n represents the hidden state of the random process $\{X_n\}$. In each state s and d, the source is drawn according to a Bernoulli law of parameter p_s and p_d respectively (Fig 1), with $p_s \leq p_d$ by definition. p_s and p_d respectively stand for the probability of having $X_n = 1$ in states s and d.

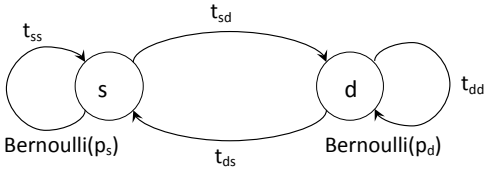

Fig. 1. The GE model: a two-state HMM.

We define the transition probabilities t_{ss}, t_{sd}, t_{ds} and t_{dd}, between the states, as shown in Fig. 1. Since $t_{ss} = (1 - t_{sd})$ and $t_{dd} = (1 - t_{ds})$, the set of parameters of the model is $\theta_X = (p_s, p_d, t_{ds}, t_{sd})$. Those parameters are defined by:

$$\begin{aligned} p_s &= \mathbb{P}_{\theta_X}(X_n = 1 | \Sigma_n = s) \\ p_d &= \mathbb{P}_{\theta_X}(X_n = 1 | \Sigma_n = d) \\ t_{ds} &= \mathbb{P}_{\theta_X}(\Sigma_n = s | \Sigma_{n-1} = d) \\ t_{sd} &= \mathbb{P}_{\theta_X}(\Sigma_n = d | \Sigma_{n-1} = s) \end{aligned} \tag{1}$$

where $\mathbf{\Sigma} = \Sigma_1^N$ is the underlying N-long state sequence, and $\sigma = \sigma_1^N \in \{s, d\}^N$ is its realization.

These equations lead to the following property of the source:

Property 1. *$\forall n \in [1, N]$, given the state Σ_n, $\{X_n\}$ is a memoryless Bernoulli process of parameter $p_{X_n} = \mathbb{P}_{\theta_X}(X_n = 1 | \sigma_n)$, where $p_{X_n} = p_s$ if $\sigma_n = s$, and $p_{X_n} = p_d$ if $\sigma_n = d$.*

C. Gilbert-Elliott modeling in DSC: properties and rate bounds

We now turn back to the DSC problem. Let $\{X_n\}$ be a GE source and let a second source $\{Y_n\}$ be correlated to $\{X_n\}$, where the correlation is modeled as a virtual additive BSC (X, Y, p) (see Def. 1). We assume that $\forall n, Y_n = X_n \oplus Z_n$, with $\mathbb{P}(Y_n \neq X_n) = \mathbb{P}(Z_n = 1) = p$. Given the characteristics of the correlation model, we state the following Lemma 1 which characterizes the GE nature of the side-information $\{Y_n\}$.

Lemma 1. *Let $\{X_n\}$ be a GE process of parameter $\theta_X = (p_s, p_d, t_{ds}, t_{sd})$. $\{\Sigma_n\}$ denotes the underlying hidden state process. Let $\{Y_n\}$ be a source correlated to $\{X_n\}$ according to the additive channel model (X, Y, p). Therefore, there exists an i.i.d. Bernoulli process Z independent of $\{X_n\}$ s.t. $\forall n, Y_n = X_n \oplus Z_n$, and $\{Y_n\}$ is a GE source with the same underlying state process $\{\Sigma_n\}$ as $\{X_n\}$.*

Proof: When conditioned on the state process $\{\Sigma_n\}$, the source $\{X_n\}$ is memoryless according to Property 1. Since Z is memoryless too, the source $\{Y_n\}$ is memoryless when conditioned on the same state process $\{\Sigma_n\}$. Moreover, $\forall n$, given the state Σ_n, $\{Y_n\}$ is a (memoryless) Bernoulli process of parameter $p_{Y_n} = p_{X_n}(1 - p) + (1 - p_{X_n})p$, where p_{Xn} is described in Property 1. Therefore $\{Y_n\}$ is a Gilbert Elliott source with the same state process as for $\{X_n\}$. ■

We now turn back to the asymmetric DSC problem, where $\{Y_n\}$ is available at the decoder only, and $\{X_n\}$ is transmitted at a rate greater than its conditional entropy-rate $H(\mathcal{X}|\mathcal{Y})$. It has been shown in [19] that, if the source $\{X_n\}$ is memoryless but not uniform, the minimum coding rate for $\{X_n\}$ can be reduced by an amount of $H(Y) - H(X) \geq 0$ with respect to the uniform memoryless case. In the following Lemma 2, we extend this result to the case of memory sources.

Lemma 2. *Let $\{X_n\}$ and $\{Y_n\}$ be two correlated binary ergodic sources, where the correlation is modeled as a virtual (X, Y, p) additive BSC. We consider the asymmetric DSC problem, where $\{Y_n\}$ is available at the decoder and $\{X_n\}$ is transmitted at a rate greater than its conditional entropy-rate $H(\mathcal{X}|\mathcal{Y})$.*

The minimum coding rate for $\{X_n\}$ is $H(\mathcal{X}|\mathcal{Y}) = H(Z) - [H(\mathcal{Y}) - H(\mathcal{X})]$. If the source $\{X_n\}$ is uniform (and therefore memoryless), this rate $H(\mathcal{X}|\mathcal{Y}) = H(Z)$. Thus, since $H(\mathcal{Y}) - H(\mathcal{X}) \geq 0$, $H(\mathcal{X}|\mathcal{Y})$ is reduced by $H(\mathcal{Y}) - H(\mathcal{X})$ compared to the minimum coding rate $H(\mathcal{X}|\mathcal{Y}) = H(Z)$ for a uniform source.

Proof: Since the BSC is additive, $\exists$ Z an i.i.d. binary process (of parameter p), independent of $\{X_n\}$ s.t. $\forall n, Y_n = X_n \oplus Z_n$. The conditional entropy-rate of the source $\{X_n\}$ is computed as:

$$\begin{aligned} H(\mathcal{X}|\mathcal{Y}) &= H(\mathcal{X}, \mathcal{Y}) - H(\mathcal{Y}) \\ &= H(\mathcal{X}) + H(\mathcal{Y}|\mathcal{X}) - H(\mathcal{Y}) \\ &= H(Z) - [H(\mathcal{Y}) - H(\mathcal{X})] \end{aligned} \tag{2}$$

where the last equality follows from the independence between $\{X_n\}$ and Z.

If $\{X_n\}$ is uniform, so is $\{Y_n\}$, and $H(\mathcal{Y}) = H(\mathcal{X}) = 0.5$ i.e. $H(\mathcal{X}|\mathcal{Y}) = H(Z)$. Now, consider the case of an ergodic process $\{X_n\}$. On the one hand we have $H(\mathcal{X}) = H(\mathcal{X} \oplus Z|Z)$, and on the other hand we have $H(\mathcal{Y}) = H(\mathcal{X} \oplus Z)$. Since conditioning reduces entropy [20, Theorem 2.6.5], it implies $H(\mathcal{X}) \leq H(\mathcal{Y})$ with equality *if, and only if,* $\{Y_n\}$ and Z are independent i.e. if $p = 0$ or 1, or if the source $\{X_n\}$ is uniform.

As $H(\mathcal{Y}) - H(\mathcal{X}) \geq 0$ (with equality *if, and only if,* the source $\{X_n\}$ is uniform or in the degenerate case $p = 0$ or 1), the lower transmission rate bound of a GE source is lower

than that of a uniform source, and the rate gain is given by $H(\mathcal{Y}) - H(\mathcal{X})$. ■

The dotted curve in Fig. 2 shows the theoretical rates $H(\mathcal{X}|\mathcal{Y})$ that can be achieved for a GE source with parameter $\theta_X = (t_{ds} = 0.03, t_{sd} = 0.01, p_s = 0.07, p_d = 0.7)$, when p varies in $[0, 1]$, and the correlation is modeled as a virtual additive BSC. If instead the same source is modeled as a memoryless source (with parameter $\frac{p_s t_{ds} + p_d t_{sd}}{t_{ds} + t_{sd}}$), the minimum achievable rate (solid line) is greater than the one with the GE modeling (dotted curve). Therefore, knowing the GE source distribution, one can achieve smaller compression rates.

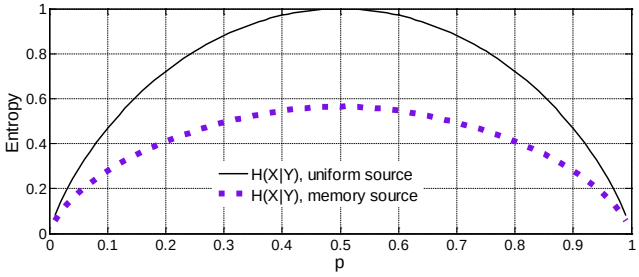

Fig. 2. Comparison of the source conditional entropies, when X is uniform, and when X is a GE process.

III. Joint estimation-decoding for distributed coding of GE sources: the EM algorithm

This section describes the algorithm used to jointly estimate the source $\{X_n\}$ and its model parameters. The approach is based on the Expectation-Maximization (EM) algorithm. The EM algorithm has already been used in [10], [7], [12] [9], and [23] to estimate the HMM parameters of a communication channel or of the correlation channel in DSC problems. Instead, here the EM algorithm is used to estimate the HMM parameters of the source; the initialization of the estimator relies on the source properties stated in Lemma 1. A Baum-Welch algorithm is performed in order to estimate the parameters and the state realizations of the SI, which are then used to initialize the parameters and the state realizations of the source $\{X_n\}$.

A. Formalization of the maximization problem

The SW decoder estimates the source realization $\mathbf{x}$ from its syndrome $\mathbf{s_x}$ and the side-information $\mathbf{y}$, assuming that $\{X_n\}$ is a GE source. However, the decoder is not aware neither of the parameter θ_X nor of the hidden state sequence $\{\Sigma_n\}$. This estimation is carried out by an *Expectation-Maximization* (EM) algorithm, which is an optimization procedure that learns new parameters from the observed variables ($\mathbf{y}$ and $\mathbf{s_x}$), and a previously estimated set of hidden variables ($\hat{\mathbf{x}}$, $\hat{\sigma}_{\mathbf{x}}$ and $\hat{\theta}_X$).

Let l be the current decoding iteration, and θ_X^l the current estimate of the GE source parameters. Then, the updated value $\theta_X^{(l+1)}$ for the next iteration is computed so as to maximize the mean log-likelihood function:

$$\theta_X^{(l+1)} = \arg\max_{\theta_X} \left(\mathbb{E}_{\mathbf{x},\mathbf{\Sigma_x}|\mathbf{Y},s_\mathbf{x},\theta_X^l} \left[\log \left(\mathbb{P}_{\theta_X}(\mathbf{y}, \mathbf{x}, \sigma_\mathbf{x}, \mathbf{s_x}) \right) \right] \right) \tag{3}$$

To simplify the notation, in the sequel we denote "$\mathbb{P}(\mathbf{X} = \mathbf{x}|\theta_X)$" by "$\mathbb{P}_{\theta_X}(\mathbf{x})$".

Since the logarithm is a strictly increasing function, the value $\theta_X^{(l+1)}$ that maximizes $\mathbb{P}_{\theta_X^l}(\mathbf{y}, \mathbf{x}, \sigma_\mathbf{x}, \mathbf{s_x})$ also maximizes $\log\left(\mathbb{P}_{\theta_X^l}(\mathbf{y}, \mathbf{x}, \sigma_\mathbf{x}, \mathbf{s_x})\right)$. The algorithm converges since it increases the likelihood at each iteration [24].

We now consider that the code used to compress the source X is a syndrome-based LDPC code. LDPC codes can be equivalently represented by their sparse parity-check matrix or by their factor graph. For an LDPC code yielding a compression rate $N : (N-K)$, let $\mathbf{H} = (h_{mn})_{m\in[1,(N-K)],n\in[1,N]}$ be the sparse matrix of size $(N-K) \times N$. Fig. 3 presents the factor graph [25] that describes the dependencies between the observed and the hidden variables of the problem.

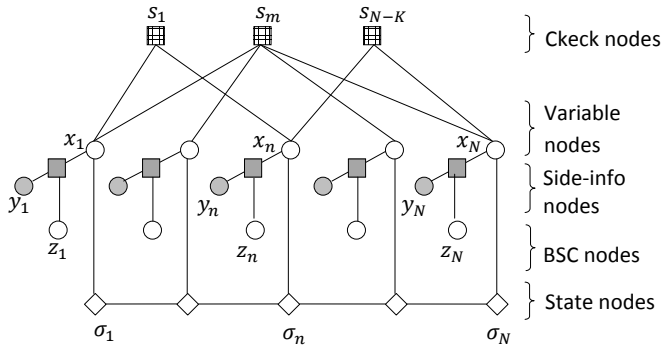

Fig. 3. Factor graph describing the joint estimation-decoding EM.

We introduce the following notation for the variables and the messages that are passed on the factor graph during the estimation-decoding process:

- x_n are the source symbols, represented by the *variable nodes*; their estimates are denoted $\hat{x}_n$;
- y_n are the side-information symbols, represented by the *side-information nodes*;
- z_n are the noise symbols, represented by the *BSC nodes*;
- s_m are the syndrome symbols, represented by the *check nodes*. x_n is connected to s_m in the bipartite graph if $h_{mn} = 1$ in the parity-check matrix;
- d_{xn} is the *degree* of x_n, i.e. the number of check nodes connected to it;
- d_{sm} is the *degree* of s_m, i.e. the number of variable nodes connected to it;
- I_n is the intrinsic information for the node z_n;
- $E_{n,e}, e \in [1, d_{xn}]$ are the messages passed from the variable nodes, on their e-th edge, to the check nodes;
- E_n is the *a posteriori* Log-Likelihood Ratio of $\hat{x}_n$;
- $Q_{m,e}, e \in [1, d_{sm}]$ are the messages passed from the check nodes, on their e-th edge, to the variable nodes;
- B_n are the messages passed from the BSC node z_n to the variable node x_n;
- S_n are the messages passed from the state nodes to the variable nodes;

- V_n are the messages passed from the variable nodes to the state nodes.

All the messages are Log-Likelihood Ratio (LLR), they are labeled (in) or (out) if they come *to* or *from* the considered node. In the following, we give the update rules for the messages that are passed on the graph.

B. Expectation step: computation of the mean log-likelihood function

First, we expand the likelihood function using the Bayes rule:

$$\mathbb{P}_{\theta_X^l}(\mathbf{y}, \mathbf{x}, \sigma_{\mathbf{x}}, \mathbf{s}_{\mathbf{x}}) = \mathbb{P}(\mathbf{y}, \mathbf{s}_{\mathbf{x}} | \mathbf{x}, \sigma_{\mathbf{x}}) \mathbb{P}_{\theta_X^l}(\mathbf{x}, \sigma_{\mathbf{x}}) \tag{4}$$

where $\mathbb{P}(\mathbf{y}, \mathbf{s}_{\mathbf{x}} | \mathbf{x}, \sigma_{\mathbf{x}})$ is independent of θ_X^l.

Then, the log-likelihood function reduces to:

$$\begin{aligned} &\log\left(\mathbb{P}_{\theta_X^l}(\mathbf{x}, \sigma_{\mathbf{x}})\right) = \log\left(\mathbb{P}_{\theta_X^l}(\sigma_1)\right) \\ &+ \sum_{n=2}^{N} \sum_{i=s}^{d} \sum_{j=s}^{d} \delta_{\sigma_{n-1}=i, \sigma_n=j} \log(t_{ij}^l) \\ &+ \sum_{n=1}^{N} \sum_{i=s}^{d} \delta_{\sigma_n=i, x_n=1} \log(p_i^l) + \delta_{\sigma_n=i, x_n=0} \log(1-p_i^l) \end{aligned} \tag{5}$$

where $\delta_{bool} = \begin{cases} 1, \ if \ bool = true \\ 0, \ otherwise \end{cases}$

Finally, the mean log-likelihood function is obtained by taking the expectation of the log-likelihood (5), where

$$\begin{aligned} &\mathbb{E}_{\mathbf{X}, \mathbf{\Sigma}_{\mathbf{x}} | \mathbf{Y}, s_{\mathbf{X}}, \theta_X^l} \left[\delta_{\sigma_n=i, x_n=k} \right] \\ &= \mathbb{P}_{\theta_X^l}(\Sigma_n = i, X_n = k | \mathbf{y}, \mathbf{s}_{\mathbf{x}}) \\ &= \mathbb{P}_{\theta_X^l}(\Sigma_n = i | \mathbf{y}, \mathbf{s}_{\mathbf{x}}) \mathbb{P}_{\theta_X^l}(X_n = k | \Sigma_n = i, \mathbf{y}, \mathbf{s}_{\mathbf{x}}) \\ &\mathbb{E}_{\mathbf{X}, \mathbf{\Sigma}_{\mathbf{x}} | \mathbf{Y}, s_{\mathbf{X}}, \theta_X^l} \left[\delta_{\sigma_{n-1}=i, \sigma_n=j} \right] \\ &= \mathbb{P}_{\theta_X^l}(\Sigma_{n-1} = i, \Sigma_n = j | \mathbf{y}, \mathbf{s}_{\mathbf{x}}) \end{aligned} \tag{6}$$

See Equation 26 in Appendix A for the expanded expression of the mean log-likelihood function.

C. Maximization step

Here, the mean log-likelihood function is maximized versus θ_X, given the estimate θ_X^l [19], and under the constraints, $\forall i, j \in \{s, d\}$:

$$p_i \in [0,1], \ and \ t_{ij} \in \ ]\,0,1\,[\ , \ and \sum_{j \in \{s,d\}} t_{ij} = 1 \tag{7}$$

The new parameters, solutions to the maximization problem (3), are:

$$\begin{aligned} p_i^{(l+1)} &= \frac{\sum_{n=1}^{N} \mathbb{P}_{\theta_X^l}(\Sigma_n = i | \mathbf{y}, \mathbf{s}_{\mathbf{x}}) \mathbb{P}_{\theta_X^l}(X_n = 1 | \Sigma_n = i, \mathbf{y}, \mathbf{s}_{\mathbf{x}})}{\sum_{n=1}^{N} \mathbb{P}_{\theta_X^l}(\Sigma_n = i | \mathbf{y}, \mathbf{s}_{\mathbf{x}})} \\ t_{ij}^{(l+1)} &= \frac{\sum_{n=2}^{N} \mathbb{P}_{\theta_X^l}(\Sigma_{n-1} = i, \Sigma_n = j | \mathbf{y}, \mathbf{s}_{\mathbf{x}})}{\sum_{n=1}^{N-1} \mathbb{P}_{\theta_X^l}(\Sigma_n = i | \mathbf{y}, \mathbf{s}_{\mathbf{x}})} \end{aligned} \tag{8}$$

See Appendix A for a detailed demonstration of the results in Equation (8).

Therefore, the maximization step needs the current *a posteriori probabilities* (APP) of the states $\mathbf{\Sigma}$ and the source $\mathbf{X}$. Due to the cycles in the graph, these quantities are too complex to compute, but they can efficiently be approximated by a *belief propagation* algorithm run on the graph shown in Fig. 3. The algorithm proceeds in two steps explained below: the soft estimates of the states result from a *forward-backward-like algorithm* (see Appendix B for detailed presentation of the algorithm), and the soft estimates of the source symbols result from an *LDPC-decoding-like* algorithm.

D. LDPC belief propagation for the soft estimate of $\mathbf{X}$

Given the current estimate of the parameters θ_X^l and the soft estimate of the states $\sigma_{\mathbf{x}}^l$, we find the best approximation of the *a posteriori* probabilities (APP) $\mathbb{P}_{\theta_X^l}(X_n = k | \sigma_n, \mathbf{y}, \mathbf{s}_{\mathbf{x}}), n \in [1, N], k \in \{0, 1\}$ needed for the parameters updates in Equation (8) of the maximization step. As side products, we also obtain the estimate $\mathbf{x}^l$. Here, we describe the update rules for the belief-propagation run on the graph in Fig. 3.

1) Messages from the state nodes to the variable nodes:

$$S_n = \log\left(\frac{\mathbb{P}_{\theta_X^l}(X_n = 0)}{\mathbb{P}_{\theta_X^l}(X_n = 1)}\right) = \log\left(\frac{1 - p_{Xn}^l}{p_{Xn}^l}\right) \tag{9}$$

where p_{Xn}^l is obtained from the states probabilities $\mathbb{P}_{\theta_X^l}(\sigma_n | \mathbf{y}, \mathbf{s}_{\mathbf{x}})$ (from the forward-backward algorithm), and the current estimate θ_X^l. More precisely, $\forall n \in [1, N]$:

$$p_{Xn}^l = \sum_{i \in [s,d]} p_i^l \cdot \mathbb{P}_{\theta_X^l}(\Sigma_n = i | \mathbf{y}, \mathbf{s}_{\mathbf{x}}) \tag{10}$$

2) Intrinsic information of the BSC nodes:

$$I_n = \log\left(\frac{\mathbb{P}(Z_n = 0)}{\mathbb{P}(Z_n = 1)}\right) = \log\left(\frac{1-p}{p}\right) \tag{11}$$

Here, the value of p is assumed to be known prior to decoding.

3) Messages from the BSC nodes to the variable nodes:

$$B_n = (1 - 2y_n) I_n \tag{12}$$

4) Messages from the variable nodes to the check nodes:

$$E_{n,e}^{(out)} = B_n + \sum_{k=1, k \neq e}^{d_{xn}} E_{n,k}^{(in)} + S_n, \tag{13}$$

Each $E_{n,e}^{(out)}$ is mapped to the corresponding $Q_{m,e}^{(in)}$ according to the connections in the factor graph.

5) Messages from the check nodes to the variable nodes:

$$Q_{m,e}^{(out)} = 2\tanh^{-1}\left[(1-2s_n)\prod_{k=1,k\neq e}^{d_{sm}}\tanh\frac{Q_{m,e}^{(in)}}{2}\right] \quad (14)$$

Each $Q_{m,e}^{(out)}$ is mapped to the corresponding $E_{n,k}^{(in)}$.

6) Messages from the variable nodes to the state nodes: We compute the *extrinsic* LLR E_n for each x_n, as:

$$E_n = B_n + \sum_{k=1}^{d_{xn}} E_{n,k}^{(in)} \quad (15)$$

Then, the variable-to-state messages are given by:

$$\begin{cases} V_n(0) = \dfrac{e^{E_n}}{1+e^{E_n}} \\ V_n(1) = 1 - V_n(0) \end{cases} \quad (16)$$

For this LDPC decoding, we have decided to propagate LLR, which implies their conversion to probabilities, in (16), for use in the maximization step. So far, the values of $V_n(0)$ and $V_n(1)$ are the best guess on the *a posteriori* probabilities $\mathbb{P}_{\theta_X^l}(X_n = 0|\sigma_n, \mathbf{y}, \mathbf{s_x})$ and $\mathbb{P}_{\theta_X^l}(X_n = 1|\sigma_n, \mathbf{y}, \mathbf{s_x})$.

7) Decision:
After each iteration of the EM algorithm, a hard decision is made on $V_n(1)$ to get the estimated symbols of $\mathbf{x}^{(l+1)}$.

$$\forall n \in [1, N], x_n^{(l+1)} = \begin{cases} 1, \ if \ V_n(1) \geq 0.5 \\ 0, \ otherwise \end{cases} \quad (17)$$

E. Stopping criteria: syndrome check, convergence test, and maximum number of iterations

The estimation-decoding EM algorithm stops either if the estimated $\mathbf{x}^{(l+1)}$ satisfies the parity check equations (defined by $\mathbf{H}\mathbf{x}^{(l+1)} = \mathbf{s_x}$), or if the syndrome test has failed while no symbol of $\hat{\mathbf{x}}$ has been updated during the current iteration (we decide that the decoder has converged to a wrong word), or if the maximum number of iterations has been reached (100 iterations is a good compromise between performance and complexity).

F. Initialization of the EM algorithm

As exhibited in Lemma 1, the side-information $\{Y_n\}$ is a GE source that has the same states as the source $\{X_n\}$. Therefore, to initialize the EM algorithm, we use the estimated GE parameters associated to $\mathbf{y}$, $\{\hat{\theta}_Y, \hat{\sigma}_\mathbf{y}\}$; that is the best guess on $\{\theta_X^0, \sigma_\mathbf{x}^0\}$ so far. To estimate these parameters, we use a simplified version of the EM algorithm: the Baum-Welch algorithm [18]. In the Baum-Welch algorithm, the hard values of the bits are directly used, instead of their probabilities, and no LDPC decoding is performed. To that end, we take $\mathbb{P}(Y_n = 1) = y_n$, $\mathbb{P}(Y_n = 0) = (1 - y_n)$ in Equation (8) of Section III. This Baum-Welch algorithm is initialized with the arbitrary values:

$$\theta_X^0 = (p_s^0 = 0.49, p_d^0 = 0.51, t_{ds}^0 = 0.1, t_{sd}^0 = 0.1) \quad (18)$$

Since there is no *a priori* on the Bernoulli parameters p_s and p_d, they have to be initialized by a value close to 0.5. However they cannot be initialized with the same value $p^0 = 0.5$, since the parameters would not be updated from an iteration to the next (this is a consequence of Equation (8)). Moreover, since by definition of the GE process $p_s \leq p_d$, this order must be kept for the initialization. As for the initialization of the transition probabilities, we observed that the value 0.1 allows to speed the convergence rate of the EM algorithm.

G. Simulation results: Distributed coding of GE sources

We consider a GE source $\{X_n\}$ with parameters $\theta_X = (p_s = 0.07, p_d = 0.7, t_{ds} = 0.03, t_{sd} = 0.01)$, having realization $\mathbf{x}$ of length $N = 1584$ (same length as the video bit-planes, in the DVC experiments with QCIF sequences reported in Section IV). We also consider a memoryless binary noise $Z \sim \mathcal{B}(p)$ with realization $\mathbf{z}$, and a second source $\{Y_n\}$, with realization $\mathbf{y}$, correlated to $\{X_n\}$ s.t. $\forall n, Y_n = X_n \oplus Z_n$ (additive BSC). Therefore, the entropy of the noise is $H(Z) = H(p) = -p\log(p) - (1-p)\log(1-p)$. The entropy-rates $H(\mathcal{X})$ and $H(\mathcal{Y})$ are calculated using the statistical approach in [26]. The Slepian-Wolf bound for the coding of $\{X_n\}$, i.e. its conditional entropy-rate, is thus obtained by $H(\mathcal{X}|\mathcal{Y}) = H(Z) - [H(\mathcal{Y}) - H(\mathcal{X})]$.

To prove the enhanced performance of the proposed DSC decoder, the syndrome of $\mathbf{x}$, as well as the side-information $\mathbf{y}$, are transmitted to *three* different decoders:

- ⓐ the *standard* decoder, which views $\{X_n\}$ as a uniform source;
- ⓑ the *proposed* decoder, which assumes that $\{X_n\}$ is a GE source and uses the EM algorithm to iteratively estimate its parameter θ_X;
- ⓒ a *genie-aided* decoder, which assumes that $\{X_n\}$ is a GE source and knows the parameter θ_X.

All *three* decoders have the same variable degree distribution $\Lambda(x) = 0.457827x + 0.323775x^2 + 0.0214226x^3 + 0.0592851x^5 + 0.0389015x^6 + 0.0248109x^7 + 0.00884569x^8 + 0.0176697x^{18} + 0.04746251x^{19}$, and check degree distribution $\Phi(x) = x^8$. The BER of $\{X_n\}$ corresponding to the *three* decoders are plotted in Fig 4, the estimated parameters from the decoder ⓑ are plotted in Fig. 5.

First, we see in Fig. 4 that the proposed decoder ⓑ performs considerably better than the original decoder ⓐ. In particular, the source is retrieved error-free for $H(p) = 0.5$, which corresponds to the SW bound for uniform sources. Moreover, the plots ⓑ and ⓒ in Fig. 4 show that knowing the true θ_X is not essential to the decoder, since it does not improve the performance of the decoder in a significant amount: the rate improvement from ⓑ to ⓒ is less than 0.02 bit for any value taken by p.

It is shown in Fig. 5 that the EM algorithm manages to closely retrieve the parameters of $\{X_n\}$ in the range where the BER is small (i.e. $(H(p) < 0.65)$, see Fig. 4).

IV. Distributed video coding exploiting the memory of the WZ bit-planes

We consider the transform-domain DVC codec described in [27], developed by the European IST-DISCOVER project [28],

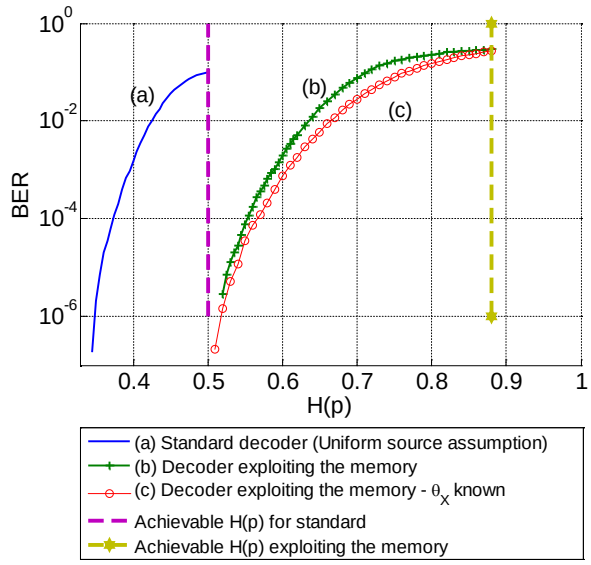

Fig. 4. Performances of the *three* decoders, for a GE source $\{X_n\}$ of parameter $\theta_X = (p_s = 0.07, p_d = 0.7, t_{ds} = 0.03, t_{sd} = 0.01)$. When exploiting the memory, $H(\mathcal{X}|\mathcal{Y}) = 0.5$ occurs for $H(p) = 0.88$ ($p = 0.299$), instead of $H(p) = 0.5$ ($p = 0.11$) when considering the source as uniform.

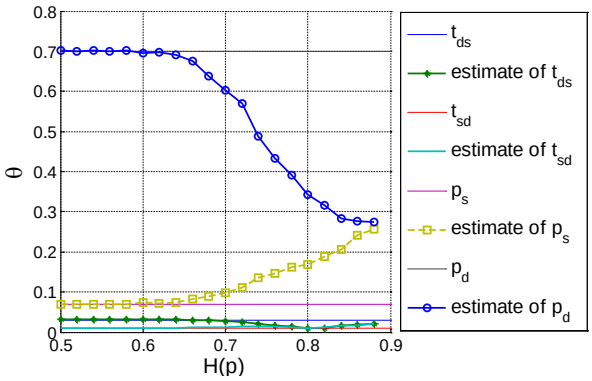

Fig. 5. Performance of the parameter estimation, for a GE source X of parameter $\theta_X = (p_s = 0.07, p_d = 0.7, t_{ds} = 0.03, t_{sd} = 0.01)$.

and which will be referred to as the *DISCOVER codec* in the sequel.

A. Review of the DISCOVER codec

Fig. 6 shows the DISCOVER codec block diagram.

The encoder first splits the video frames into key frames and WZ frames. The key frames are conventionally encoded using an H264/AVC encoder and transmitted to the decoder. The WZ frames are first transformed with a Discrete Cosine Transform (DCT), and the obtained transform coefficients are quantized. The quantized coefficients are organized into bands, where every band contains the coefficients associated to the same frequency in different blocks. Then, the quantized coefficients bands are fed bit-plane by bit-plane to a SW encoder, which computes their syndromes.

At the decoder, the key frames are first decoded using a conventional video decoder. Then a motion compensated interpolation between every two closest key frames is performed, in order to produce the SI for a given WZ frame. The WZ frame is also split into blocks which are then DCT transformed. The correlation channel between the WZ and SI DCT coefficients is approximated by a Laplacian model [29]. Knowing the side-information and the syndrome bits, the decoder estimates the transmitted bit-planes. If the number of syndrome bits is not sufficient to have a BER lower than 10^{-4} at the output of the decoder, more syndrome bits are requested from the encoder.

B. Accuracy of the GE model

The question we deal with here is whether the bit-planes are better approximated by the proposed GE source model (Section II-B) or by a non-uniform source model. To that end, we investigate the distribution of the bursts of 1's, i.e. the number of consecutive 1's, in the bit-planes.

First, consider a binary source X without memory. It can be modeled as a Bernoulli process with parameter $p_X = \mathbb{P}(X = 1)$. In this case, the burst length distribution, $(\mathcal{P}_k)_{k\geq 1}$, is defined as the probability of having a sequence with k consecutive 1's, given that the sequence starts with a 0 (i.e. $0\underbrace{1\ldots1}_{k}0$). For the Bernoulli process,

$$\mathcal{P}_k = (1 - p_X)p_X^{k-1} \tag{19}$$

Therefore, for Bernoulli sequences, the log-scale burst distribution $\log(\mathcal{P}_k)$ is linear in the burst length.

Now, consider a GE source. Its steady-state distribution is $\Pi = \frac{1}{t_{ds}+t_{sd}}[p_s t_{ds}\ p_d t_{sd}]$. Let $B = \left(\begin{smallmatrix} p_s & 0 \\ 0 & p_d \end{smallmatrix}\right)$ be the matrix which diagonal values are the Bernoulli parameters. Let $P = \left(\begin{smallmatrix} (1-t_{sd}) & t_{sd} \\ t_{ds} & (1-t_{ds}) \end{smallmatrix}\right)$ be the state transition probability matrix. Let $I = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ be the identity matrix. We also introduce the following notation $C = \left(\begin{smallmatrix} 1-p_s \\ 1-p_d \end{smallmatrix}\right)$ and $\mathbb{I} = \left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right)$. Then, the burst length distribution is:

$$\mathcal{P}_k = \frac{\Pi\,(I - B)\,(PB)^k\,C}{\Pi\,(I - B)\,(PB)^2\mathbb{I}} \tag{20}$$

Let us consider the bit-plane sequences obtained within the DISCOVER codec. For those sequences, we plot the *empirical* burst length distributions and the *theoretical* ones. More precisely, the *empirical* distribution is obtained by counting the occurrence of each burst length directly from the bit-plane. Given the GE parameters estimated from a given bit-plane (that are estimated thanks to the Baum-Welch algorithm), the *theoretical* burst length distribution is computed from Equation (20). Fig. 7 shows the comparison of the empirical distribution of a given bit-plane *(bold line)* with two theoretical distributions *(dashed lines)*: one obtained by assuming that the bit-plane has no memory, from Equation (19), and the other obtained by taking into account the memory, from Equation (20). Interestingly, the empirical distribution of the bit-plane matches well with the memory-aware theoretical distribution.

The plots in Fig. 7 only prove that the GE model is accurate for the particular bit-plane that is analyzed. To quantify the memory in a larger number of bit-planes, we look at the behavior of the parameter θ_X in the 100 first bit-planes of some video sequences. The bit-planes are taken one after another, following the decoding order in the DISCOVER decoder. More precisely, for each bit-plane, we observe t_{ds}, t_{sd}, and the ratio $\frac{p_d}{p_s}$. Memory is present in the bit-planes if the *three* following criteria are satisfied:

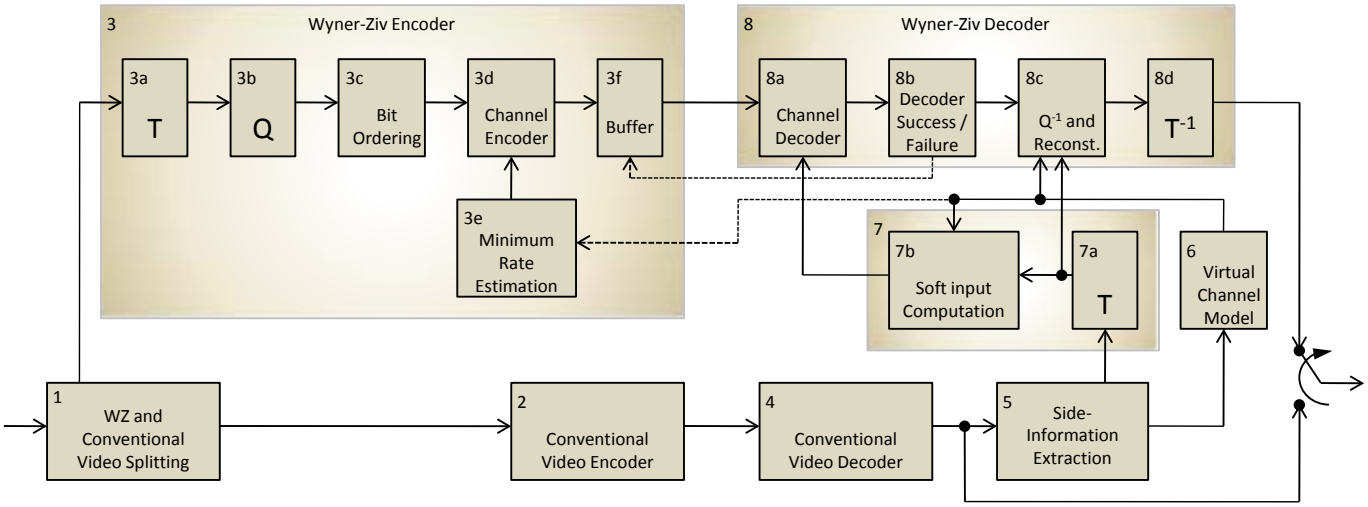

Fig. 6. Block diagram of the IST-DISCOVER DVC codec. The approach is *rate-adaptive*, meaning that the syndrome is transmitted incrementally to avoid re-encoding the data.

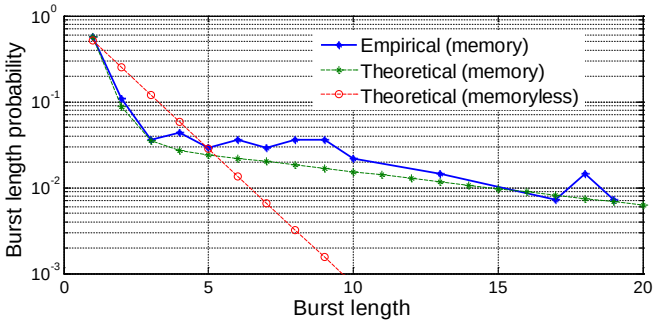

Fig. 7. Distribution of the bursts of 1's for a selected bit-plane of the video sequence *Soccer*.

$$(a)\ t_{ds} \ll 0.5, \qquad (b)\ t_{sd} \ll 0.5, \qquad (c)\ \frac{p_d}{p_s} \gg 1 \quad (21)$$

The criteria (a) and (b) imposing low transition probabilities mean that the memory is *persistent*; the criterion (c) imposing a high ratio of the two Bernoulli parameters means that the two states are clearly distinguishable. Otherwise, the two states are similar and the EM algorithm is not able to differentiate them.

The estimated transition parameters are shown in Fig. 8 for the three video sequences *Hall Monitor*, *Foreman* and *Soccer*. Note that the transition probabilities can be very low, mainly for the sequence *Soccer*; this does account for a huge persistence of the states, hence of the memory in the bit-planes.

The ratio of the estimated Bernoulli parameters are shown in Fig. 9 for the same video sequences, and the same 100 first bit-planes. Here, we note that the ratio can be greater than 5 in some of the bit-planes.

The combination of the low transition probabilities (Fig. 8) and the huge Bernoulli parameters ratio (Fig. 9) accounts for the presence and for the persistence of memory in the WZ bit-planes. The amount of WZ bit-planes that fulfill the criteria stated in (21) is large enough to justify the modeling.

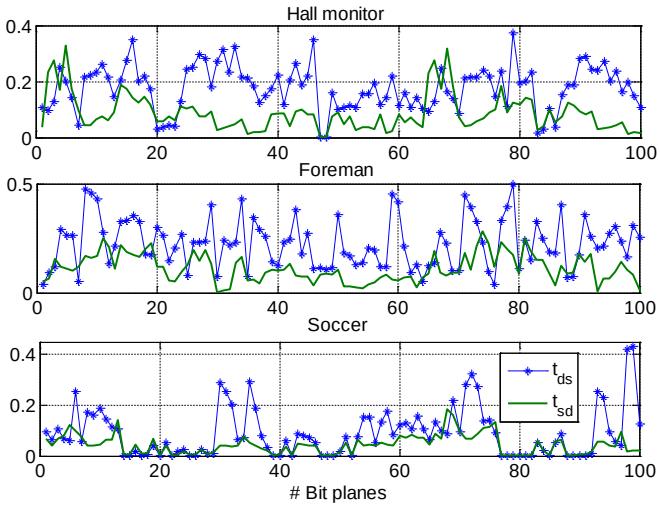

Fig. 8. The estimated transition parameters (t_{sd} and t_{ds}) from the 100 first bit-planes from the video sequences *Hall Monitor*, *Foreman* and *Soccer*.

C. Exploiting the bit-planes memory together with the Laplacian correlation channel

In the DISCOVER codec, the WZ bit-planes, modeled as the realizations of GE sources, are obtained from the binarization of the quantized coefficients resulting from the DCT transform applied on the WZ frames. First, we obtain the SI bit-planes from the binarization of the quantized SI DCT coefficients. Then, the WZ syndromes and the SI bit-planes are input to the EM algorithm (see Section III), which estimates the WZ bit-planes of the current frame. The EM algorithm estimates the statistics of the source while the statistics of the (BSC) correlation noise is estimated with [30].

However, the binarization of the SI is suboptimal since only partial information of the SI is used. Therefore, we now consider the Laplacian model between the obtained WZ DCT coefficients $\mathbf{X}_k$ and the SI DCT coefficients $\mathbf{Y}_k$ introduced in [29], i.e. $\mathbf{Y}_k = \mathbf{X}_k + \mathbf{Z}_k$, where $\mathbf{Z}_k$ stands for the Laplacian

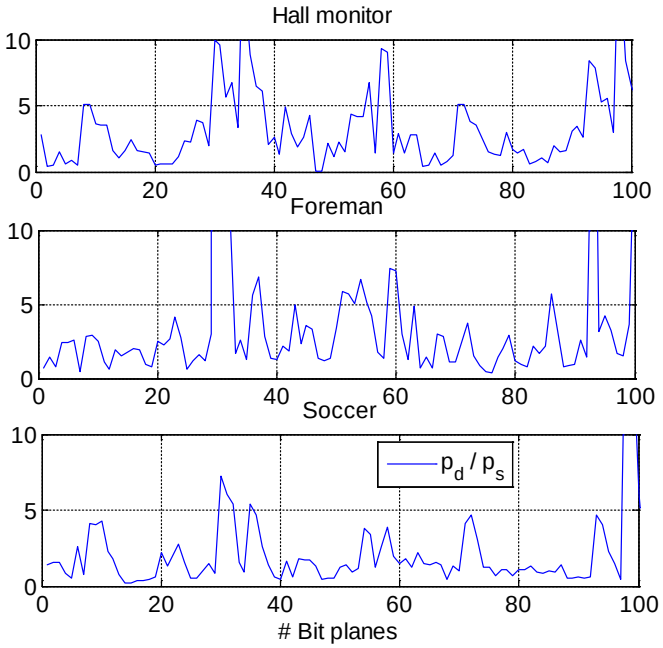

Fig. 9. The ratio of the estimated Bernoulli parameters ($\frac{p_d}{p_s}$) from the video sequences *Hall Monitor*, *Foreman* and *Soccer*.

noise, and k for the frequency band. The density function of the noise is given by:

$$p_{Z_k}(z) = \frac{\hat{\alpha}_k}{2} e^{-\hat{\alpha}_k |z|} \tag{22}$$

where $\hat{\alpha}_k$ is the Laplacian parameter estimated from each frequency band k of the SI. We further assume that the noise $\mathbf{Z}_k$ is independent of the WZ coefficients $\mathbf{X}_k$, as in the definition of the additive channel. Note that this definition differs from the assumption in [31, Proposition 1], where the noise is assumed to be independent of the SI. It follows that the *a priori* probabilities have to be recomputed with respect to previous DVC decoders taking into account the GE model and the independence between the WZ coefficients and the noise. More precisely, $\forall n \in [1, N]$ and for each DCT frequency band k, the WZ coefficient $x_{k,n}$ is quantized into B bits. The b-th bit denoted $x^b_{k,n}$ has an associated *a priori* probability that takes into account the previously decoded bit-planes. The corresponding LLR is given by:

$$\log\left(\frac{\mathbb{P}\left(X^b_{k,n}=1|y_{k,n},x^1_{k,n},\ldots,x^{b-1}_{k,n}\right)}{\mathbb{P}\left(X^b_{k,n}=0|y_{k,n},x^1_{k,n},\ldots,x^{b-1}_{k,n}\right)}\right) = \log\left(\frac{\int\limits_{x\in Q(1)} p_{Z_k}(y_{k,n}-x)p(x|x^1_{k,n},\ldots,x^{b-1}_{k,n})dx}{\int\limits_{x\in Q(0)} p_{Z_k}(y_{k,n}-x)p(x|x^1_{k,n},\ldots,x^{b-1}_{k,n})dx}\right) \tag{23}$$

where

- $y_{k,n}$ is the n-th DCT coefficient of the frequency band k;
- $Q(m) = Q(m, x^1_{k,n}, \ldots, x^{b-1}_{k,n})$ is the set of all the quantization intervals corresponding to the quantized symbols, which b most significant bits are $(x^1_{k,n}, \ldots, x^{b-1}_{k,n}, m)$.

At iteration l of the EM algorithm, $p(x|x^1_{k,n}, \ldots, x^{b-1}_{k,n})$ can be computed from the current probabilities of the GE source and its parameters. More precisely, the density of the WZ coefficient is assumed to be constant on each quantization interval. It follows that (23) becomes:

$$\log\left(\frac{\mathbb{P}\left(X^b_{k,n}=1|y_{k,n},x^1_{k,n},\ldots,x^{b-1}_{k,n}\right)}{\mathbb{P}\left(X^b_{k,n}=0|y_{k,n},x^1_{k,n},\ldots,x^{b-1}_{k,n}\right)}\right) = \log\left(\frac{\int\limits_{x\in Q(1)} \frac{\hat{\alpha}_k}{2} e^{-\hat{\alpha}_k|y_{k,n}-x|}dx}{\int\limits_{x\in Q(0)} \frac{\hat{\alpha}_k}{2} e^{-\hat{\alpha}_k|y_{k,n}-x|}dx}\right) + \log\left(\frac{1-p^l_{X_n}}{p^l_{X_n}}\right) \tag{24}$$

where p^l_{Xn} is given in (10). This equation (24) replaces "$B_n + S_n$" in Equation (13).

D. First Performance Analysis

In order to assess the performance achieved by the proposed Slepian-Wolf decoder exploiting the memory in the bit-planes, we implement the LDPC-based estimation-decoding EM algorithm for the DISCOVER codec. All the bit-planes do not have memory: some can be i.i.d. binary sources. But this model is a particular case of the GE model, so the decoder adapts by itself to any of the models.

We show in Fig. 10 the performance of the modified decoder if the noise between the WZ coefficients and the SI is assumed to be additive i.e. independent of the WZ coefficients. For the sequence *Soccer*, the rate gain is almost 10% at the highest PSNR, with respect to the standard SW decoder of the DISCOVER codec. However, we see that there is **not always** a rate gain when exploiting the bit-planes memory; E.g., for the sequence *Foreman*, the standard and the proposed decoder have almost the same performance; for the sequence *Hall Monitor*, the performance of the proposed decoder is degraded compared to that of the standard decoder.

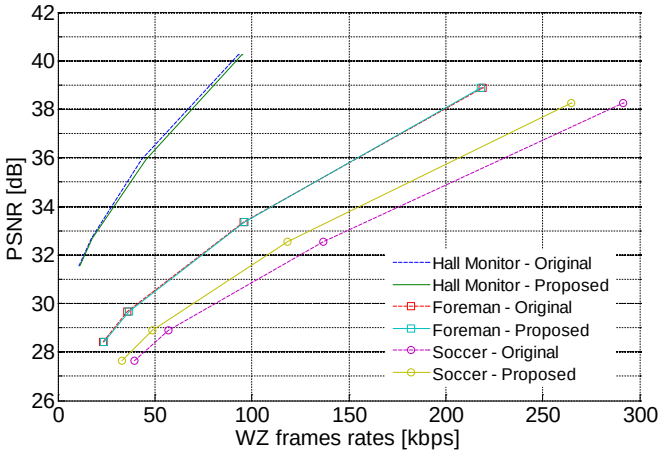

Fig. 10. Behavior of the proposed DVC decoder when the Laplacian channel is assumed to be always additive. Exploiting the memory that is present in the bit-planes does not always improve the performance of the codec.

V. Predictive correlation channel: rate bounds and practical DVC RD performance

We have seen above that, although the GE model fits well with the bit-planes distributions, exploiting the memory in the video bit-planes does not always bring a rate gain, when assuming the correlation model to be additive. We thus consider here a second correlation model, called the *predictive* model.

Definition 2. *An (X, Y, p) predictive BSC is a channel with binary ergodic input $\{X_n\}$, binary ergodic output $\{Y_n\}$. The noise is an i.i.d. binary process $Z \sim \mathcal{B}(p)$. Z is independent of the channel output s.t. $\forall n, X_n = Y_n \oplus Z_n$.*

This model has been introduced in [32], where it is called "*the reverse channel*", and in [31]. This model assumes that the correlation noise Z is independent of the SI, instead of being independent of the source $\{X_n\}$ as in the *additive* model. Note that the *additive* model is the model considered for the correlation noise in most work so far in DSC [4], [5], [6], [11], [12], [13], [14]. Interestingly, most DVC codecs assume that the bitplanes to be compressed are i.i.d. uniform processes. In this case, as shown in the next section (see Lemma 5), the correlation type does not influence neither the decoder nor the achievable compression bound. Therefore, the authors do not explicitly state what the assumed correlation model is. However, [33], [34], [35] use the duality between channel coding and asymmetric DSC, which only holds for additive BSC. So it can be deduced that an additive correlation channel is assumed. However, in [36] a reference to [31, Proposition 1] is made, where the channel is predictive.

A. Rate bounds for the predictive correlation channel

In this section, we study the case when the correlation between the binary sources $\{X_n\}$ and $\{Y_n\}$ is a *predictive* channel.

Lemma 3. *Let $\{X_n\}$ and $\{Y_n\}$ be two correlated binary ergodic sources, where the correlation is modeled as a virtual (X, Y, p) predictive BSC i.e. there exists an i.i.d. binary process (noise) $Z \sim \mathcal{B}(p)$ s.t. $\forall n, X_n = Y_n \oplus Z_n$. We consider the asymmetric DSC problem, where $\{Y_n\}$ is available at the decoder and $\{X_n\}$ is transmitted at a rate greater than its conditional entropy-rate $H(\mathcal{X}|\mathcal{Y})$.*

The minimum transmission rate for $\{X_n\}$ only depends on the noise statistics. More precisely $H(\mathcal{X}|\mathcal{Y}) = H(Z)$. Therefore the minimum transmission rate for $\{X_n\}$ is not reduced, with respect to that of a uniform source.

Proof: Here Z is independent of $\{Y_n\}$. Therefore, $H(\mathcal{X}|\mathcal{Y}) = H(Z)$ and the minimum coding rate for $\{X_n\}$ only depends on the noise statistics. So, the minimum transmission rate for $\{X_n\}$ is not reduced, with respect to that of a uniform source. ■

Lemma 4. *Let $\{X_n\}$ and $\{Y_n\}$ be two correlated binary ergodic sources, where the correlation is modeled as a virtual predictive channel i.e. there exists an i.i.d. binary process (noise) $Z \sim \mathcal{B}(p)$ s.t. $\forall n, X_n = Y_n \oplus Z_n$.*

The MAP estimate for the source $\{X_n\}$, when $\{Y_n\}$ is available at the decoder, only depends on the noise statistics. Therefore, for a given SI realization $\mathbf{y}$, whatever the binary source model, the performances of the MAP decoders are the same.

Proof: The MAP detection problem

$$\forall n \in [1, N], \hat{x}_n = \arg \max_{x_n \in \{0,1\}} \mathbb{P}(x_n|\mathbf{y}) \tag{25}$$

needs the *a priori* probabilities $\mathbb{P}(x_n|y_n)$, which only depend on $p_Z(y_n - x_n)$ and not on the source statistics. The noise statistics is therefore a sufficient statistic for the MAP detection problem (25). Therefore any MAP decoder with a predictive correlation model, whatever the source statistics, give the same performance. ■

This result is coherent with Lemma 3: since the decoding algorithm is the same whatever the binary source model, the minimum transmission rate is also the same.

Lemma 5. *Let X and Y be two correlated binary, uniform i.i.d. sources. We consider the asymmetric DSC problem, where Y is available at the decoder and X is transmitted at a rate greater than its conditional entropy $H(X|Y)$.*

The compression rate for X and the MAP decoder only depends on the noise statistics and is independent of the correlation channel type (predictive or additive BSC).

Proof: The MAP detection problem (25) needs the *a priori* probabilities $\mathbb{P}(x_n|y_n)$, which is $p_Z(y_n - x_n)$ if the correlation model is predictive and $0.5 * p_Z(y_n - x_n)$ if additive. Thus, the MAP estimate only depends on the noise statistics and is therefore independent of the correlation type.

Moreover, from Lemma 2 (additive BSC) and Lemma 3 (predictive BSC), the compression rate for X is $H(Z)$, whatever the correlation type (additive or predictive BSC). ■

From Lemma 4 and Lemma 5, we conclude that the three decoders with the assumptions detailed below give the same performances:
- either the sources are memoryless with an additive correlation model,
- or the sources are memoryless with a predictive correlation model,
- or the sources have memory with a predictive correlation.

B. Rate loss in case of mismatch between the additive and the predictive correlation models

In the previous Section, we have shown the minimum transmission rate when the correlation channel is predictive (Lemma 3). These bounds are derived under the hypothesis that the decoder is matched to the source distribution and to the correlation type. When the decoder is mismatched, there is a rate loss, which is evaluated below.

Let us first assume that $\{X_n\}$ and $\{Y_n\}$ are two GE sources and that the correlation model is additive. We consider the same parameter setup as in Section III-G and would like to estimate the rate loss when the decoder assumes that the correlation type is predictive, whereas the true one is additive. If the decoder assumes that the correlation type is additive,

the performance are given by the curve ⓑ in Fig. 4. From Lemma 4, the decoder that assumes that the correlation type is predictive is the standard decoder ⓐ (in Fig. 4), that does not take into account the source statistics. Therefore, the rate loss due to a mismatch between a true additive and assumed predictive correlation channel is the distance between the curves ⓐ and ⓑ in Fig. 4.

Now we consider the case where the true correlation channel is *predictive* between the two GE sources $\{X_n\}$ and $\{Y_n\}$, i.e. $\forall n, X_n = Y_n \oplus Z_n$. If the decoder is matched to the correlation type, the BER of X is given by the curve ⓐ in Fig. 11. If the decoder is mismatched and assumes that the correlation type is additive, the BER is greater (see curve ⓑ in Fig. 11). Therefore, we conclude that a *mismatch* between the true predictive and the assumed additive correlation channel degrades the performance of the decoder. It is therefore important to estimate the correlation type. In the following, we propose an estimator for the correlation type to be used in a DVC codec.

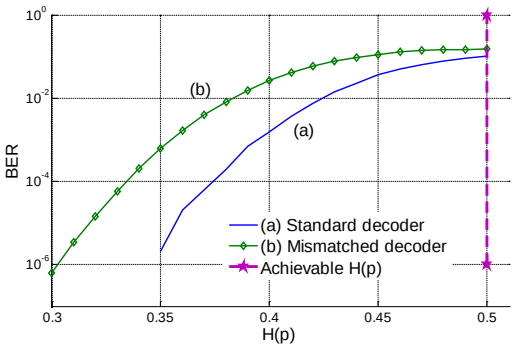

Fig. 11. Performances of the standard decoder ⓐ and the proposed decoder ⓑ exploiting the source memory. The mismatch degrades the performance when the memory is exploited.

C. DVC RD performance when estimating the correlation model

From the results above, it is important to use in the LDPC decoding algorithm the most appropriate correlation model. Therefore, we add an extra step to the DVC decoder to estimate on-the-fly the correlation model. To that end, for each syndrome bits request made by the decoder, the decoding is first performed with the *additive* channel assumption; if this decoding fails, (see the condition in Section III-E), another decoding is carried out, assuming that the channel is *predictive*.

Fig. 12 shows the results obtained for all the WZ frames of the video sequences *Hall Monitor*, *Foreman* and *Soccer*. Interestingly, we now observe that for all video sequences, the proposed algorithm decreases the rates, which was not the case when the correlation model was always assumed to be additive (see Fig. 10). The decrease of rate for the sequence *Hall Monitor* is $2.54kbps$ (-2.72%) at the highest PSNR, while it is $8.76kbps$ (-4%) for *Foreman*, and $29.55kbps$ (-10.14%) for the sequence *Soccer*. This proves that it is worth taking into account the memory estimated from the bit-planes, and that the additive channel is not sufficient to model the correlation in the bit-planes generated by the DVC codec.

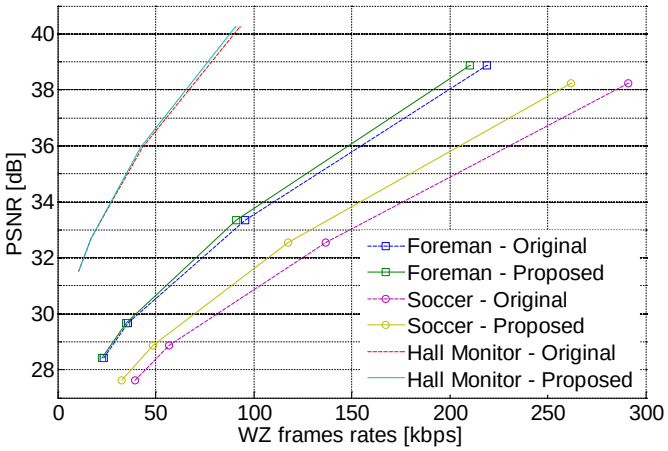

Fig. 12. The decoder that exploits the bit-planes memory needs less rate to render the videos at the same PSNR.

In Table I we show the percentage of use of the additive channel model during the decoding of the three sequences. These figures have been obtained *a posteriori*, after the decoding has ended.

Sequence	% use of additive channel model
Hall Monitor	7.6
Foreman	37.8
Soccer	58.1

TABLE I
PERCENTAGE OF USE OF THE ADDITIVE CHANNEL MODEL.

The combination of the results presented in Fig. 12 and in Table I shows that the more the additive channel model is used, the more gain is obtained for the decoding of the corresponding test sequence.

D. Non-uniform source modeling of the bit-planes

The GE process is a simple yet accurate model for the video bit-planes. However the estimation of the states of the HMM increases significantly the decoder complexity. Therefore, we propose here a simplified model, where only the non-uniformity of the source is exploited. Note that the *non-uniform source model* is a particular case of the hidden Markov model, with the particular values $p_s = p_d = p_{WZ}$.

We first investigate the statistics of the WZ bit-planes by assessing *off line* their *Bernoulli distributions*. To that end, we measure the Hamming weight of each WZ bit-plane divided by its length. We show in Fig. 13 the empirical probabilities of 1's in the 200 first WZ bit-planes from the three video sequences *Hall monitor, Foreman*, and *Soccer* when the PSNR is the highest. These Bernoulli distributions confirm that the bit-planes are non-uniformly drawn. Their distribution varies from bit-plane to bit-plane. Moreover, the parameters have a periodic structure, of period 64 bit-planes; this corresponds to the 64 bit-planes that represent each frame of the video sequence.

The results presented here are obtained for all the WZ frames of the video sequences *Hall monitor, Foreman*, and *Soccer*, with a GOP length of 2. The RD performance of the proposed SW decoder that uses the non-uniformity

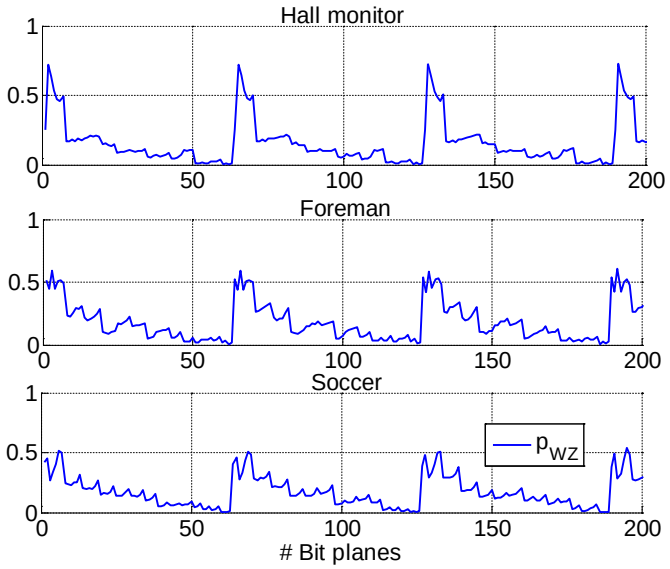

Fig. 13. Probability of 1's in the 200 first WZ bit-planes of each video sequence taken individually. This corresponds to a little more than the 3 first images. The WZ bit-planes of the three video sequences are mostly non-uniformly distributed, which justifies the model adopted here.

is compared to the standard DISCOVER's SW decoder in Fig. 14. The correlation channel model (additive or predictive) is unknown by the decoder, and has to be assessed as explained above for infinite memory sources (Fig. 12 in section V-C).

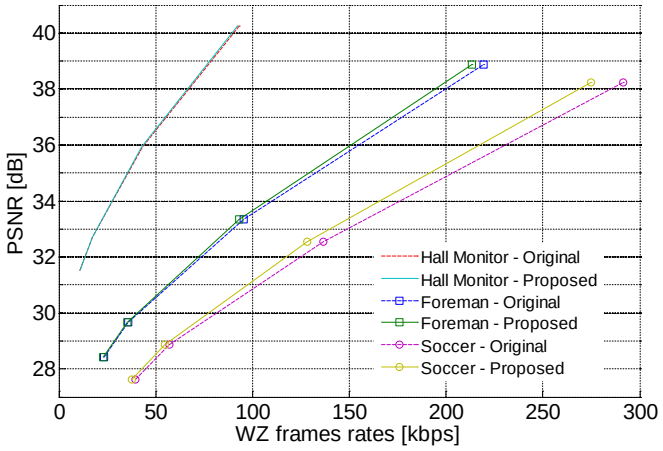

Fig. 14. The proposed decoder that exploits the non-uniformity, while assessing the type of the channel, needs less rate than the standard one to render the videos at the same PSNR.

The rate decrease is $0.94kbps$ (-1%) for the sequence *Hall monitor* at the highest PSNR; it is $5.88kbps$ (-2.68%) for *Foreman*; and $16.57kbps$ (-5.7%) for *Soccer*. That decrease proves that it is worth taking into account the non-uniformity of the bit-planes although the gain is less than by exploiting their memory. The advantage is that only *one* parameter has to be estimated by the decoder (instead of *four* parameters for the GE modeling).

E. Markov source modeling of the bit-planes

The *Markov source model* is the simplest representation of the memory that lies in the binary bit-planes. The Markov source is a particular instance of the two-state hidden Markov source, since it corresponds to the case where $p_s = 0$ and $p_d = 1$. This implies that the states are directly observed from the source realization itself, and the transition parameters can also be directly computed from the observed source realization.

We modify the estimation-decoding EM algorithm to estimate the state parameters and the transition probabilities of this Markov model. Then, we place the modified SW decoder in the DISCOVER codec. The results that we obtained show that there is practically no rate gain for the coding of the WZ bit-planes, with respect to the standard DISCOVER codec. More precisely the rate improvement is only $0.27kbps$ for the sequence *Hall Monitor* at the highest PSNR; it is $0.18kbps$ for *Foreman* and $1.15kbps$ for *Soccer*. It is worth noting that the non-uniform source model (Section V-D) brings more improvement than the Markov one, in terms of the PSNR versus rate performance of the DISCOVER codec. This might be counter-intuitive, but for two correlated Markov sources X and Y, the states of X and Y do not correspond (Lemma 1 do not stand for Markov sources); we can say that the knowledge of the states of Y is misleading for the decoding of X.

Therefore, the Markov source model is not suited to the memory that lies in the WZ bit-planes generated by the DVC codec.

VI. Conclusion

We have proposed a new source model for DVC bit-planes: a Gilbert-Elliott model that takes into account the memory in the original video bit-planes, complemented with a correlation model that takes into account the nature of the noise. More precisely, the correlation channel is modeled as a *predictive* or an *additive* BSC (or Laplacian channel). We have also shown that a mismatch between the true and the assumed correlation model degrades the compression rate. Therefore, we have proposed a test to make a decision between the two models. A joint estimation-decoding EM algorithm is performed for the parameter estimation and the decoding. For the DVC setup, we demonstrated the accuracy of the source model for the WZ bit-planes; the rate gain is up to 10.14% when the bit-planes are considered as GE sources, with respect to the case where they are considered as uniform sources. Finally, a simplified model with non-uniform sources has been proposed that achieves an improvement by up to 5.7%.

Appendix

A. Details for the maximization step of the EM algorithm

Here, we derive the parameters update rules in Equation (8) for the maximization problem (3). From Equations (5) and (6), the mean log-likelihood function is expressed as:

$$\begin{aligned}
&\mathbb{E}_{\mathbf{X},\mathbf{\Sigma_x}|\mathbf{Y},s_\mathbf{X},\theta_X^l}\left[\log\left(\mathbb{P}_{\theta_X^l}(\mathbf{x},\sigma_\mathbf{x})\right)\right] = \log\left(\mathbb{P}_{\theta_X^l}(\sigma_1)\right)\\
&+\sum_{n=2}^{N}\sum_{i=s}^{d}\sum_{j=s}^{d}\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=j|\mathbf{y},\mathbf{s_x})\log(t_{ij}^l)\\
&+\sum_{n=1}^{N}\sum_{i=s}^{d}\mathbb{P}_{\theta_X^l}(\Sigma_n=i|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=1|\mathbf{y},\mathbf{s_x})\log(p_i^l)\\
&+\mathbb{P}_{\theta_X^l}(\Sigma_n=i|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=0|\mathbf{y},\mathbf{s_x})\log(1-p_i^l)
\end{aligned} \tag{26}$$

Then the partial derivatives with respect to the Bernoulli parameters are, $\forall i \in \{s,d\}$:

$$\frac{\partial}{\partial p_i^l}\mathbb{E}_{\mathbf{X},\mathbf{\Sigma_x}|\mathbf{Y},s_\mathbf{X},\theta_X^l}\left[\log\left(\mathbb{P}_{\theta_X^l}(\mathbf{x},\sigma_\mathbf{x})\right)\right] = \left(\frac{1}{p_i^l}\right)\sum_{n=1}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_n = i|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=1|\mathbf{y},\mathbf{s_x}) + \left(\frac{1}{1-p_i^l}\right)\sum_{n=1}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_n = i|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=0|\mathbf{y},\mathbf{s_x}) \quad (27)$$

Then, the derivative (27) is zero when, $\forall i \in \{s,d\}$:

$$p_i^{(l+1)} = \frac{\sum_{n=1}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_n = i|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=1|\Sigma_n=i,\mathbf{y},\mathbf{s_x})}{\sum_{n=1}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_n=i|\mathbf{y},\mathbf{s_x})} \quad (28)$$

For the transition parameters, given that $t_{ss} = (1-t_{sd})$ and $t_{dd} = (1-t_{ds})$ (1), the partial derivatives are given by, $\forall i,j \in \{s,d\}, i \neq j$:

$$\frac{\partial}{\partial t_{ij}^l}\mathbb{E}_{\mathbf{X},\mathbf{\Sigma_x}|\mathbf{Y},s_\mathbf{X},\theta_X^l}\left[\log\left(\mathbb{P}_{\theta_X^l}(\mathbf{x},\sigma_\mathbf{x})\right)\right] = \left(\frac{1}{t_{ij}^l}\right)\sum_{n=2}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=j|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=1|\mathbf{y},\mathbf{s_x}) + \left(\frac{1}{1-t_{ij}^l}\right)\sum_{n=2}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=i|\mathbf{y},\mathbf{s_x})\mathbb{P}_{\theta_X^l}(X_n=0|\mathbf{y},\mathbf{s_x}) \quad (29)$$

Then, the derivative (29) is zero when, $\forall i,j \in \{s,d\}, i \neq j$:

$$t_{ij}^{(l+1)} = \frac{\sum_{n=2}^{N}\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=j,\mathbf{y},\mathbf{s_x})}{\sum_{n=1}^{N-1}\mathbb{P}_{\theta_X^l}(\Sigma_n=i|\mathbf{y},\mathbf{s_x})} \quad (30)$$

Note that the solutions (28) and (30) satisfy the constraints (7) of our optimization problem (3).

B. Forward-backward algorithm for the soft estimate of $\mathbf{\Sigma}$

Here, the probabilities of $\mathbf{\Sigma}$ are updated according to the values of the estimate θ_X^l. The aim is to compute, $\forall n \in [1,N]$,

$$\mathbb{P}_{\theta_X^l}(\Sigma_n = i|\mathbf{y},\mathbf{s_x}) = \frac{\mathbb{P}_{\theta_X^l}(\Sigma_n=i,\mathbf{y}|\mathbf{s_x})}{\mathbb{P}_{\theta_X^l}(\mathbf{y}|\mathbf{s_x})}$$
$$\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=j|\mathbf{y},\mathbf{s_x}) = \frac{\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=j,\mathbf{y}|\mathbf{s_x})}{\mathbb{P}_{\theta_X^l}(\mathbf{y}|\mathbf{s_x})} \quad (31)$$

To that end, we decompose the following expression, to retrieve the equations corresponding to the forward-backward recursions:

$$\begin{aligned}&\mathbb{P}_{\theta_X^l}(\Sigma_n=i,\mathbf{y}|\mathbf{s_x})\\ &= \sum_{j\in\{s,d\}}\mathbb{P}_{\theta_X^l}(\Sigma_n=i,\Sigma_{n+1}=j,\mathbf{y}|\mathbf{s_x})\\ &= \sum_{j\in\{s,d\}}\alpha_i^n\cdot\gamma_{i,j}^{n,(n+1)}\cdot\beta_j^{(n+1)}\end{aligned} \quad (32)$$

The forward-backward algorithm is run on the trellis in Fig. 15 defined by the states s and d generating the source symbols, with two branches between the states, labeled by the two possible values $x_n = 0$ and $x_n = 1$.

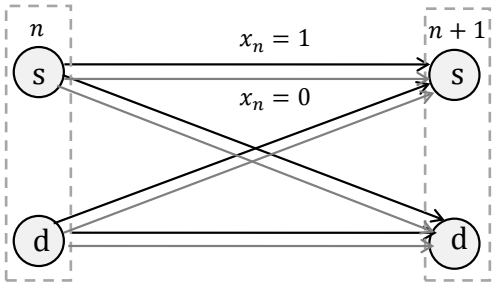

Fig. 15. Trellis on which the forward-backward algorithm is run to estimate the states $\mathbf{\Sigma}$.

We define

$$\begin{aligned}\gamma_{i,j}^{n,(n+1)} &= \mathbb{P}_{\theta_X^l}(y_n|\Sigma_n=i,\mathbf{s_x})\cdot\mathbb{P}_{\theta_X^l}(\Sigma_{n+1}=j|\Sigma_n=i,\mathbf{s_x})\\ \alpha_j^n &= \sum_{i\in\{s,d\}}\alpha_i^{(n-1)}\cdot\gamma_{i,j}^{(n-1),n}\\ \beta_i^n &= \sum_{i\in\{s,d\}}\gamma_{i,j}^{n,(n+1)}\cdot\beta_j^{(n+1)}\end{aligned} \quad (33)$$

where:

- $\gamma_{i,j}^{n,(n+1)}$ is the transition probability between the states i at position n and j at position $(n+1)$.
- α_j^n is the forward probability for the source to be in state j at position n;
- β_i^n is the backward probability for the source to be in state i at position n.

Now we define the states APP:

$$\begin{aligned}&\mathbb{P}_{\theta_X^l}(\Sigma_n=i,\mathbf{y}|\mathbf{s_x}) = \alpha_i^n\cdot\beta_i^n\\ &\mathbb{P}_{\theta_X^l}(\Sigma_{n-1}=i,\Sigma_n=j,\mathbf{y}|\mathbf{s_x}) = \alpha_i^{(n-1)}\cdot\gamma_{i,j}^{(n-1),n}\cdot\beta_j^n\end{aligned} \quad (34)$$

Normalizing $\mathbb{P}_{\theta_X^l}(\sigma_n,\mathbf{y}|\mathbf{s_x})$ and $\mathbb{P}_{\theta_X^l}(\sigma_{n-1},\sigma_n,\mathbf{y}|\mathbf{s_x})$, we get $\mathbb{P}_{\theta_X^l}(\sigma_n|\mathbf{y},\mathbf{s_x})$ and $\mathbb{P}_{\theta_X^l}(\sigma_{n-1},\sigma_n|\mathbf{y},\mathbf{s_x})$.

REFERENCES

[1] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE National Convention Record*, pp. 142–163, 1959.

[2] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions on Information Theory*, vol. 19, no. 4, pp. 471–480, July 1973.

[3] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Transactions on Information Theory*, vol. 21, no. 1, pp. 1–10, January 1976.

[4] J. Garcia-Frias and Y. Zhao, "Compression of correlated binary sources using Turbo codes," *IEEE Communications Letters*, vol. 5, pp. 417–419, October 2001.

[5] A. Aaron and B. Girod, "Compression with side information using Turbo codes," in *Data Compression Conference*, April 2002, pp. 252–261.

[6] A. D. Liveris, Z. Xiong, , and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Communications Letters*, vol. 6, no. 10, pp. 440–442, October 2002.

[7] J. Garcia-Frias, "Decoding of Low-Density Parity-Check codes over finite-state binary Markov channels," *IEEE Transactions on Communications*, vol. 52, no. 11, pp. 1840–1843, November 2004.

[8] A. W. Eckford, F. R. Kschischang, and S. Pasupathy, "Designing good LDPC codes for Markov-modulated channels," *International Symposium on Information Theory (ISIT)*, July 2004.

[9] ——, "Analysis of Low-Density Parity-Check codes for the Gilbert-Elliott channel," *IEEE Transactions on Information Theory*, vol. 51, pp. 3872–3889, November 2005.

[10] J. Garcia-Frias and J. D. Villasenor, "Turbo decoding of Gilbert-Elliot channel," *IEEE Transactions on Communications*, pp. 357–363, March 2002.

[11] J. Garcia-Frias and W. Zhong, "LDPC codes for compression of multi-terminal sources with Hidden Markov correlation," *IEEE Communications Letters*, vol. 7, no. 3, March 2003.

[12] J. D. Ser, P. M. Crespo, and O. Galdos, "Asymmetric joint source-channel coding for correlated sources with blind HMM estimation at the receiver," *EURASIP Journal on Wireless Communications and Networking*, pp. 483–492, 2005.

[13] K. Bhattad and K. R. Narayanan, "A decision feedback based scheme for Slepian-Wolf coding of sources with Hidden Markov correlation," *IEEE Communications Letters*, vol. 10, no. 5, pp. 378–380, May 2006.

[14] V. Aggarval, "Distributed joint source-channel coding for arbitrary memoryless correlated sources and source coding for Markov correlated sources using LDPC codes," *arXiv*, February 2008.

[15] A. Shokrollahi, "Raptor codes," *IEEE/ACM Transactions on Networking*, vol. 14, no. SI, pp. 2551–2567, June 2006.

[16] M. Fresia, L. Vandendorpe, and H. V. Poor, "Distributed source coding using raptor codes for hidden Markov sources," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2868–2875, 2009.

[17] V. Toto-Zarasoa, A. Roumy, and C. Guillemot, "Hidden markov model for distributed video coding," in *IEEE International Conference on Image Processing (ICIP)*, September 2010, pp. 3709–3712.

[18] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, February 1989.

[19] V. Toto-Zarasoa, A. Roumy, and C. Guillemot, "Non-uniform source modeling for distributed video coding," in *European Signal Processing Conference (EUSIPCO)*, August 2010, pp. 1889–1893.

[20] T. M. Cover and J. M. Thomas, *Elements of Information Theory*. New–York: Wiley, 1991.

[21] T. Cover, "A proof of the data compression theorem of Slepian-Wolf for ergodic sources," *IEEE Transactions on Information Theory*, vol. 22, no. 4, pp. 226–268, 1975.

[22] A. Wyner, "Recent results in the Shannon theory," *IEEE Transactions on Information Theory*, vol. 20, pp. 2–10, January 1974.

[23] A. W. Eckford, "The factor graph EM algorithm: applications for LDPC codes," *IEEE 6th Workshop on Signal Processing Advances in Wireless Communications*, 2005.

[24] C. F. J. Wu, "On the convergence properties of the EM algorithm," *Annals of Statistics*, vol. 11, no. 1, p. 95103, 1983.

[25] F. R. Kschischang, B. J. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, February 2001.

[26] M. Rezaeian, "Computation of capacity for Gilbert-Elliott channels, using a statistical method," *6th Australian Communications Theory Workshop*, pp. 56–61, February 2005.

[27] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Proceedings of Picture Coding Symposium*, November 2007.

[28] "DISCOVER project web page," http://www.discoverdvc.org/.

[29] C. Brites and F.Pereira, "Correlation noise modeling for efficient pixel and transform domain WynerZiv video coding," *IEEE Transactions on Circuits and systems for Video Technology*, vol. 18, no. 9, pp. 1177–1190, September 2008.

[30] V. Toto-Zarasoa, A. Roumy, and C. Guillemot, "Maximum likelihood BSC parameter estimation for the Slepian-Wolf problem," *IEEE Communications Letters*, vol. 15, no. 2, pp. 232–234, February 2011.

[31] S. Cheng and Z. Xiong, "Successive refinement for the Wyner-Ziv problem and layered code design," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 3269–3281, August 2005.

[32] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Transactions on Information Theory*, vol. 48, pp. 1250–1276, 2002.

[33] R. Puri and K. Ramchandran, "A new robust video coding architecture based on distributed compression principles," *Proceedings of the 40th Allerton Conference on Communication, Control and Computing*, 2002.

[34] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–73, January 2005.

[35] J. Nayak, G. R. D. Kubasov, and C. Guillemot, "Spatial prediction in distributed video coding using Wyner-Ziv DPCM," *IEEE Workshop on Multimedia Signal Processing*, pp. 311–316, October 2008.

[36] C. Brites and F. Pereira, "Encoder rate control for transform domain Wyner-Ziv video coding," *IEEE International Conference on Image Processing*, pp. 5–8, September 2007.