

# Non-asymmetric Slepian-Wolf coding of non-uniform Bernoulli sources

V. Toto-Zarasoá, A. Roumy and C. Guillemot

INRIA Rennes/University of Rennes, Campus Universitaire de Beaulieu, 35042 Rennes-Cedex, France.

**Abstract**—We address the problem of non-asymmetric Slepian-Wolf (SW) coding of two correlated non-uniform Bernoulli sources. We first show that the problem is not symmetric in the two sources, contrarily to the case of uniform sources, due to the asymmetry induced by two underlying channel models, namely additive and predictive Binary Symmetric Channels (BSC). That asymmetry has to be accounted for during the decoding. In view of that result, we describe the implementation of a joint non-asymmetric decoder of the two sources based on Low-Density Parity-Check (LDPC) codes and Message Passing (MP) decoding. We also give a necessary and sufficient condition for the recovery of the two sources, that imposes a triangular structure of a sub-part of the equivalent matrix representation of the code.

## I. INTRODUCTION

The concept of Distributed Source Coding (DSC) was introduced by Slepian and Wolf (SW) in their seminal paper [1]. They raise the problem of lossless coding of two correlated sources  $X$  and  $Y$ , where the encoding is disjoint and the decoding is joint. It is stated in [1] that the disjoint coding does not incur any rate loss, with respect to the Shannon's bound, as long as the decoding is joint, and the respective transmission rates of  $X$  and  $Y$ ,  $R_X$  and  $R_Y$ , fulfill the three conditions  $R_X + R_Y \geq H(X, Y)$ ,  $R_X \geq H(X|Y)$ , and  $R_Y \geq H(Y|X)$ . Those bounds define the so-called *SW region*.

The usual setup considered in the literature is the *asymmetric* DSC. One source, say  $Y$ , is encoded at its entropy rate  $H(Y)$  and is thus available at the decoder. The other source  $X$  is encoded at its conditional entropy rate  $H(X|Y)$  by sending a compressed version  $S_X$ . The correlation between the two sources is modeled as a virtual *Binary Symmetric Channel* (BSC), represented by the noise  $Z$ . Using that channel interpretation, the first practical DSC codes were based on error-correcting (or *channel*) codes, such as Low-Density Parity-Check (LDPC) codes, and Turbo codes. This setup is shown to be optimal in [2], in the sense that a capacity-achieving channel code can reach the SW bounds as a DSC code.

In this paper, we deal with *non-asymmetric* DSC. This is the general setup where none of the sources is available at the decoder. For example, for joint optimization of the rate and power of transmission in a sensor network application [3], it is shown that the optimal rate allocation depends on the transmission conditions and can therefore be any point in the SW region. It is therefore of interest to construct DSC codes that can achieve any point of the SW region. To that

end, we propose a non-asymmetric SW coder using LDPC codes [4], which have a decoding complexity that is linear with the block length. The decoding is done on the code's *graph* using the sum-product algorithm [5]. Our solution differs from the matrix-inversion-based decoder presented in [6] in the sense that we do not need to recover the difference pattern  $Z$  between the two sources first. We also avoid the drawbacks of the *source splitting*-based methods [7], [8], by the absence of error propagation. And, contrarily to the solution for the dual *Multiple Access Coding* problem [9], we do not need to optimize the degree distribution of the LDPC code, since we want to adapt to any distribution of the sources.

The Bernoulli sources that are usually simulated in the literature are uniformly drawn. However, experimental sources are better described by non-uniformly drawn Bernoulli sources, which have a lower entropy than uniform sources. We raise the problem of asymmetry of the respective minimum achievable rates in the coding of the correlated sources, due to their non-uniformity. Actually if the correlation channel is described by  $Y = X \oplus Z$ , where  $Z$  is independent of  $X$ , then the noise is *additive* for  $X$  and *predictive* for  $Y$ . This implies  $H(Y) \geq H(X)$ , and *a fortiori*  $H(X|Y) \leq H(Y|X)$ . In [10], we presented the non-asymmetric coding of uniform sources using Convolutional and Turbo codes, and derived a necessary and sufficient condition to recover the two sources under turbo decoding. Here, we consider non-uniform sources, and derive a necessary and sufficient condition to recover the sources under the Message-Passing (MP) decoding of an LDPC code. The DSC decoder that we propose in this paper accounts for the non-uniformity of the sources, while being able to reach any point in the SW rate region.

The remainder of this paper is structured as follows. Section II reviews the challenges behind the non-asymmetric coding of two non-uniform sources, and exhibits the achievable rates. Our proposed non-asymmetric decoder, based on LDPC codes, is presented in section III. In section IV, we give the condition on the code's structure to recover both sources. Finally, section V describes the simulation setup, and compares the performance of our coder with the one described in [6].

## II. CHALLENGES BEHIND THE NON-ASYMMETRIC SLEPIAN-WOLF CODING OF BERNOULLI SOURCES

Let  $X \sim \mathcal{B}(p_X)$  denote a binary variable, Bernoulli distributed with parameter  $p_X = \mathbb{P}(X = 1)$ . Another source  $Y$  is correlated to  $X$  s.t.  $\mathbb{P}(Y \neq X) = p$  ( $p < 0.5$ ). Let  $Z$  represent the correlation noise between the sources. Let the

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).

$N$ -long vectors  $\mathbf{x} = x_1^N, \mathbf{y} = y_1^N, \mathbf{z} = z_1^N$  be the respective realizations of  $\mathbf{X} = X_1^N, \mathbf{Y} = Y_1^N, \mathbf{Z} = Z_1^N$ .

#### A. The asymmetric setup: correlation channels and bounds

The asymmetric setup corresponds to the SW problem where one source is available at the decoder. In [2], Wyner showed that binary linear codes offer an optimal solution to this problem with a scheme called *the syndrome approach*. More precisely, consider an  $(N, K)$  channel code  $\mathcal{C}$  defined by its  $(N-K) \times N$  parity check matrix  $\mathbf{H}$ :  $\mathcal{C} = \{\mathbf{x} : \mathbf{H}\mathbf{x} = \mathbf{0}\}$ . This code defines a partition of the  $N$ -length sequences into cosets, where all sequences in a coset share the same syndrome:  $\mathcal{C}_s = \{\mathbf{x} : \mathbf{H}\mathbf{x} = \mathbf{s}\}$ . To encode  $\mathbf{x}$ , the encoder transmits its syndrome  $\mathbf{s}_x = \mathbf{H}\mathbf{x}$ , achieving a compression ratio of  $N : (N - K)$ . At the decoder,  $\mathbf{y}$  is sent at its entropy rate  $H(Y)$  and can therefore be retrieved. Then, the estimated sequence  $\hat{\mathbf{x}}$  is found as the closest sequence to  $\mathbf{y}$  with syndrome  $\mathbf{s}_x$ . This search has been implemented in [11] using systematic Convolutional codes and systematic Turbo codes, and using LDPC codes in [4].

In all previous contributions [6]-[12], the correlation model between the sources  $X$  and  $Y$  is modeled as a BSC. More precisely,  $Z$  is a Bernoulli variable independent of the source  $X$ , and  $Y$  is obtained as  $Y = X \oplus Z$ . In the following, we refer to this model as the *additive BSC*, and introduce a novel correlation model called *predictive channel*. We now define these channels formally, and derive some properties.

**Definition 1.** An  $(X, Y, p)$  additive BSC is a channel with binary input  $X$ , binary output  $Y$ . The noise  $Z \sim \mathcal{B}(p)$  is independent of the channel input, and the channel output is obtained by  $Y = X \oplus Z$ .

The classical BSC is an additive BSC. The concavity of the entropy in the probability imposes  $H(Y) \geq H(X)$ ; and  $H(X|Y) = H(Z) - [H(Y) - H(X)]$  since  $X$  is independent of  $Z$ . Therefore, the achievable rate for the asymmetric coding of the non-uniform source  $X$  is decreased by  $H(Y) - H(X)$  with respect to the uniform case, where  $H(Y) = H(X)$ .

In this case,  $Y$  is also non-uniform, and:

$$p_Y = p_X(1 - p) + (1 - p_X)p \quad (1)$$

**Definition 2.** An  $(X, Y, p)$  predictive BSC is a channel with binary input  $X$ , binary output  $Y$ . The noise  $Z \sim \mathcal{B}(p)$  is independent of the channel output s.t.  $X = Y \oplus Z$ .

The predictive model corresponds to the case where  $Y$  is a prediction of  $X$ .  $Z$  is then an innovation noise independent of  $Y$ . Therefore,  $H(X) \geq H(Y)$ ; and  $H(X|Y) = H(Z)$  since  $X$  is dependent of  $Z$ ; the achievable rate for the coding of  $X$  is the same regardless of the non-uniformity of  $X$ , and no rate gain is expected.

In the remainder of this paper, we assume that the noise  $Z$  is independent of  $X$ , and the other source is derived by  $Y = X \oplus Z$ .

#### B. Channel asymmetry in the non-asymmetric setup

In the non-asymmetric problem, neither  $X$  nor  $Y$  is available at the decoder: the two sources are compressed at rates  $R_X$  and  $R_Y$  fulfilling the constraints  $R_X \in [H(X|Y), H(X)]$ ,

$R_Y \in [H(Y|X), H(Y)]$ , and  $R_X + R_Y \geq H(X, Y)$ ; they have to be decoded jointly. Two non-asymmetric schemes have been proposed: in [13], [14], the asymmetric code is partitioned into two subcodes, one for each source, whereas the original code is used in [6], [12]. In both approaches, first the difference pattern,  $\mathbf{z} = \mathbf{x} \oplus \mathbf{y}$ , is estimated, then the sources are recovered. In this *two step* procedure, error propagation can occur if the error pattern  $\mathbf{z}$  is not correctly estimated [10].

Since  $X$  is independent of  $Z$ , and  $Y = X \oplus Z$ , the BSC is additive for  $X$  and predictive for  $Y$ . Consequently, the problem is not symmetric in the two sources. When coding non-uniform Bernoulli sources, that asymmetry implies that the non-uniformity of  $X$  decreases  $H(X|Y)$ , but the non-uniformity of  $Y$  does not decrease  $H(Y|X)$ , with respect to the case where  $X$  and  $Y$  are uniform.

#### C. The non-asymmetric encoding

Let  $\mathbf{H}$ , of size  $(N-K) \times N$ , denote the matrix of the channel code  $\mathcal{C}$ . Prior to the encoding, we prepare  $\mathbf{H}$  so as to isolate two parts “ $\mathbf{A}$ ” and “ $\mathbf{B}$ ”, of respective sizes  $(N-K) \times K$  and  $(N-K) \times (N-K)$ , as described in [10].  $\mathbf{B}$  is chosen so as to be composed of  $(N-K)$  independent columns of  $\mathbf{H}$ ;  $\mathbf{B}$  is thus invertible, and we denote  $\mathbf{B}^{-1}$  its inverse.  $\mathbf{A}$  is composed of the  $K$  remaining columns in  $\mathbf{H}$ . Let  $k' \in [0, K]$  be an arbitrary integer.

The encoding of  $\mathbf{x}$  consists into transmitting its  $(N-K)$ -long syndrome  $\mathbf{s}_x = \mathbf{H}\mathbf{x}$  and  $k'$  systematic bits  $x_1^{k'}$  corresponding to  $k'$  columns of  $\mathbf{A}$ . The encoding of  $\mathbf{y}$  consists into transmitting  $\mathbf{s}_y = \mathbf{H}\mathbf{y}$  and the  $(K-k')$  systematic bits  $y_{k'+1}^K$  corresponding to the remaining  $(K-k')$  columns of  $\mathbf{A}$ . The rates of  $X$  and  $Y$  are thus  $R_X = \frac{N-K+k'}{N}$  and  $R_Y = \frac{N-k'}{N}$ . When  $k'$  varies from 0 to  $K$ ,  $R_X$  (respectively  $R_Y$ ) varies from  $H(X|Y)$  (resp.  $H(Y)$ ) to  $H(X)$  (resp.  $H(Y|X)$ ).

#### D. The matrix-inversion-based non-asymmetric decoding

This procedure was first proposed in [6]. As the channel code is linear, the syndrome of  $\mathbf{z}$  is  $\mathbf{s}_z = \mathbf{s}_x \oplus \mathbf{s}_y$ . The decoding first assesses the difference pattern  $\mathbf{z}$  between  $\mathbf{x}$  and  $\mathbf{y}$ , by searching the word of minimal weight in the coset of syndrome  $\mathbf{s}_z$ . We note  $\hat{\mathbf{z}}$  the decoded error pattern. Then, as  $\mathbf{s}_x = \mathbf{H}\mathbf{x} = [\mathbf{A} \ \mathbf{B}]\mathbf{x} = \mathbf{A}\mathbf{x}_1^{k'} \oplus \mathbf{B}\mathbf{x}_{k'+1}^N$ ,  $\hat{\mathbf{x}}$  and  $\hat{\mathbf{y}}$  are rebuilt as:

$$\hat{\mathbf{x}} = \begin{pmatrix} \hat{x}_1^{k'} = x_1^{k'} \\ \hat{x}_{k'+1}^K = y_{k'+1}^K \oplus \hat{z}_{k'+1}^K \\ \hat{x}_{K+1}^N = \mathbf{B}^{-1}(\mathbf{A}\hat{x}_1^{k'} \oplus \mathbf{s}_x) \end{pmatrix}$$

$$\hat{\mathbf{y}} = \begin{pmatrix} \hat{y}_1^{k'} = x_1^{k'} \oplus \hat{z}_1^{k'} \\ \hat{y}_{k'+1}^K = y_{k'+1}^K \\ \hat{y}_{K+1}^N = \mathbf{B}^{-1}(\mathbf{A}\hat{y}_1^{k'} \oplus \mathbf{s}_y) \end{pmatrix}$$

If the decoding of  $\mathbf{z}$  is successful, the matrix-inversion-based decoding is able to retrieve the original values of the sources since  $\mathbf{B}$  is invertible. Anyhow, this matrix-inversion has some limitations. More precisely:

- It does not exploit the non-uniformity of the sources, so the SW bound of  $X$  is not reached;

- It suffers from the error propagation phenomena reported in [10], since the decoding cannot take into account the uncertainty from the decoding of  $\mathbf{z}$ ;
- It does not allow to take into account the BSC type between the two sources (additive or predictive).

The proposed LDPC-based non-asymmetric decoder is described in the following section. We have designed it so as to take into account the non-uniformity of the sources. The error propagation phenomena is also dealt with, since the decoding is not dependent on the error pattern  $\mathbf{z}$ . Finally, the type of the BSC between the sources is taken into account.

### III. THE PROPOSED NON-ASYMMETRIC DECODING

The decoder must find the best estimates  $\hat{\mathbf{x}}$  and  $\hat{\mathbf{y}}$  given their syndromes  $\mathbf{s}_x, \mathbf{s}_y$ , their systematic parts  $x_1^{k'}, y_{k'+1}^K$ , and knowing *a priori* that  $X$  and  $Y$  are non-uniform binary sources with parameters  $p_X$  and  $p_Y$ . This amounts to solving the maximization problem:  $\forall n \in [1, N]$

$$\begin{cases} \hat{x}_n = \arg \max_x \mathbb{P}(X_n = x | x_1^{k'}, y_{k'+1}^K, \mathbf{s}_x, \mathbf{s}_y) \\ \hat{y}_n = \arg \max_y \mathbb{P}(Y_n = y | x_1^{k'}, y_{k'+1}^K, \mathbf{s}_x, \mathbf{s}_y) \end{cases} \quad (2)$$

The computation of the *a posteriori* probabilities (APP) for this problem is too complex, but it can be approached with a MP algorithm. We now present the graph that shows the factorization of the APP.

#### A. Graph for the decoding algorithm

In the context of DSC, an  $(N, K)$ -LDPC code  $\mathcal{C}$  can be represented either by its *parity-check matrix*  $\mathbf{H} = (h_{mn})$  of size  $(N - K) \times N$  or by a *bipartite graph*. The bipartite graph is composed of  $(N - K)$  check nodes and  $N$  variable nodes, representing respectively the syndrome symbols and the source symbols.  $h_{mn} = 1$  if the  $m$ -th check node is connected to the  $n$ -th variable node. Moreover, the correlation between the sources  $X$  and  $Y$  is modeled by a check node called the *BSC node* and represents the modulo-2 sum  $Y = X \oplus Z$ . The graph that describes all the dependencies among the variables is shown in Fig. 1.

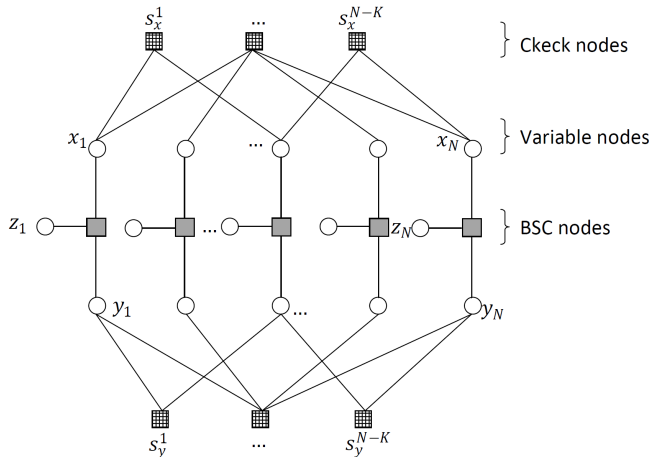


Fig. 1. Joint graph describing the *joint* Slepian-Wolf decoding.

#### B. Definition of the messages

Consider the following notation and definition of the messages that are passed in the graph for the decoding procedure.

- $d_{xn}$ : degree of  $x_n$ ,  $d_{sm}$ : degree of  $s_m$ .
- $I(x_n)$  (resp.  $I(y_n), I(z_n)$ ): *intrinsic* information for the node  $x_n$  (resp.  $y_n, z_n$ ).
- $E_{n,e}, e \in [1, d_{xn}]$ : messages passed from the variable nodes, on their  $e$ -th edge, to the check nodes.
- $Q_{m,e}, e \in [1, d_{sm}]$ : messages passed from the check nodes, on their  $e$ -th edge, to the variable nodes.
- $V(x_n)$ : messages passed from the variable nodes to the BSC nodes.
- $B(x_n)$  (resp.  $B(y_n)$ ): messages passed from the BSC nodes to the variable nodes  $x_n$  (resp.  $y_n$ ).

All the messages are Log-Likelihood Ratio (LLR). They are labeled (*in*) or (*out*) if they respectively come *to* or *from* the considered node.

#### C. Update rules

Now, we describe the messages that are passed between the nodes of the graph, and show how they are updated through the joint sum-product decoding, for non-uniform sources. The factorization of the probabilities (2) shows that:

##### 1) Intrinsic information computation:

$$I(x_n) = \begin{cases} (1 - 2x_n) \cdot \infty, \forall n \in [1, k'] \\ \log \left( \frac{1 - p_X}{p_X} \right), \forall n \in [(k' + 1), N] \end{cases}$$

$$I(y_n) = \begin{cases} 0, \forall n \in [1, k'] \cup [(K + 1), N] \\ (1 - 2y_n) \cdot \infty, \forall n \in [(k' + 1), K] \end{cases}$$

$$I(z_n) = \log \left( \frac{1 - p}{p} \right)$$

##### 2) Messages from the variable nodes to the check nodes:

$$E_{n,e}^{(out)} = I(x_n) + \sum_{k=1, k \neq e}^{d_{xn}} E_{n,k}^{(in)} + B(x_n)$$

##### 3) Messages from the check nodes to the variable nodes:

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

##### 4) Messages from the variable nodes to the BSC nodes:

$$V(x_n) = I(x_n) + \sum_{k=1}^{d_{xn}} E_{n,k}^{(in)}$$

##### 5) Messages from the BSC nodes to the variable nodes:

$$B(x_n) = 2 \tanh^{-1} \left[ \tanh \left( \frac{V(y_n)}{2} \right) \tanh \left( \frac{I(z_n)}{2} \right) \right]$$

$$B(y_n) = 2 \tanh^{-1} \left[ \tanh \left( \frac{V(x_n)}{2} \right) \tanh \left( \frac{I(z_n)}{2} \right) \right]$$

As the expressions of  $E$ ,  $Q$  and  $V$  are similar for  $X$  and  $Y$ , only the update rules for  $X$  have been described.

#### IV. CONDITION FOR THE RECOVERY OF BOTH SOURCES

The soft decoding algorithm presented in section III does not perform exact computation of the APP. Instead, the proposed sub-optimal MP algorithm may not be able to recover the sources. In the following, we derive necessary and sufficient conditions to recover the sources.

Let us first consider the asymmetric case.  $K$  symbols of a source (say  $X$ ) as well as its  $(N - K)$  syndrome bits and the  $(N - K)$  syndrome bits of the source  $Y$ , are available. First, the remaining  $(N - K)$  symbols of  $X$ , corresponding to the part  $\mathbf{B}$  of  $\mathbf{H}$ , have to be decoded by a BEC MP algorithm. This is similar to the problem of solving a linear system of  $(N - K)$  equations with  $(N - K)$  unknowns. In that case, the following lemma stands:

**Lemma 1.** *Let  $\mathbf{x} \in \mathbb{F}_2^N$  satisfy a set of  $(N - K)$  linear equations defined by a matrix  $\mathbf{H}$  of size  $(N - K) \times N$  s.t.  $\mathbf{H}\mathbf{x} = \mathbf{s}_x$ . Assume that the first  $K$  bits of  $\mathbf{x}$  (denoted  $x_1^K$ ) are known and that the last  $(N - K)$  bits ( $x_{K+1}^N$ ) are unknown. The system of linear equations can be rewritten as  $\mathbf{H}\mathbf{x} = [\mathbf{A} \ \mathbf{B}][x_1^K \ x_{K+1}^N]^T$  where  $\mathbf{B}$  is assumed to be a square invertible matrix. A necessary and sufficient condition to recover the unknowns, with the BEC message-passing (MP) algorithm, is that  $\mathbf{B}$  is triangular (up to a permutation of its columns).*

*Proof:* First, recall that the BEC MP algorithm is equivalent to greedily checking whether any of the parity-constraints can solve a yet unknown value from already known ones. It follows that the condition is sufficient. Conversely, assume that we can recover the unknowns under the BEC MP algorithm. To start the process, there must be at least one equation with only one unknown. Moreover there is at most one equation with the same unknown (if not,  $\mathbf{B}$  would not be invertible). This unknown can therefore be recovered and we now have to solve a system of  $(N - K - 1)$  equations and  $(N - K - 1)$  unknowns. The rest of proof follows by induction since at each step of the algorithm the number of equations and the number of unknowns is reduced by one. ■

Lemma 1 derives a necessary and sufficient condition for the exact recovery of the source  $X$ . The second source  $Y$  can now be recovered since its syndrome is known and since the source  $X$  is available. Therefore Lemma 1 derives the necessary and sufficient condition for the estimation of the sources  $X$  and  $Y$  in the *asymmetric* case, when decoded with the proposed algorithm presented in Section III.

We now consider the non-asymmetric problem. Note that the asymmetric case is the best case scenario of the non-asymmetric one. Thus, the condition in Lemma 1 becomes a necessary condition for the recovery of the two sources.

#### V. EXPERIMENTAL SETUP AND SIMULATION RESULTS

##### A. Code design

According to Lemma 1, the invertible part  $\mathbf{B}$  of the channel code's parity-check matrix has to be triangular in order to ensure the decoding. To design our LDPC code, we take ideas from Irregular Repeat-Accumulate (IRA) codes [15], and

follow the design rules presented in [16]. It imposes the shape in Fig. 2 to the parity-check matrix, with a pattern of weight-two columns forming two diagonal lines of ones, and a pattern of weight-three columns ensuring a good degree distribution for the code.

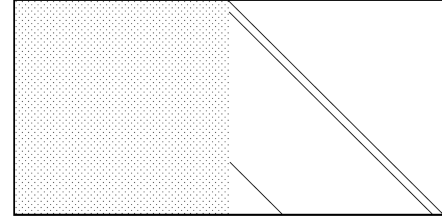


Fig. 2. Shape of the designed LDPC code's parity-check matrix, inspired by the structure of IRA codes. The columns forming the weight-two and weight-three columns are chosen as the triangular part  $\mathbf{B}$ .

That structure is the best way to impose structured weight-two and weight-three columns without adding cycles of short lengths amongst the variable nodes. We take the concatenation of these columns as our part  $\mathbf{B}$ . We generate a rate- $\frac{1}{2}$  LDPC code of block length 1000. More precisely, we first find the optimal *variable degree distribution*  $\Lambda(x)$ , and *check degree distribution*  $\Phi(x)$ , of a rate- $\frac{1}{2}$  LDPC code by density evolution [17]. Then, the LDPC code is obtained using the Progressive Edge Growth (PEG) principle [18] by imposing the structure shown in Fig.2 to the generated parity-check matrix.

The code we have generated has the following variable degree distribution:  $\Lambda(x) = 0.483949x + 0.294428x^2 + 0.085134x^5 + 0.074055x^6 + 0.062433x^{19}$ . Note that for this rate- $\frac{1}{2}$  LDPC code, the proportion of variable nodes of degree two and three  $\Lambda_2 + \Lambda_3 = 0.7784 \geq 0.5$  ensures that there are enough weight-two and weight-three columns to form the part  $\mathbf{B}$  that we need. The check degree distribution is  $\Phi(x) = 0.741935x^7 + 0.258065x^8$ . Note that these degree distributions do not depend on the source statistics. The same code is used for all the simulations presented in this paper.

##### B. Performance of the non-asymmetric codec

Now, we turn to comparing the performance of our codec and the matrix-inversion-based decoder for different values of  $p$ . The source  $X$  is non-uniform, with  $p_X = 0.12$ ,  $Z$  is a BSC of cross-over probability  $p$ , and  $Y$  is obtained by  $Y = X \oplus Z$ . For each value of  $p$ , 10 values of  $k'$  are considered, and for each value of  $k'$ ,  $2 \cdot 10^4$  realizations  $\mathbf{x}$  are tested, the obtained BER have been averaged over all the values of  $k'$ .

The performance of *four* systems are presented in Fig. 3:

- (1) Asymmetric, for the decoding of  $X$  (additive BSC).
- (2) Asymmetric, for the decoding of  $Y$  (predictive BSC).
- (3) Non-asymmetric, with the matrix inversion [6]; (3a) for the decoding of  $X$  and (3b) for the decoding of  $Y$ .
- (4) Non-asymmetric, proposed in this paper; (4a) for the decoding of  $X$  and (4b) for the decoding of  $Y$ .

The decoding involving the matrix inversion is the worst solution, since no measures are taken to prevent the error propagation between the decoding of  $Z$  and the decoding of the sources, that is why its BER in the non-asymmetric setup is worse than the BER in the asymmetric one. The joint

method that we propose, with cross message passing during the decoding of the two sources, outperforms the matrix-inversion based method. The statistics of the sources are taken into account for the decoding, as well as the BSC model between the sources.

### C. Behavior analysis of the non-asymmetric SW codec

In this section, we compare the behavior of our codec and the behavior of the matrix-based codec in function of  $k'$ , for a given value of  $H(p) = 0.62$  ( $p = 0.1541$ ). 10 values of  $k'$ , varying from 0 to  $K$ , are considered; and, for each value taken by  $k'$ ,  $2 \cdot 10^4$  realizations of  $X$  are tested. The obtained BER is plotted in Fig. 4 in function of  $\frac{k'}{K}$ .

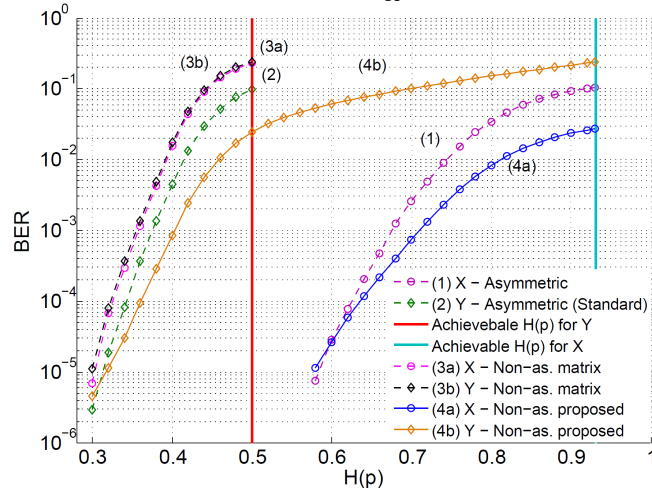


Fig. 3. Performance of the non-asymmetric SW codec, for a non-uniform source  $X \sim \mathcal{B}(p_X = 0.12)$ .  $H(X|Y) = 0.5$  occurs for  $H(p) = 0.93$ , whereas  $H(Y|X) = 0.5$  occurs for  $H(p) = 0.5$ . This difference comes from the BSC between the sources which is not symmetric in the two sources.

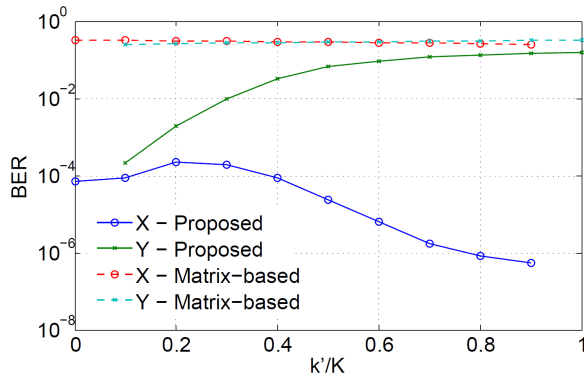


Fig. 4. Performance of the non-asymmetric SW codec in function of  $\frac{k'}{K}$ , for a non-uniform source  $X \sim \mathcal{B}(p_X = 0.12)$  at  $H(p) = 0.62$ .

The results presented in Fig. 4 confirm that the joint decoding of the non-uniform sources  $X$  and  $Y$  is not symmetric in the two sources, as claimed in Section II. Indeed, if the source  $X$  is decoded with a very low BER for all the values of  $k'$ , the source  $Y$  is only well decoded when  $k' = 0$  (which corresponds to the asymmetric case). Besides, the decoding of the two sources is not symmetric in  $k'$ , and their BER is not equal even when  $\frac{k'}{K} = 0.5$ . That comes from the non-uniformities of the sources that are not equal: if  $p_X = 0.12$  is constant,  $p_Y$  depends on  $p$  and is given by (1). Finally, we note that the matrix-inversion based decoding is symmetric in

the two sources and with  $k'$ , since it does not take into account the probabilities of the sources.

## VI. CONCLUSION

We have addressed the problem of non-asymmetric SW coding of non-uniform Bernoulli sources, by designing a joint decoder that can account for the non-uniformity of the sources. We have introduced a BSC, that models the correlation between the sources, which can be *additive* or *predictive* noise. That difference changes the minimum achievable rate for the coding of each source. We stated a necessary and sufficient condition for the recovery of the two sources, that imposes a triangular form to a sub-part of the parity-check matrix. Our solution has thus been implemented using an LDPC code which parity-check matrix structure is designed as the parity-check matrix of an IRA code, in order to match to the condition. We showed that the joint decoder that we have proposed outperforms the performance of the common two-step decoder involving a matrix inversion.

## REFERENCES

- [1] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Th.*, vol. 19, no. 9, pp. 471–480, 1973.
- [2] A. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inf. Th.*, vol. 20, pp. 2–10, 1974.
- [3] A. Roumy and D. Gesbert, "Optimal matching in wireless sensor networks," *IEEE Selected Topics Sig. Proc.*, vol. 1, pp. 725–735, 2007.
- [4] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Comm. Lett.*, vol. 6, no. 10, pp. 440–442, 2002.
- [5] F. R. Kschischang, B. J. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Th.*, vol. 47, no. 2, pp. 498–519, 2001.
- [6] N. Gehrgh and P. L. Dragotti, "Symmetric and asymmetric Slepian-Wolf codes with systematic and non-systematic linear codes," *IEEE comm. let.*, vol. 9, no. 1, pp. 61–63, Jan 2005.
- [7] B. Rimoldi and R. Urbanke, "Asynchronous Slepian-Wolf Coding Via Source-Splitting," in *IEEE Int. Symp. Inf. Th.*, 1997, p. 271.
- [8] T. P. Coleman, A. H. Lee, M. Mdard, and M. Effros, "Low-complexity approaches to Slepian-Wolf near-lossless distributed data compression," *IEEE Trans. Inf. Th.*, vol. 52, pp. 3546–3561, 2006.
- [9] A. Amraoui, S. Dusad, and R. Urbanke, "Achieving General Points in the 2-User Gaussian MAC Without Time-Sharing or Rate-Splitting by Means of Iterative Coding," in *IEEE Int. Symp. Inf. Th.*, 2002, p. 334.
- [10] C. Herzet, V. Toto-Zarasoia, and A. Roumy, "Error resilient non-asymmetric Slepian-Wolf coding," in *ICC*, Jun 2009, pp. 1–5.
- [11] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction," *IEEE Data Comp. Conf.*, pp. 158–167, March 1999.
- [12] P. Tan and J. Li, "A practical and optimal symmetric Slepian-Wolf compression strategy using syndrome formers and inverse syndrome formers," in *Allerton Conf. Comm., Control and Comp.*, Sept 2005.
- [13] S. S. Pradhan and K. Ramchandran, "Distributed source coding: symmetric rates and applications to sensor networks," in *DCC*, March 2000, pp. 363–372.
- [14] V. Stankovic, A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Design of Slepian-Wolf codes by channel code partitioning," in *IEEE Data Comp. Conf.*, 2004, pp. 302–311.
- [15] H. Jin, D. Khandekar, and R. J. McEliece, "Irregular repeat-accumulate codes," *ISTC*, pp. 1–8, Sept 2000.
- [16] S. J. Johnson and S. R. Weller, "Constructions for irregular repeat-accumulate codes," in *IEEE Int. Symp. Inf. Th.*, Sept 2005.
- [17] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Th.*, vol. 47, no. 2, Feb. 2001.
- [18] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular Progressive Edge-Growth Tanner graphs," *IEEE Trans. Inf. Th.*, 2005.