Density Evolution for the Design of Non-Binary Low Density Parity Check Codes for Slepian-Wolf Coding

Elsa DUPRAZ[1], Valentin SAVIN[2] and Michel KIEFFER[3,4,5]

[1] ETIS - ENSEA - Univ. Cergy-Pontoise - CNRS, Cergy-Pontoise, France

[2] CEA-LETI, Minatec Campus, Grenoble, France

[3] L2S - CNRS - SUPELEC - Univ Paris-Sud, Gif-sur-Yvette, France

[4] Partly on leave at LTCI – CNRS Télécom ParisTech Paris, France, [5] Institut Universitaire de France

Abstract

In this paper, we investigate the problem of designing good non-binary LDPC codes for Slepian-Wolf coding using density evolution. In channel coding, if the channel is symmetric, density evolution can be performed assuming that the all-zero codeword was transmitted. Such an assumption does not hold in Slepian-Wolf coding, even if the correlation channel is symmetric, because of the possibly non-uniform source distribution. In this paper, we show that any, even non-symmetric correlation channel in Slepian-Wolf coding is equivalent under density evolution to a symmetric channel in channel coding. Consequently, density evolution with the all-zero codeword assumption can be performed on this equivalent channel in order to obtain the asymptotic performance of an LDPC code in Slepian-Wolf coding. From this equivalence, we are able to determine good code degree distributions for Slepian Wolf coding.

I. INTRODUCTION

In this paper, we consider the lossless coding of a source X with the help of some side information Y available at the decoder only (see Figure 1). This setup is called asymmetric Slepian-Wolf (SW)

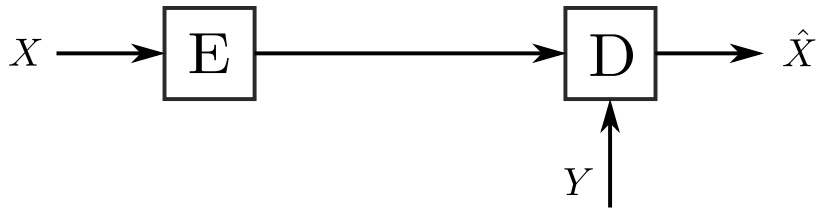

Fig. 1. Asymetric Slepian-Wolf coding

coding [27]. Here, for simplicity, it is referred to as SW coding. For this problem, it is well known that the infimum of achievable rates is given by $H(X|Y)$, the conditional entropy of X knowing Y and several practical coding schemes have been proposed [8], [23], [35]. Most of them are based on channel codes [7], [28], and particularly Low Density Parity Check (LDPC) codes [10], [19], [21]. In source coding, the source symbols are in general non-binary (for example the pixels or the quantized coefficients of the transformed blocks of an image). A usual coding solution is to transform the non-binary symbols into bits and to encode the bit planes independently with binary LDPC codes. To avoid a performance loss, the dependency between bit planes has to be taken into account at the decoder [17], [33], at the price of a complexity increase. In this paper, in order to avoid this operation, we consider directly non-binary LDPC codes [11].

Many efforts have been made in channel coding for the design of good LDPC codes. In particular, density evolution techniques have been developed both for binary [24], [25], [31] and non-binary [1], [18] codes. Denote $\mathcal{C}(\lambda, \rho)$ the ensemble of codes of variable and check node degree distributions $\lambda(x)$ and $\rho(x)$. From an asymptotic analysis, density evolution gives an evaluation of the average error rate over $\mathcal{C}(\lambda, \rho)$ for a given channel of input U and output W described by the conditional distribution $P(W|U)$. Optimization techniques such as differential evolution [30] can then be implemented in order to obtain good degree distributions for the considered channel. Although the issue of constructing properly the coding matrix at finite length remains [22], it constitutes a good starting point for practical code design.

In SW coding, from the joint probability distribution $P(X, Y)$, one could think of identifying the

correlation channel $P(Y|X)$ and then simply applying the standard density evolution derived for channel coding. Unfortunately, as pointed out in [2], [4], a good LDPC code for channel coding is not necessarily good for SW coding. As an example, consider the case of a Binary Symmetric Channel (BSC) defined by $P(W=1|U=0)=P(W=0|U=1)=p$. The capacity of the BSC is $C_{W|U}=\max_{p(u)} I(U;W)=1-H(p)$ [9], *i.e.*, the max is achieved when U is distributed uniformly. In SW coding, consider as well a binary symmetric correlation channel $P(Y=1|X=0)=P(Y=0|X=1)=p$. Unfortunately in this case, the source X is not necessarily uniformly distributed and $H(X|Y)\leq H(p)$ with equality if and only if X is distributed uniformly. As a numerical example, for $p=0.2$ and $P(X=0)=0.5$, $H(p)=0.72$ bit/symbol while for $P(X=0)=0.2$, $H(X|Y)=0.54$ bit/symbol. Consequently, the channel coding scheme and the SW coding scheme require codes of different rate. On the other hand, in channel coding, the decoding error probability for a symmetric channel does not depend on the input codeword [18], [24]. This allows considering that the all-zero codeword was transmitted and greatly simplifies the density evolution. In SW coding, this result holds also if the source is distributed uniformly and the correlation channel $P(Y|X)$ is symmetric. However, these assumptions are difficult to satisfy in source coding, particularly for the uniform source distribution.

For binary LDPC codes, [4] shows that for any joint distribution $P(X,Y)$, there exists an equivalent joint distribution $P(U,W)$ such that U is distributed uniformly and $P(W|U)$ is symmetric. The equivalent $P(U,W)$ is such that the average error rate over $\mathcal{C}(\lambda,\rho)$ is the same for $P(X,Y)$ and $P(U,W)$. Furthermore, $H(U|W)=H(X|Y)$. Thus for any joint distribution $P(X,Y)$ in SW coding, it suffices to identify the equivalent $P(U,W)$ and then to perform standard density evolution on $P(W|U)$. In this paper, we generalize this result to codes in GF(q), the Galois Field of size q. In particular, we derive the equivalence and explain how to perform density evolution for non-binary LDPC codes in SW coding. More in details, the contributions of the papers are as follows.

1) In channel coding, we derive an analytical form of the density evolution for symmetric channels. The obtained analytic expression is difficult to write in an explicit form, and is not convenient

for practical density evolution. However, it allows us to express the equivalence between channel and SW coding.

2) We also derive an analytical form of the density evolution in SW coding for any joint probability distribution $P(X,Y)$
3) From the two previous recursions, we derive the expression of the joint distribution $P(U,W)$ equivalent to $P(X,Y)$ under density evolution.
4) We present the example of a q-ary symmetric correlation channel $P(Y|X)$ where X is not necessarily distributed uniformly. We derive the equivalent channel and give some numerical results on the performance of some optimized codes.

The paper is organized as follows. Section II presents the related works. Section III introduces the notations and recalls some results on Galois Fields. Section IV restates the non-binary LDPC decoding algorithm for SW coding. Section V expresses the density evolution for SW coding and derives the channel equivalence in the non-binary case. Section VI presents the example of the q-ary symmetric channel..

II. Related Works

Binary LDPC codes have been used for SW coding in [3], [5], [19], [21], [29] and references therein. In all of the above cases, the LDPC decoder consists of a message passing procedure referred to as the sum-product algorithm. In the same way, [32] proposes to use non-binary LDPC codes and derives the decoding algorithm expressions. Nevertheless these works do not provide a solution for the design of good non-binary LDPC codes for SW coding.

On the other hand, density evolution was initially introduced in [24] for binary symmetric channels and then used in [25] for irregular code optimization. The case of binary non-symmetric channels was further investigated in [31]. All these works give an analytic expression of the density evolution. Then, [4] considered density evolution for binary SW coding and non-symmetric channels. In [4], an equivalence between SW coding and channel coding under density evolution is derived.

For non-binary LDPC codes, the exact density evolution equations are only known for erasure channels [26]. Alternatively, approximation methods have been proposed, *e.g.*, the density evolution under Gaussian approximation, which can be applied for the AWGN channel model, for binary [6] as well as for non-binary LDPC codes [18]. Then, [1] considered density evolution for coset non-binary LDPC codes. In this case, the channels are not necessarily symmetric, because it is shown that the coset has a symmetrizing effect. As before, no analytic expression of the density evolution is given, except with the Gaussian approximation. Although SW codes can be seen as particular coset LDPC codes, [1] considers channel coding and consequently fixed input symbols distribution. To finish, [14] shows that, if the all-zero codeword assumption holds, density evolution in channel coding can be approximated through the use of Monte-Carlo methods (referred to as MC-DE).

III. Notations and preliminaries

In the following, upper case letters, *e.g.*, X, denote random variables whereas lower case letters, x, represent their realizations. Vectors, *e.g.*, $\mathbf{X} = \{X_k\}_{k=1}^{n}$, are in bold. When it is clear from the context that the distribution of a random variable X_k does not depend on k, the index k is omitted. The imaginary unit is denoted i. The Kronecker function is denoted $\delta(x)$, *i.e.*, $\delta(x) = 1$ if $x = 0$, $\delta(x) = 0$ otherwise. In the following, $\bigotimes$ stands for the convolution product (not to be confused with $\otimes$, the multiplicative operator in GF(q)) and $\circ$ is the composition operator. In SW coding (see Figure 1), the source X to be compressed and the SI Y available at the decoder produce sequences of independent and identically distributed (*i.i.d.*) discrete symbols $\{X_n\}_{n=1}^{+\infty}$ and $\{Y_n\}_{n=1}^{+\infty}$ respectively. The realizations of the random variable X belong to GF(q) with $q = \kappa^\alpha$ and κ is prime. The realizations of Y belong to a discrete alphabet $\mathcal{Y}$. Denote $P(X = x) = p_x$ where $0 < p_x < 1$ and assume $\forall (x, y)$, $0 < P(Y = y|X = x) < 1$.

A. Operations in GF(q)

In order to introduce some notations and conventions that will be used in the paper, we recall here some standard definitions related to Galois Fields. See [20, Chapter 4] for more details. Define $\mathbb{Z}_\kappa[D]$

as the set of polynomials with coefficients in $\mathbb{Z}/\kappa\mathbb{Z}$ and let $P(D) \in \mathrm{G}_\kappa[D]$ an irreducible polynomial of degree α. Define GF(q) $= \mathrm{G}_\kappa[D]/P(D)$. It follows that every element of GF(q) can be uniquely represented by a polynomial of degree less than α, *i.e*, $\forall P_a(D) \in$ GF(q),

$$P_a(D) = a_0 + a_1 D + \ldots a_{\alpha-1} D^{\alpha-1}. \tag{1}$$

where $a_k \in \{0 \ldots \kappa - 1\}$ As a consequence, one can define a one-to-one correspondence between $\{0, \ldots, q-1\}$ and GF(q) by associating to each $P_a(D) \in$ GF(q) a value $a \in \{0, \ldots, q-1\}$. Remarking that any $a \in \{0, \ldots, q-1\}$ can be uniquely decomposed as

$$a = a_0 + a_1 \kappa + \ldots a_{\alpha-1} \kappa^{\alpha-1}, \tag{2}$$

a is by convention associated to the polynomial $P_a(D)$ (1). In the following, $\oplus$, $\ominus$, $\otimes$, $\oslash$ are the usual operators in GF(q). By an abuse of notation, we will denote by a both its integer value and the corresponding element of GF(q). Thus, for any real or complex value x, x^a is evaluated from the integer version of a, but in the expression $x^{a \oplus b}$, $a \oplus b$ is performed in GF(q). Throughout the remaining of the paper, we denote by r the κ-th root of unity defined by $r = \exp(\frac{2\mathrm{i}\pi}{\kappa})$. With the above convention, one can show that $r^{a \oplus b} = r^a r^b$.

B. Probability evaluation in GF(q)

Let Z be a random variable with values in GF(q). Denote $\mathbf{p}$ the probability vector of size q with k-th component $p_k = P(Z = k)$ and $0 < p_k < 1$. Denote $\mathbf{m}$ the message vector of size q with k-th component $m_k = \log \frac{p_0}{p_k} = \log \frac{P(Z=0)}{P(Z=k)}$. From the previous expression, one has $p_k = \frac{e^{-m_k}}{\sum_{k'=0}^{q-1} e^{-m'_k}}$. As part of the LDPC decoder consists of the evaluation of the probability of linear combinations of random variables, we first express here the probabilities of $Z \oplus a$, $Z \otimes a$, where $a \in$ GF(q), and of $Z_1 \oplus Z_2$. Note that the operators we describe here to realize these evaluations were initially introduced in [1] and [18]. We restate them here to make the paper more self contained.

Denote $\mathbf{p}^{\times a}$ and $\mathbf{m}^{\times a}$ ($\forall a \in$ GF(q)$\backslash\{0\}$), $\mathbf{p}^{+a}$ and $\mathbf{m}^{+a}$ ($\forall a \in$ GF(q)) the probability and message vectors associated to $Z \otimes a$ and $Z \oplus a$. By definition, $\forall a \neq 0$, $p_k^{\times a} = P(Z \otimes a = k) = P(Z = k \oslash a)$

and

$$m_k^{\times a} = \log \frac{P(Z \otimes a = 0)}{P(Z \otimes a = k)} = \log \frac{P(Z = 0)}{P(Z = k \oslash a)} \ . \tag{3}$$

Let $W[a]$ be a $q \times q$ matrix such that $\forall k, j = 0, \ldots, q-1$, $W_{k,j}[a] = \delta(a \otimes j \ominus k)$. Then, $\mathbf{p}^{\times a} = W[a]\mathbf{p}$ and $\mathbf{m}^{\times a} = W[a]\mathbf{m}$. On the other hand, $p_k^{+a} = P(Z \oplus a = k) = P(Z = k \ominus a)$ and

$$m_k^{+a} = \log \frac{P(Z \oplus a = 0)}{P(Z \oplus a = k)} = \log \frac{P(Z = \ominus a)}{P(Z = k \ominus a)} \tag{4}$$

Denote $R[a]$ the $q \times q$ matrix such that $\forall k, j = 0, \ldots, q-1$, $R_{k,j}[a] = \delta(a \oplus k \ominus j)$. Denote $\mathcal{A}[a]$ the $q \times q$ matrix such that $\mathcal{A}_{0,0}[a] = 1$ and $\forall k, j = 0, \ldots, q-1$, $(k, j) \neq (0, 0)$, $\mathcal{A}_{k,j}[a] = \delta(a \oplus k \ominus j) - \delta(a \ominus j)$. Then, $\mathbf{p}^{+a} = R[a]\mathbf{p}$ and $\mathbf{m}^{+a} = \mathcal{A}[\ominus a]\mathbf{m}$. Here, two different transforms are needed because of the numerator in (4). The notations $\mathbf{m}^{\times a}$ and $\mathbf{m}^{+a}$ come from [1] while $W[a]$ and $\mathcal{A}[a]$ come from [18].

Now, let Z_1 and Z_2 be two random variables with realizations in GF(q) and probability vectors $\mathbf{p}_1$ and $\mathbf{p}_2$. Then,

$$P(Z_1 \oplus Z_2 = k) = \sum_{j=0}^{q-1} P(Z_1 = j) P(Z_1 \oplus Z_2 = k | Z_1 = j) = \sum_{j=0}^{q-1} p_{1,j} p_{2,k \ominus j} \tag{5}$$

$$:= (\mathbf{p}_1 \overline{\otimes} \mathbf{p}_2)_k \ . \tag{6}$$

The operator $\overline{\otimes}$ represents a discrete convolution product but *does not* correspond to the classical circular convolution product. Consequently, as pointed out in [15], the usual discrete Fourier Transform cannot be used for the evaluation of (5) and there is a need to define an adapted Fourier-like transform $\mathcal{F}$. Let $\mathbf{f} = \mathcal{F}(\mathbf{p})$ and $\mathbf{p} = \mathcal{F}^{-1}(\mathbf{f})$ with from [18],

$$f_j = \sum_{k=0}^{q-1} r^{k \otimes j} p_k \ , \quad p_k = \frac{1}{q} \sum_{j=0}^{q-1} r^{-k \otimes j} f_j \ . \tag{7}$$

Then

$$P(Z_1 \oplus Z_2 = k) = \left(\mathcal{F}^{-1} \left(\mathcal{F}(\mathbf{p}_1) \mathcal{F}(\mathbf{p}_2) \right) \right)_k \ . \tag{8}$$

This expression can easily be generalized to a sum of K elements. A message version of the Fourier-like transform can also be defined as $\mathbf{f} = \tilde{\mathcal{F}}(\mathbf{m})$ and $\mathbf{m} = \tilde{\mathcal{F}}^{-1}(\mathbf{f})$ with

$$f_j = \sum_{k=0}^{q-1} r^{k \otimes j} \frac{e^{-m_k}}{\sum_{k'=0}^{q-1} e^{-m'_k}} \ , \quad m_k = \log \frac{\sum_{j=0}^{q-1} f_j}{\sum_{j=0}^{q-1} r^{-k \otimes j} f_j} \ . \tag{9}$$

Note that if q is a power of 2, then $\mathcal{F}$ becomes the Hadamard transform [12].

IV. LDPC ENCODING AND DECODING

LDPC codes initially introduced for channel coding can also be used for SW coding, after adaptation of the coding process and the decoding algorithm [19], [21]. The SW coding of a source vector $\mathbf{x}$ of length n is performed by producing a vector $\mathbf{s} = H^T\mathbf{x}$ of length $m < n$. The matrix H is sparse, with coefficients in GF(q). In the bipartite graph representing the dependencies between the random variables of $\mathbf{X}$ and $\mathbf{S}$, the entries of $\mathbf{X}$ are represented by Variable Nodes (VN) and the entries of $\mathbf{S}$ are represented by Check Nodes (CN). The set of CN connected to a VN n is denoted $\mathcal{N}_{\mathrm{C}}(n)$ and the set of VN connected to a CN m is denoted $\mathcal{N}_{\mathrm{V}}(m)$. The sparsity of H is determined by the edge-perspective VN degree distribution $\lambda(x)$ and CN degree distribution $\rho(x)$, where

$$\lambda(x) = \sum_{k\geq 2} \lambda_k x^{k-1}, \quad \rho(x) = \sum_{j\geq 2} \rho_j x^{j-1} \tag{10}$$

The constant $0 \leq \lambda_k \leq 1$ is the fraction of edges emanating from a VN of degree k and $0 \leq \rho_j \leq 1$ is the fraction of edges emanating from a CN of degree j. In SW coding, the coding efficiency $r(\lambda, \rho)$ of a code is given by $r(\lambda, \rho) = \frac{m}{n} = \frac{\sum_{j\geq 2} \rho_j / j}{\sum_{k\geq 2} \lambda_k / k}$. A code is said to be regular if the VN and CN have constant degrees d_v and d_c. In this case, $r(d_v, d_c) = \frac{d_v}{d_c}$.

The sum-product LDPC decoder performs an approximate Maximum *A Posteriori* (MAP) estimation of $\mathbf{x}$ from the received codeword $\mathbf{s}$ and the observed side information $\mathbf{y}$ by the mean of message exchange in the bipartite graph. In non-binary channel coding, the sum-product LDPC decoder is described in [18]. We expressed the SW version of the algorithm in [13] and restate it here for the sake of completeness. The initial message for a VN n is denoted $\mathbf{m}^{(0)}(n)$, and its k-th component is

$$m_k^{(0)}(n) = \log \frac{P(X_n = 0 | Y_n = y_n)}{P(X_n = k | Y_n = y_n)}, \ k = 0 \ldots q-1 \ . \tag{11}$$

Note that, here, the messages are expressed as vectors of log-likelihood ratios (LLR). Although exchanged messages may alternatively be represented as vector of probabilities [34], it is more convenient

for our purpose to assume that they are represented as vectors of LLR values. At iteration ℓ, the message $\mathbf{m}^{(\ell)}(m,n)$ from CN m to VN n is

$$\mathbf{m}^{(\ell)}(m,n) = \mathcal{A}[\overline{s}_m]\tilde{\mathcal{F}}^{-1}\left(\prod_{n'\in\mathcal{N}_{\mathrm{V}}(m)\backslash n}\tilde{\mathcal{F}}\left(W\left[\overline{g}_{n'm}\right]\mathbf{m}^{(\ell-1)}(n',m)\right)\right) \tag{12}$$

where the product is componentwise, $\bar{s}_m = \ominus s_m \oslash H_{n,m}$, and $\overline{g}_{n'm} = \ominus H_{n',m} \oslash H_{n,m}$. Note that $\mathcal{A}[\overline{s}_m]$ does not appear in the channel coding version of the algorithm and is specific to SW coding (since in channel coding the syndrome is zero). At a VN n, a message $\mathbf{m}^{(\ell)}(n,m)$ is sent to the CN m and an *a posteriori* message $\tilde{\mathbf{m}}^{(\ell)}(n)$ is computed. They both satisfy

$$\mathbf{m}^{(\ell)}(n,m) = \mathbf{m}^{(0)}(n) + \sum_{m'\in\mathcal{N}_{\mathrm{C}}(n)\backslash m}\mathbf{m}^{(\ell)}(m',n)\ , \tag{13}$$

$$\tilde{\mathbf{m}}^{(\ell)}(n) = \mathbf{m}^{(0)}(n) + \sum_{m'\in\mathcal{N}_{\mathrm{C}}(n)}\mathbf{m}^{(\ell)}(m',n)\ . \tag{14}$$

The channel version of the algorithm has the same VN message computation. From (14), each VN n produces an estimate $\widehat{x}_n^{(\ell)} = \arg\max_k \tilde{m}_k^{(\ell)}(n)$ of x_n. The algorithm ends if $H^{\mathrm{T}}\widehat{\mathbf{x}}^{(\ell)} = \mathbf{s}$ or if $\ell = L_{\max}$, the maximum number of iterations.

The CN message (12) is calculated from linear operators and a componentwise product. Since the probability density of these products may be difficult to derive, we introduce the following transform $\boldsymbol{\gamma}$. The function $\boldsymbol{\gamma}$ applies on vectors of size q and has $k-$th component $\gamma_k : \mathbb{C} \to \mathbb{R}\times[-\pi,\pi]$ with

$$\gamma_k(x_k + \mathrm{i}y_k) = (z_k,t_k) = \begin{cases} \left(\frac{1}{2}\log(x_k^2+y_k^2), \arctan\frac{y_k}{x_k}\right) & \text{if } x_k \geq 0, y_k \neq 0 \\ \left(\frac{1}{2}\log(x_k^2+y_k^2), \arctan\frac{y_k}{x_k} + \pi\right) & \text{if } x_k \leq 0\ ,\ y_k \geq 0 \\ \left(\frac{1}{2}\log(x_k^2+y_k^2), \arctan\frac{y_k}{x_k} - \pi\right) & \text{if } x_k \leq 0\ ,\ y_k < 0\ . \end{cases} \tag{15}$$

where x_k and y_k are real numbers. Note that γ_k can also be seen as a function from $\mathbb{R}^2$ to $\mathbb{R}\times[-\pi,\pi]$. We complete the definition of γ_k by assuming that when $x_k + \mathrm{i}y_k = 0$, the value of t_k is given by the realization of a random variable Θ taking its values in $[0,2\pi]$ and with probability density function $f_\Theta(\theta) = \frac{1}{2\pi}$. The inverse function $\boldsymbol{\gamma}^{-1}$ applies on vectors of size q and has $j-$th component $\gamma_j^{-1} : \mathbb{R}\times[-\pi,\pi] \to \mathbb{C}$ with

$$\gamma_j^{-1}(z_j,t_j) = \exp(z_j)\cos t_j + \mathrm{i}\exp(z_j)\sin t_j\ . \tag{16}$$

The CN to VN equation (12) can then be restated as

$$\mathbf{m}^{(\ell)}(m,n) = \mathcal{A}[\overline{s}_m]\tilde{\mathcal{F}}^{-1}\left(\boldsymbol{\gamma}^{-1}\left(\sum_{n'\in\mathcal{N}(m)\setminus n}\boldsymbol{\gamma}\left(\tilde{\mathcal{F}}\left(W\left[\overline{g}_{n'm}\right]\mathbf{m}^{(\ell-1)}(n',m)\right)\right)\right)\right) . \tag{17}$$

Density evolution consists in the evaluation of the probability densities of the messages at each iteration. The decoding error probability can then be calculated from the probability of the messages giving false estimates $\hat{x}_k$. In this way, the probability densities of $\mathbf{m}^{(\ell)}(n,m)$ and $\tilde{\mathbf{m}}^{(\ell)}(n)$ in (13), (14), are easy to evaluate from the probability densities of the $\mathbf{m}^{(\ell)}(m',n)$ (assuming the $\mathbf{m}^{(\ell)}(m',n)$ are realizations of independent random variables). On the opposite, the probability density of $\mathbf{m}^{(\ell)}(m,n)$ in (12) is difficult to derive because of the componentwise product. That is why we introduced the function $\boldsymbol{\gamma}$ that transforms the product in (12) into a sum.

V. Density evolution

This section evaluates the probability densities of the messages in channel coding and in SW coding. The messages exchanged in the graph during the decoding can be seen as random variables. From the density of the initial messages (11), we want to calculate recursively the probability density of the messages at iteration ℓ, exploiting (13) and (17). For this, several simplifying assumptions can be performed. First, it is assumed that the messages arriving at a node at iteration ℓ are independent. The so-called independence assumption was originally discussed in [25] and proved formally to be reasonable in [31]. The main idea is that the messages are independent if they have been calculated on independent subtrees of the bipartite graph. It is called the cycle-free case. In [31], it is shown that this cycle-free case happens with probability arbitrarily closed to 1 when $n \to \infty$.

The second simplifying assumption is called the all-zero codeword assumption. In channel coding, it is shown to apply only for symmetric channel. Thus, before explaining the assumption, we restate the definition of a symmetric channel.

Definition 1. [18] *Let $P(W|U)$ be a channel with q-ary input U and arbitrary output W. Denote $\mathcal{I}[a]$ the $(q-1)\times(q-1)$ diagonal matrix with $\mathcal{I}[a]_{i,i} = r^{i\otimes a}$, $i = 1,\dots,(q-1)$. The channel $P(W|U)$ is*

said to be q-ary input symmetric-output *if the possible values of* W *can be relabeled into length* $(q-1)$ *complex-valued vectors* $\tilde{\mathbf{W}}$ *such that*

$$\forall a \in \{0 \dots (q-1)\}, P(\tilde{\mathbf{W}} = \tilde{\mathbf{w}}|U = a) = P(\tilde{\mathbf{W}} = \mathcal{I}[a]\tilde{\mathbf{w}}|U = 0) \ . \tag{18}$$

As this definition is not intuitive, we also derive the following equivalent definition when both U and W take their values in GF(q).

Proposition 1. *Let* U *and* W *be two random variables taking their values in GF(*q*). Then the channel* $P(W|U)$ *is symmetric if and only if there exists a bijective function* $h : GF(q) \to GF(q)$ *such that*

$$P(W = w|U = u) = P(W = h^{-1}(h(w) \oplus u)|U = 0). \tag{19}$$

For the proof, see Appendix A.

As a consequence, from (19), at most q parameters are needed to describe the channel. These parameters correspond to the transition probabilities for the input $U = 0$. Then, the transition probabilities for any other input $U = i$ are simply the permuted transition probabilities for $U = 0$. The permutation is defined by the function h. In channel coding, [18, Proposition 2] shows that for symmetric channels, the error probability of the decoding algorithm is independent of the transmitted codeword. Consequently, the recursion on the probability density is calculated assuming the all-zero codeword was transmitted. In SW coding, this result applies only if X is distributed uniformly and $P(Y|X)$ is symmetric In this case, density evolution for channel coding can be performed directly with $P(Y|X)$. For any other source distribution, as originally proposed by [4] for the binary case, we show that for every joint distribution $P(X, Y)$, there exists an equivalent $P(U, W)$ such that U is uniformly distributed and $P(W|U)$ is symmetric. Equivalent means the two joint distributions induce the same density evolution equations and have the same conditional entropy.

In the following, we first express recursions on the probability densities of the messages in the case of channel coding for symmetric channels. Then, we express the recursion for SW coding, for any channel. At the end, remarking similarities between the two recursions, we derive the equivalence.

A. Density evolution in channel coding for symmetric channels

In the case of a symmetric channel, the probability densities of the messages exchanged in the graph do not depend on the transmitted codeword [18]. Consequently, we assume that the all-zero codeword was transmitted and express the density evolution with this assumption. First, denote $\tilde{P}^{(\ell)}$ the probability density of the *a posteriori* messages (14) at iteration ℓ under the all-zero codeword assumption. It is shown in [18] that the error probability of the sum-product LDPC decoder at iteration ℓ can be calculated as

$$p_e^{(\ell)} = 1 - \int_{\mathbf{m}\in\mathbb{R}_+^q} \tilde{P}^{(\ell)}(\mathbf{m}) d\mathbf{m} \tag{20}$$

where $\mathbb{R}_-^q$ is the set of length q real-valued vectors with positive components only. It thus suffices to express $\tilde{P}^{(\ell)}$ at each iteration to obtain the error probability. For the purpose of the paper, we need an analytical form of DE for non-binary channel coding. As [18] (and any other paper, to the best of our knowledge) does not provide such an analytical form, we state it in the following proposition.

Proposition 2. *Consider a q-ary input symmetric-output channel $P(W|U)$, a code ensemble $\mathcal{C}(\lambda,\rho)$, and sum-product LDPC decoding for channel coding. Assume that the decoding graph is cycle-free and that the all-zero codeword is transmitted. At iteration ℓ, denote $P^{(\ell)}$ the probability density of the messages from VN to CN, $Q^{(\ell)}$ the probability density of the messages from CN to VN, and $\tilde{P}^{(\ell)}$ the probability density of the* a posteriori *messages. Then*

$$Q^{(\ell)}(\mathbf{m}) = \Gamma_d^{-1}\left(\frac{1}{q-1}\sum_{g=1}^{q}\rho\left(\Gamma_c^g(P^{(\ell-1)})\right)\right)(\mathbf{m}) \tag{21}$$

$$P^{(\ell)}(\mathbf{m}) = P^{(0)} \bigotimes \lambda\left(Q^{(\ell)}\right)(\mathbf{m}) \tag{22}$$

where Γ_d^{-1} and Γ_c^h are density transform operators defined in Appendix B. Consequently,

$$P^{(\ell)}(\mathbf{m}) = P^{(0)} \bigotimes \lambda\left(\Gamma_d^{-1}\left(\frac{1}{q-1}\sum_{g=1}^{q}\rho\left(\Gamma_c^g\left(P^{(\ell-1)}\right)\right)\right)\right)(\mathbf{m}) \tag{23}$$

$$\tilde{P}^{(\ell)}(\mathbf{m}) = P^{(0)} \bigotimes \tilde{\lambda}\left(\Gamma_d^{-1}\left(\frac{1}{q-1}\sum_{g=1}^{q}\rho\left(\Gamma_c^g\left(P^{(\ell-1)}\right)\right)\right)\right)(\mathbf{m}) \tag{24}$$

where $\tilde{\lambda}(x) = \sum_{k\geq 2} \lambda_k x^k$.

Proof. The channel coding version of the message computation from VN to CN is given by (13). Consequently, (21) is obtained directly from (13) (sum of *i.i.d.* random variables of probability distribution $P^{(\ell-1)}$ and marginalization according to the VN degree distribution). The channel version of the message computation from CN to VN is given removing $\mathcal{A}[\overline{s}_m]$ in (17). Denote $\bar{G}$ a random variables taking its values in GF(q). For any message $\mathbf{m}$, the density $\Gamma_W^{\bar{G}}$ of $W[\bar{G}]\mathbf{m}$ can be obtained by marginalizing with respect to $\bar{G}$. From the density transform operator obtained in Appendix B1, it is

$$\Gamma_W^{\bar{G}}(\mathbf{m}) = \frac{1}{q-1} \sum_{\bar{g}=1}^{q-1} \Gamma_W^{\bar{g}}(P^{(\ell-1)})(\mathbf{m}). \tag{25}$$

Furthermore, denote $\Gamma_{\mathbf{m}}$, $\Gamma_{\mathcal{F}}$, $\Gamma_{\boldsymbol{\gamma}}$ the density transform operators obtained respectively for the transform of $\mathbf{m}$ into $\mathbf{p}$ (see Appendix B2), for the Fourier Transform (Appendix B3), and for $\boldsymbol{\gamma}$ (Appendix B4) and denote $\Gamma_c^{\bar{g}} = \Gamma_{\boldsymbol{\gamma}}\Gamma_{\mathcal{F}}\Gamma_{\mathbf{m}}\Gamma_W^{\bar{g}}$. The density $\Gamma_\gamma^{\bar{G}}$ of $\boldsymbol{\gamma}\left(\tilde{\mathcal{F}}\left(W\left[\overline{G}\right]\mathbf{m}\right)\right)$ is given by

$$\Gamma_\gamma^{\bar{G}}(\mathbf{m}) = \frac{1}{q-1} \sum_{\bar{g}=1}^{q-1} \Gamma_c^{\bar{g}}(P^{(\ell-1)})(\mathbf{m}) \tag{26}$$

by the linearity of the density transform operators. To finish, from the density transform operators $\Gamma_{\mathbf{p}}$, $\Gamma_{\mathcal{F}^{-1}}$, $\Gamma_{\boldsymbol{\gamma}^{-1}}$ obtained respectively for the transformation of $\mathbf{p}$ into $\mathbf{m}$ (see Appendix B2), for the inverse Fourier Transform (see Appendix B3), and for $\boldsymbol{\gamma}^{-1}$ (see Appendix B4), we get (22) where $\Gamma_d^{-1} = \Gamma_{\mathbf{p}}\Gamma_{\boldsymbol{\gamma}^{-1}}\Gamma_{\mathcal{F}^{-1}}$. Finally combining (21) and (22) gives (23). To finish, (24) directly derives from (23). □

The initial $P^{(0)}$ is obtained by evaluating the probability density of (11) conditioned on the fact that $U = 0$. Note that (23) is not convenient for practical density evolution (see the expressions of the operators in Appendix B). The objective here is only to express a recursion in order to show that a similar form is obtained in SW coding.

B. Density evolution in SW coding

In SW coding, the all-zero codeword transmission cannot be assumed anymore, even if the correlation channel $P(Y|X)$ is itself symmetric, because of the source distribution. Denote respectively $P_k^{(\ell)}$, $Q_k^{(\ell)}$,

and $\tilde{P}_k^{(\ell)}$ the probability densities of the messages from VN to CN, from CN to VN, and of the *a posteriori* messages conditioned on the fact that $X = k$. Note that $P_k^{(\ell)}$, $Q_k^{(\ell)}$, and $\tilde{P}_k^{(\ell)}$ are probability densities conditioned on the fact that $X = k$ but *do not* correspond to an all-k codeword assumption. In fact, *e.g.*, $P_k^{(\ell)}$ can be expressed by marginalizing according to the node neighbor values and thus depend on all the $P_j^{(\ell-1)}$, $j = 0, \ldots, (q-1)$. The following proposition gives the expression of the error probability of the sum-product LDPC decoder in case of SW coding.

Proposition 3. *Consider a joint distribution $P(X, Y)$, where X and Y take their values in GF(q) and $\mathcal{Y}$ respectively, a code ensemble $\mathcal{C}(\lambda, \rho)$, and sum-product LDPC decoding. Let $\tilde{P}_k^{(\ell)}$ be the probability density of the* a posteriori *messages conditioned on the fact that $X = k$ and define*

$$\langle \tilde{P}^{(\ell)} \rangle(\mathbf{m}) = \sum_{k=0}^{q-1} P(X = k) \tilde{P}_k^{(\ell)} \circ \mathcal{A}[\ominus k](\mathbf{m}) \tag{27}$$

Then, in SW coding, the error probability of the LDPC decoder at iteration ℓ is given by

$$p_e^{(\ell)} = 1 - \int_{\mathbf{m} \in \mathbb{R}_+^q} \langle \tilde{P}^{(\ell)} \rangle(\mathbf{m}) d\mathbf{m}. \tag{28}$$

See Appendix C1 for the proof.

Proposition 3 can be interpreted as follows. For a randomly selected variable node of the bipartite graph (see Section IV), $p_e^{(\ell)}$, the probability of error at iteration ℓ, is the probability for an *a posteriori* message to produce a false estimate of the symbol value at the variable node. For example, in the binary case, if $X = 0$ but the scalar message $m^{(\ell)} < 0$, a false estimate of X is produced. Consequently, in the non-binary case, the error probability can be obtained by marginalizing according to $k = 0, \ldots, (q-1)$ and, for each k, by integrating $\tilde{P}_k^{(\ell)}$ over the set of messages producing an error. For $X = k$, this corresponds to the set of messages $\mathbf{m}$ such that there exists $i \neq k$ such that $m_i < m_k$. The marginalization operation appears in (27). Moreover, the operators $\mathcal{A}[\ominus k]$ realize the projection of the space $\mathbb{R}_-^q$ on the set of messages producing an error, thus giving (28).

The following proposition gives the expression of $\langle \tilde{P}^{(\ell)} \rangle$ obtained in SW coding.

Proposition 4. *Consider a joint distribution $P(X,Y)$, where X and Y takes their values in GF(q) and $\mathcal{Y}$ respectively, a code ensemble $\mathcal{C}(\lambda,\rho)$, and sum-product LDPC decoding. Assume that the decoding graph is cycle-free. Denote $P_k^{(\ell)}$ and $\tilde{P}_k^{(\ell)}$ the respective probability densities of the messages from VN to CN and of the* a posteriori *messages at iteration ℓ conditioned on the fact that $X=k$. Denote also $\langle P^{(\ell)}\rangle(\mathbf{m}) = \sum_{k=0}^{q-1} P(X=k) P_k^{(\ell)} \circ \mathcal{A}[\ominus k](\mathbf{m})$ and $\langle \tilde{P}^{(\ell)}\rangle(\mathbf{m}) = \sum_{k=0}^{q-1} P(X=k) \tilde{P}_k^{(\ell)} \circ \mathcal{A}[\ominus k](\mathbf{m})$. In SW coding, the following expressions holds*

$$\langle P^{(\ell)}\rangle(\mathbf{m}) = \langle P^{(0)}\rangle \bigotimes \lambda\left(\Gamma_d^{-1}\left(\frac{1}{q-1}\sum_{g=1}^{q}\rho\left(\Gamma_c^g\left(\langle P^{(\ell-1)}\rangle\right)\right)\right)\right)(\mathbf{m}) \tag{29}$$

$$\langle \tilde{P}^{(\ell)}\rangle(\mathbf{m}) = \langle P^{(0)}\rangle \bigotimes \tilde{\lambda}\left(\Gamma_d^{-1}\left(\frac{1}{q-1}\sum_{g=1}^{q}\rho\left(\Gamma_c^g\left(\langle P^{(\ell-1)}\rangle\right)\right)\right)\right)(\mathbf{m}) \tag{30}$$

where Γ_d^{-1} and Γ_c^g are density transform operators defined in Appendix B and $\tilde{\lambda}(x) = \sum_{k\geq 2}\lambda_k x^k$.

See Appendix C for the proof. The initial density is given by

$$\langle P^{(0)}\rangle = \sum_{k=0}^{q-1} P(X=k) P_k^{(0)} \circ \mathcal{A}[\ominus k](\mathbf{m}) \tag{31}$$

where $P_k^{(0)}$ is calculated $\forall k = 0,\dots q-1$ from the expression of the initial messages (11).

We see that the recursion in SW coding is exactly that obtained in channel coding, except that it now applies on $\langle P^{(\ell)}\rangle$. Consequently, the only difference is on the initial $\langle P^{(0)}\rangle$ which, as expected, takes into account the probability distribution of X. Consequently, we see that if two joint probability distributions $P(X,Y)$ and $P(U,W)$ have the same initial probability densities respectively $\langle P^{(0)}\rangle$ and $P^{(0)}$, *i.e.*, $\langle P^{(0)}\rangle = P^{(0)}$, then they have the same density evolution equations. The following source equivalence is derived from this remark.

C. The source equivalence

From Proposition 4, we would like to identify a source distribution $P(U,W)$ with initial probability density $P^{(0)}$ such that $P^{(0)} = \langle P^{(0)}\rangle$. In the following theorem, we give the expression of the equivalent channel and show that it is symmetric.

Theorem 1. *Consider a joint distribution $P(X,Y)$, where X and Y take their values in GF(q) and $\mathcal{Y}$ respectively, a code ensemble $\mathcal{C}(\lambda,\rho)$, and sum-product LDPC decoding. Denote $\langle P^{(0)} \rangle$ the density of the initial messages. Then there exists an equivalent joint distribution $P(U,W)$ such that U is distributed uniformly, $P(W|U)$ is symmetric, and the initial density under density evolution is given by $P^{(0)} = \langle P^{(0)} \rangle$. Let $W = (W_1, W_2)$ where W_1 takes its values in GF(q) and W_2 takes its values in $\mathcal{Y}$. An equivalent source distribution is such that*

$$P(W_1 = i, W_2 = j | U = k) = P(X = k \oplus i, Y = j). \tag{32}$$

Furthermore, $H(X|Y) = H(U|W)$.

See appendix D for the proof. Consequently, in order to perform density evolution for $P(X,Y)$, it suffices to identify an equivalent $P(U,W)$ from (32). Then one can perform density evolution on this equivalent channel by applying the all-zero codeword assumption.

VI. Examples

In this section, we consider two particular correlation channels $P(Y|X)$ and various input probability distributions $P(X)$. One of the considered correlation channels is symmetric, while the other is not. For each of the considered source models, we perform code degree distribution optimization based on density evolution for the equivalent channel, using a differential evolution algorithm [30].

A. Symmetric correlation channel

Consider a source X taking its values in GF(q) and such that $P(X = x) = p_x$. Here, the correlation channel between X and Y is described by a q-ary symmetric channel in GF(q) with

$$\begin{aligned} P(Y = x | X = x) &= 1 - p \\ \forall y \neq x,\ P(Y = y | X = x) &= \frac{p}{q-1} \end{aligned} \tag{33}$$

where $0 < p < 1$. For given source parameters p_x and p, density evolution gives the error probability $P_e^{(\ell)}(\lambda, \rho)$ of an LDPC code of degree distributions $(\lambda(x), \rho(x))$.

Here, density evolution is computed on the equivalent channel obtained from Theorem 1. However, despite the all-zero codeword assumption simplification, no convenient closed-form expression of the density evolution is known for this model. Thus, here, an approximate $P_e^{(\ell)}(\lambda, \rho)$ will be obtained from an MCMC-based density evolution method called MC-DE [14]. From this, and assuming that the distribution of X is fixed, we get an approximate *threshold* of the code, that is the largest parameter p for which $P_e^{(\ell)}(\lambda, \rho)$ goes to 0 when ℓ goes to infinity.

Now, we want to fix the rate r of the code, and find degree distributions $(\lambda(x), \rho(x))$ of rate r that maximizes the threshold. This optimization can be realized using a genetic algorithm called differential evolution [30]. Here, the code degree optimization will be on the VN degree distribution $\lambda(x)$ only. The CN degree distribution $\rho(x)$ can then be calculated from $\lambda(x)$ and r.

In the following optimization runs, we always perform MC-DE on 1000 samples and 100 iterations. This parameters are shown in [14] to be sufficient to obtain good error probability approximations. For the differential evolution, we consider populations of size 500, with 100 iterations, a crossover probability of 1, and a mutation factor of 0.85 (see [30]). The optimization is then performed for a given maximum VN degree value. For each considered setup and maximum VN degree, the following tables give the best obtained threshold p and the corresponding entropy $H(p) = H(X|Y)$ [1]. The obtained threshold values are also compared to the threshold for a regular code. In the following, $\overline{p}$ denotes the approximate maximum parameter that can be coded with a code of rate r (*i.e.* for which $H(X|Y) \leq r$). The following setups are considered.

a) GF(4), $X \sim [0.25, 0.25, 0.25, 0.25]$, $r = 3/4$, $\overline{p} = 0.355$: In this case, the input probability distribution is symmetric and density evolution can be directly performed on the original channel. The

[1]The optimized degree distributions leading to these threshold values are available online at http://www.elsa-dupraz.fr/documents/degrees.dat

following results are obtained.

Max VN deg.	7	10	15	Reg $(3,4)$
p	0.340	0.346	0.347	0.278
$H(p)$	0.731	0.739	0.740	0.647

b) GF(4), $X \sim [0.5, 0.25, 0.125, 0.125]$, $r = 1/2$, $\overline{p} = 0.225$: Now, the input probability distribution is not uniform anymore, and density evolution is performed on the equivalent channel.

Max VN deg.	7	10	15	Reg $(3,6)$
p	0.214	0.220	0.221	0.175
$H(p)$	0.483	0.492	0.494	0.421

c) GF(16), $X \sim [0.4, 0.04, \ldots, 0.04]$, $r = 1/2$, $\overline{p} = 0.367$: Here, the input probability distribution is not uniform, and we consider a bigger Galois field.

Max VN deg.	10	15	21	Reg $(3,6)$
p	0.321	0.325	0.325	0.294
$H(p)$	0.454	0.458	0.458	0.426

In all cases, increasing the maximum VN degree enables to increase the performance of the code. Moreover, the obtained codes perform much better than the regular code.

B. Non-symmetric correlation channel

We now consider a correlation channel that is no more symmetric. The correlation channel between X and Y is now described by

$$\begin{aligned} P(Y=0|X=1) &= 1-p, & \forall y \neq 0,\ P(Y=y|X=1) &= \frac{p}{q-1} \\ \forall x \neq 1,\ P(Y=x|X=x) &= 1-p, & \forall x \neq 1, \forall y \neq x,\ P(Y=y|X=x) &= \frac{p}{q-1} \end{aligned} \tag{34}$$

where $0 < p < 1$. The equivalent channel is derived from Theorem 1 and the optimization process is the same as before.

d) GF(4), $X \sim [0.25, 0.25, 0.25, 0.25]$, $r = 1/2$, $\overline{p} = 0.114$:

Max VN deg.	7	10	15	Reg $(3, 6)$
p	0.089	0.094	0.097	0.091
$H(p)$	0.456	0.465	0.470	0.460

e) GF(4), $X \sim [0.5, 0.25, 0.125, 0.125]$, $r = 3/4$, $\overline{p} = 0.360$:

Max VN deg.	7	10	15	Reg $(3, 6)$
p	0.306	0.316	0.317	0.257
$H(p)$	0.714	0.721	0.722	0.677

f) GF(16), $X \sim [0.4, 0.04, \ldots, 0.04]$, $r = 1/2$, $\overline{p} = 0.367$:

Max VN deg.	10	15	20	Reg $(3, 6)$
p	0.341	0.345	0.346	0.281
$H(p)$	0.494	0.498	0.499	0.436

We get the same conclusions as for the symmetric case.

Note that, as mentioned earlier, the degree distribution optimization with density evolution can only be seen as a good departure point at the code design process. The finite-length construction can then be performed with an LDPC PEG (Progressive Edge Growth) algorithm [16]. However, at finite length, the decimal coefficients are in fact truncated, which gives different degree distributions with possibly different asymptotic error probabilities. Thus, the degree distribution coefficients have to be adjusted carefully in order to maintain good performance for the code. In addition, once the code is constructed, one has to deal with potentially harmful local structures (mainly short cycles) in order to obtain low error floors [22].

VII. Conclusion

In this paper, we derived an equivalence between SW coding and channel coding under density evolution for non-binary LDPC codes. Thanks to this equivalence, density evolution can be performed for any, even non-symmetric correlation channel and non-uniform source distribution, assuming that the all-zero codeword was transmitted. In addition, an explicit expression of the equivalent channel has been obtained. From this equivalence, we were able to perform code degree optimization for several source models. Future work will be related to the finite-length code design.

Appendix

A. Symmetry

First, from Definition 1, to each value $w \in \text{GF}(q)$, one has to associate a vector $\tilde{\mathbf{w}}(w) \in \mathbb{C}^{q-1}$. Denote $\tilde{\Omega} = \{\tilde{\mathbf{w}}(0), \ldots, \tilde{\mathbf{w}}(q-1)\}$.

From (18),

$$\text{if } \tilde{\mathbf{w}} \in \tilde{\Omega}, \text{ then } \forall u = 0, \ldots q-1, I[u]\tilde{\mathbf{w}} \in \tilde{\Omega}. \tag{35}$$

Consequently, from the expressions of $I[u]$ and r, every non-zero component of $\tilde{\mathbf{w}}$ can take at least κ different values. On the other side, from (18),

$$\forall \tilde{\mathbf{w}} \in \tilde{\Omega}, \{I[u]\tilde{\mathbf{w}}\}_{u=0,\ldots,q-1} = \tilde{\Omega}. \tag{36}$$

Consequently, each non-zero component of $\tilde{\mathbf{w}}$ can take at most κ different values. Thus each non-zero component of $\tilde{\mathbf{w}}$ takes exactly κ different values and any vector $\tilde{\mathbf{w}}$ has exactly α non-zero independent components. We restrict the analysis to these α components of interest and assume without loss of generality that the other components are always equal to 0.

From the previous restriction, we now assume that $\tilde{\mathbf{w}}(w) \in \mathbb{C}^{\alpha}$ and denote

$$\forall k = 1 \ldots \alpha, \tilde{w}_k(w) = a_k(w) \exp(\mathrm{i} b_k(w)) \tag{37}$$

where $a_k(w), b_k(w) \in \mathbb{R}$. From (35) and (36), $a_k(w)$ does not depend on w. Consequently, without loss of generality, we take $\forall w \in \text{GF}(q)$, $\forall k = 1, \ldots, \alpha$, $a_k(w) = 1$. In the same way, we show that the $b_k(w)$ can be decomposed into

$$b_k(w) = c_k + \frac{2\pi}{\kappa} d_k(w) \tag{38}$$

where $c_k \in \mathbb{R}$ and $d_k(w) \in \{0, \ldots, \kappa - 1\}$. As before, without loss of generality, we denote $c_k = 0$, $\forall k = 1, \ldots, \alpha$. Finally, one has $\tilde{w}_k = \exp\left(\mathrm{i}\frac{2\pi}{\kappa} d_k(w)\right)$.

Define

$$\begin{aligned} \mathbf{d} : \text{GF}(q) &\to \{0, \ldots, \kappa - 1\}^\alpha \\ w &\mapsto (d_1(w), \ldots, d_\alpha(w)). \end{aligned} \tag{39}$$

$\mathbf{d}$ is necessarily bijective because every value of GF(q) has to be represented differently. Consequently, there exists a function $\mathbf{d}^{-1} : \{0, \ldots, \kappa - 1\}^\alpha \to \text{GF}(q)$. Then

$$\left(I[u]\tilde{w}(w)\right)_k = r^{\mathrm{i}\otimes u} \exp\left(\mathrm{i}\frac{2\pi}{\kappa} d_k(w)\right) = \exp\left(\mathrm{i}\frac{2\pi}{\kappa}\left(d_k(w) \oplus k \otimes u\right)\right) \tag{40}$$

and from (18),

$$P(W = w | U = u) = P\left(W = \mathbf{d}^{-1}\left(\mathbf{d}(w) \oplus [1, \ldots, \alpha] \otimes u\right) | U = 0\right) \tag{41}$$

in which the operations $\oplus$ and $\otimes$ are componentwise. Further denote $\mathbf{h}(w) = [1, \ldots, \alpha] \otimes \mathbf{d}(w)$ ($\mathbf{h}$ is necessarily bijective). Define an invertible mapping from $\{0, \ldots, \kappa - 1\}^\alpha$ to GF(q) and denote h : GF(q) $\to$ GF(q) the composition of $\mathbf{h}$ and of the invertible mapping. We get

$$P(W = w | U = u) = P(W = h^{-1}(h(w) \oplus u) | U = 0). \tag{42}$$

B. Recursion for channel coding

We look for recursive expressions of $Q^{(\ell)}$ from $P^{(\ell)}$ from (13) and (17). For this, we express the probability density transformations of the operators involved in (17).

1) $\mathcal{W}[g]$ and $R[s]$: In the following, $g \in \mathrm{GF}(q)\backslash\{0\}$ and $s \in \mathrm{GF}(q)$. Let $\mathbf{m}$ be a real-valued vector of size q and $\boldsymbol{\ell} = W[g]\mathbf{m}$. Denote $P_{\mathbf{M}}$ and $P_{\mathbf{L}}$ their respective probability densities and define $\varphi(\boldsymbol{\ell}) = W[g^{-1}]\boldsymbol{\ell}$. The function φ is invertible, and both φ and its inverse φ^{-1} are $\mathcal{C}^1$. The Jacobian matrix of φ is $J_\varphi = W[g^{-1}]$ and $\det(J_\varphi) \neq 0$. Consequently, φ is a $\mathcal{C}^1$-diffeomorphism. By expressing $E[f(\mathbf{L})]$ for any $\mathcal{L}^1$ function f and by variable change we get

$$P_{\mathbf{L}}(\boldsymbol{\ell}) = \det(J_\varphi) P_{\mathbf{M}}(W[g^{-1}]\boldsymbol{\ell}) = \Gamma_W^g(P_{\mathbf{M}})(\boldsymbol{\ell}) \tag{43}$$

where Γ_W^g is the density transform operator.

Using a similar derivative, a density transform operator Γ_R^s can be obtained for $R[s]$.

2) From LLR to probability representation: Define $\mathcal{P}$ as the set of vectors of q components such that $\forall k = 0 \dots q-1$, $0 < p_k < 1$ and $\sum_{k=0}^{q-1} p_k = 1$. Let $\mathbf{m} \in \{0\} \times \mathbb{R}^{q-1}$ and $\mathbf{p} \in \mathcal{P}$ be vectors of size q. The probability densities of $\mathbf{m}$ and $\mathbf{p}$ are denoted respectively $P_{\mathbf{M}}$ and $P_{\mathbf{P}}$. Define the function $\varphi : \{0\} \times \mathbb{R}^{q-1} \to \mathcal{P}$ with $\varphi(\mathbf{m}) = (\varphi_0(\mathbf{m}), \dots, \varphi_{q-1}(\mathbf{m}))$ and $\forall k = 0 \dots q-1$,

$$\varphi_k(\mathbf{m}) = \frac{\exp(-m_k)}{\sum_{k'=0}^{q-1} \exp(-m'_k)} \ . \tag{44}$$

The function φ is invertible with inverse $\varphi^{-1} : \mathcal{P} \to \{0\} \times \mathbb{R}^{q-1}$ with $\varphi^{-1}(\mathbf{p}) = (\phi_0(\mathbf{p}), \dots, \phi_{q-1}(\mathbf{p}))$ and $\forall j = 0 \dots q-1$,

$$\phi_j(\mathbf{p}) = \log \frac{1 - \sum_{j'=1}^{q-1} p'_j}{p_j} \ . \tag{45}$$

Both φ and φ^{-1} are $\mathcal{C}^1$. The Jacobian matrix J_φ of φ is given by

$$\begin{aligned}
(J_\varphi(\mathbf{m}))_{k,k} &= -\exp(-m_k) \left(\sum_{k'=0, k' \neq k}^{q-1} \exp(-m'_k) \right) / \left(\sum_{k=0}^{q-1} \exp(-m'_k) \right)^2 \\
j \neq k : (J_\varphi(\mathbf{m}))_{j,k} &= \exp(-m_k) \exp(-m_j) / \left(\sum_{k=0}^{q-1} \exp(-m'_k) \right)^2
\end{aligned} \tag{46}$$

and $\det(J_\varphi(\mathbf{m})) \neq 0$. Consequently φ is a $\mathcal{C}^1$-diffeomorphism and by variable change in $E\left[f(\mathbf{M})\right]$ for every $\mathcal{L}^1$ function f,

$$P_{\mathbf{M}}(\mathbf{m}) = \det(J_\varphi(\mathbf{m})) P_{\mathbf{P}}(\varphi_1(\mathbf{m}) \dots \varphi_{q-1}(\mathbf{m})) = \Gamma_{\mathbf{m}}(P_{\mathbf{P}})(\mathbf{m}) \tag{47}$$

where $\Gamma_{\mathbf{m}}$ is the density transform operator. On the other hand, the Jacobian matrix $J_{\varphi^{-1}}$ of φ^{-1} is given by

$$\forall j \neq 0 : (J_{\varphi}^{-1}(\mathbf{p}))_{j,j} = -\frac{1}{p_j} - \frac{1}{\sum_{j'=1}^{q-1} p'_j} \tag{48}$$

$$\forall j \neq 0 : (J_{\varphi}^{-1}(\mathbf{p}))_{j,0} = 0 \tag{49}$$

$$\forall j \neq 0 : (J_{\varphi}^{-1}(\mathbf{p}))_{0,j} = -\frac{1}{\sum_{j'=1}^{q-1} p'_j} \tag{50}$$

$$\forall j,k \neq 0 : (J_{\varphi}^{-1}(\mathbf{p}))_{j,k} = -\frac{1}{\sum_{j'=1}^{q-1} p'_j} \tag{51}$$

$$(J_{\varphi}^{-1}(\mathbf{p}))_{0,0} = -\frac{1}{\sum_{j'=1}^{q-1} p'_j} \tag{52}$$

$$\tag{53}$$

Thus $\det(J_{\varphi}^{-1}(\mathbf{p})) \neq 0$ and from the same arguments as before, a density transform operator $\Gamma_{\mathbf{p}}$ can be obtained for the transformation of $\mathbf{m}$ into $\mathbf{p}$.

3) Fourier Transform and inverse Fourier Transform: We consider the Fourier Transform $\mathbf{f} = \mathcal{F}(\mathbf{p})$ of a vector $\mathbf{p}$. As $\mathcal{F}$ is an invertible linear application, by variable change and from the arguments of Appendix B1, we show that

$$P_{\mathbf{F}}(\mathbf{f}) = \det(J_{\mathcal{F}^{-1}}) P_{\mathbf{P}}(\mathcal{F}^{-1}(\mathbf{f})) = \Gamma_{\mathcal{F}}(P_{\mathbf{P}})(\mathbf{f}) \tag{54}$$

where $J_{\mathcal{F}^{-1}}$ is the Jacobian of $\mathcal{F}^{-1}$ and $\Gamma_{\mathcal{F}}$ is the defined density transform operator. A density transform operator $\Gamma_{\mathcal{F}^{-1}}$ can also be obtained from the inverse Fourier transform $\mathbf{p} = \mathcal{F}^{-1}(\mathbf{f})$.

4) γ transform: Define the restricted equivalent function $\tilde{\gamma} : \mathbb{R}^2 \backslash \{0,0\} \to \mathbb{R} \times [-\pi, \pi]$ and

$$\tilde{\gamma}(x,y) = \begin{cases} \left(\frac{1}{2}\log(x^2+y^2), \arctan\frac{y}{x}\right) & \text{if } x \geq 0 \\ \left(\frac{1}{2}\log(x^2+y^2), \arctan\frac{y}{x} + \pi\right) & \text{if } x < 0 \ , \ y \geq 0 \\ \left(\frac{1}{2}\log(x^2+y^2), \arctan\frac{y}{x} - \pi\right) & \text{if } x < 0 \ , \ y < 0 \ . \end{cases} \tag{55}$$

We show that $\tilde{\gamma}$ is $\mathcal{C}^1$ over its interval of definition even in the particular points $(x,0)\ \forall x \neq 0$ and $(0,y)$ $\forall y \neq 0$. Its inverse application is $\tilde{\gamma}^{-1} : \mathbb{R} \times [-\pi, \pi] \to \mathbb{R}^2 \backslash \{0,0\}$ and $\gamma^{-1}(z,t) = (\exp(z)\cos t, \exp(z)\sin t)$.

The determinants of the Jacobian matrices $J_{\tilde{\gamma}}$ of $\tilde{\gamma}$ and $J_{\tilde{\gamma}^{-1}}$ of $\tilde{\gamma}^{-1}$ are given by

$$\det(J_{\tilde{\gamma}}(x,y)) = \frac{1}{x^2+y^2} > 0 \quad , \quad \det(J_{\tilde{\gamma}^{-1}}(z,t)) = \exp(2z) > 0 \ . \tag{56}$$

Consequently, $\tilde{\gamma}$ and $\tilde{\gamma}^{-1}$ are $\mathcal{C}^1-$diffeomorphisms. Denote $P_{X,Y}$ and $P_{Z,T}$ the probability densities associated to random variables (X,Y) and (Z,T). By expressing $E[f(X,Y)]$ and $E[f(Z,T)]$ for every $\mathcal{L}^1$ function f and by variable change, we show that density transform operators can be obtained $\forall (x,y) \in \mathbb{R}^2 \backslash \{0,0\}$ and $\forall (z,t) \in \mathbb{R} \times [-\pi,\pi]$ as

$$\tilde{P}_{X,Y}(x,y) = \Gamma_\gamma(P_{Z,T})(x,y) = \frac{1}{x^2+y^2} P_{Z,T} \circ \tilde{\gamma}(x,y) \tag{57}$$

$$\tilde{P}_{Z,T}(z,t) = \Gamma_{\gamma^{-1}}(P_{X,Y})(z,t) = \exp(z) P_{X,Y} \circ \tilde{\gamma}^{-1}(z,t) \ . \tag{58}$$

The density cannot be obtained in $(0,0)$ by the same method because γ is not continuous in $(0,0)$. However, the probability density functions have to be completed. We get

$$\lim_{z \to -\infty} \tilde{P}_{Z,T}(z,t) = \frac{1}{2\pi} P_{X,Y}(0,0) \tag{59}$$

$$\tilde{P}_{X,Y}(0,0) = \lim_{z \to -\infty} P_{Z,T}(z,t) = \lim_{z \to -\infty} P_Z(z) \tag{60}$$

where $\lim_{z \to -\infty} P_{Z,T}(z,t)$ does not depend on t and P_Z is the marginal density of the random variable Z.

Note that in (15), a transform $\boldsymbol{\gamma}$ involving vectors of size $q-1$ is defined. Its components γ_j, $j = 1 \dots q-1$ apply independently on the components of the input vector (not necessarily composed by independent random variables). Consequently, the transforms defined in (57) can be directly generalized to the vector version.

C. Recursion for Slepian-Wolf coding

1) Expression of the error probability: The error probability $p_e^{(\ell)}$ can be expressed as

$$p_e^{(\ell)} = 1 - \sum_{k=0}^{q-1} P(X=k) \int_{\mathbf{m} \in \Omega_k} \tilde{P}_k^{(\ell)}(\mathbf{m}) d\mathbf{m} \tag{61}$$

where $\Omega_k = \{\mathbf{m} \in \mathbb{R}^q : \forall k' \neq k : m_{k'} > m_k\}$ is the set of messages giving the right value of X. The function $\tilde{\mathbf{m}} \to \mathcal{A}[\ominus k]\tilde{\mathbf{m}}$ is invertible, $\mathcal{C}_1$, and its inverse is also $\mathcal{C}_1$. The Jacobian of the application is $\mathcal{A}[\ominus k]$ and $\det(\mathcal{A}[\ominus k]) \neq 0$. Thus the application is a $\mathcal{C}_1$-diffeomorphism. By change of variable,

$$p_e^{(\ell)} = 1 - \sum_{k=0}^{q-1} P(X=k) \int_{\tilde{\mathbf{m}} \in \mathbb{R}_+^q} \tilde{P}_k^{(\ell)}(\mathcal{A}[\ominus k]\tilde{\mathbf{m}}) d\tilde{\mathbf{m}} \tag{62}$$

To finish (and by replacing $\tilde{\mathbf{m}}$ by $\mathbf{m}$),

$$p_e^{(\ell)} = 1 - \int_{\mathbf{m} \in \mathbb{R}_+^q} \langle \tilde{P}^{(\ell)} \rangle(\mathbf{m}) d\mathbf{m}. \tag{63}$$

2) Multinomial formula: The multinomial formula is restated here because it will be useful for the proof of the recursion. Let $(x_1 \dots x_m)$ be m scalar values. The multinomial formula gives

$$\left(\sum_{k=1}^{m} x_k \right)^n = \sum_{k_1 + \dots + k_m = n} \binom{n}{k_1, \dots, k_m} \prod_{i=1}^{m} x_i^{k_i} \tag{64}$$

where $\binom{n}{k_1,\dots,k_m} = \frac{n!}{k_1! \dots k_m!}$ is the multinomial coefficient. On the other hand, denote $\mathcal{S}_x = \{x_1, \dots, x_m\}$. One can show that the multinomial formula (64) gives also

$$\left(\sum_{k=1}^{m} x_k \right)^n = \sum_{(x'_1 \dots x'_n) \in \mathcal{S}_x^n} \prod_{i=1}^{n} x'_i \ . \tag{65}$$

3) Recursion: For the sake of simplicity, the code is assumed regular with degrees d_v and d_c. The irregular version of the recursion is directly obtained by marginalization according to the degree distributions.

The expression of the density $P_x^{(\ell)}$ is directly obtained from (13) (sum of random variables) as

$$P_x^{(\ell)}(\mathbf{m}) = P_x^{(0)} \bigotimes (Q_x^{(\ell-1)})^{\otimes (d_v - 1)}(\mathbf{m}). \tag{66}$$

On the other hand, $Q_x^{(\ell)}(\mathbf{m})$ can be developed as

$$Q_x^{(\ell)}(\mathbf{m}) = \sum_{\bar{g}_1 \dots \bar{g}_{d_c-1}} \sum_{x_1 \dots x_{d_c-1}} \left(\prod_{i=1}^{d_c-1} \frac{p_{x_i}}{q-1} \right) P(\mathbf{m} | x, x_1 \dots x_{d_c-1}, \bar{g}_1 \dots \bar{g}_{d_c-1}) \tag{67}$$

$$P(\mathbf{m} | x, x_1 \dots x_{d_c-1}, \bar{g}_1 \dots \bar{g}_{d_c-1}) = \Gamma_d^{-1} \left(\bigotimes_{i=1}^{d_c-1} \Gamma_c^{\bar{g}_i}(P_{x_i}^{(\ell-1)}) \right) \circ \mathcal{A}[\ominus \bar{s}](\mathbf{m}) \tag{68}$$

where $\bar{s} = x + \sum_{i=1}^{d_c-1} \bar{g}_i x_i$ and (68) is obtained from (22) completed with $\mathcal{A}$ and from the multinomial formula. Furthermore, $\mathcal{A}[c \oplus b]\mathbf{m} = \mathcal{A}[c]\mathcal{A}[b]\mathbf{m}$ and from (67),

$$Q_a^{(\ell)}(\mathbf{m}) = Q_b^{(\ell)} \circ \mathcal{A}[a \ominus b](\mathbf{m}) \tag{69}$$

Moreover,

$$Q_0^{(\ell)}(\mathbf{m}) = \sum_{\bar{g}_1 \dots \bar{g}_{d_c-1}} \sum_{x_1 \dots x_{d_c-1}} \left(\prod_{i=1}^{d_c-1} \frac{p_{x_i}}{q-1} \right) \Gamma_d^{-1} \left(\bigotimes_{i=1}^{d_c-1} \Gamma_c^{\bar{g}_i} \left(P_{x_i}^{(\ell-1)} \circ \mathcal{A}[\ominus x_i] \right) \right) (\mathbf{m}) \tag{70}$$

$$= \Gamma_d^{-1} \left(\left(\sum_{\bar{g}=1}^{q-1} \sum_{x=0}^{q-1} \frac{p_x}{q-1} \Gamma_c^{\bar{g}} \left(P_x^{(\ell-1)} \circ \mathcal{A}[\ominus x] \right) \right)^{\otimes (d_c-1)} \right) (\mathbf{m}) \tag{71}$$

by the multinomial formula. Finally, by linearity of the density transform operators

$$Q_0^{(\ell)}(\mathbf{m}) = \Gamma_d^{-1} \left(\left(\frac{1}{q-1} \sum_{\bar{g}=1}^{q-1} \Gamma_c^{\bar{g}} \left(\sum_{x=0}^{q-1} p_x P_x^{(\ell-1)} \circ \mathcal{A}[\ominus x] \right) \right)^{\otimes (d_c-1)} \right) (\mathbf{m}) \tag{72}$$

$$= \Gamma_d^{-1} \left(\left(\frac{1}{q-1} \sum_{\bar{g}=1}^{q-1} \Gamma_c^{\bar{g}} \left(\langle P^{(\ell-1)} \rangle \right) \right)^{\otimes (d_c-1)} \right) (\mathbf{m}). \tag{73}$$

Then from (66)

$$\langle P^{(\ell)} \rangle (\mathbf{m}) = \sum_{x=0}^{q-1} p_x \left(P_x^{(0)} \bigotimes \left(Q_x^{(\ell-1)} \right)^{\otimes (d_v-1)} \right) \circ \mathcal{A}[\ominus x](\mathbf{m}) \tag{74}$$

$$= \sum_{x=0}^{q-1} p_x \left(P_x^{(0)} \circ \mathcal{A}[\ominus x] \right) \bigotimes \left(Q_x^{(\ell-1)} \circ \mathcal{A}[\ominus x] \right)^{\otimes (d_v-1)} (\mathbf{m}) \tag{75}$$

by property of the convolution product. Furthermore, from (69),

$$\langle P^{(\ell)} \rangle = \langle P^{(0)} \rangle \bigotimes \left(Q_0^{(\ell-1)} \right)^{\otimes (d_v-1)} (\mathbf{m}). \tag{76}$$

To finish, replacing $Q_0^{(\ell-1)}$ from (73) gives (29) and (30) derives directly from (29).

D. Proof of Theorem 1

Assume that U is distributed uniformly and define

$$\forall k, u \in \mathbf{GF}(q), y \in \mathcal{Y}, P(W_1 = k, W_2 = y | U = u) = P(W = k \oplus u, W_2 = y | U = 0). \tag{77}$$

By setting $(\tilde{\mathbf{w}}(k,y))_j = y\exp\left(2\pi\mathrm{i}\frac{k\otimes j}{\kappa}\right)$ and from Definition 1, we show that the defined channel is symmetric.

For $P(X,Y)$, we show that the initial messages $\mathbf{m}^{(0)}(y)$ are given by

$$\forall j \in \mathrm{GF}(q), y \in \mathcal{Y}, m_j^{(0)}(y) = \log\frac{P(X=0,Y=y)}{P(X=j,Y=y)}. \tag{78}$$

For $P(U,W_1,W_2)$, we show from (77) that the initial messages $\boldsymbol{\eta}^{(0)}(k,y)$ are given by

$$\forall k,j \in \mathrm{GF}(q), y \in \mathcal{Y}, \eta_j^{(0)}(k,y) = \log\frac{P(X=k,Y=y)}{P(X=k\oplus j,Y=y)}. \tag{79}$$

As a consequence,

$$\forall k \in \mathrm{GF}(q), y \in \mathcal{Y}, \boldsymbol{\eta}^{(0)}(k,y) = \mathcal{A}[k]\mathbf{m}^{(0)}(y). \tag{80}$$

For $P(X,Y)$, the initial message distribution is given by

$$\langle P^{(0)}\rangle(\mathbf{m}) = \sum_{k=0}^{q-1} P(X=k)P_k^{(0)} \circ \mathcal{A}[\ominus k](\mathbf{m}). \tag{81}$$

and for $P(U,W_1,W_2)$, the initial message distribution is given by $P^{(0)}(\mathbf{m})$, from the all-zero codeword assumption. The support of $\langle P^{(0)}\rangle$ is thus given by the message vectors $\mathcal{A}[k]\mathbf{m}^{(0)}(y)$, while the one of $P^{(0)}$ is given by the $\boldsymbol{\eta}^{(0)}(k,y)$. Consequently, from (80), the two distributions have the same support.

Furthermore, first assuming that the supports of the $P_k^{(0)}$ are all distinct, one has

$$\langle P^{(0)}\rangle(\mathcal{A}[k]\mathbf{m}^{(0)}(y)) = P(X=k)P_k^{(0)}(\mathbf{m}^{(0)}(y)) = P(X=k)P(Y=y|X=k) \tag{82}$$

and

$$P^{(0)}(\boldsymbol{\eta}^{(0)}(k,y)) = P(W_1=k,W_2=y|U=0) = P(X=k,Y=k). \tag{83}$$

Thus from (82) and (83),

$$\langle P^{(0)}\rangle(\mathcal{A}[k]\mathbf{m}^{(0)}(y)) = P^{(0)}(\boldsymbol{\eta}^{(0)}(k,y)). \tag{84}$$

If the assumption of distinct support is not true, the probability values correspond to a sum in both cases, and (84) still holds.

To conclude, from (80) and (84) and from Propositions 2 and 4, we show that $P(X,Y)$ and $P(U,W_1,W_2)$ have the same equations under density evolution.

It remains to show the entropy equality. First,

$$H(X|Y) = \sum_{y\in\mathcal{Y}} E_{P(X|Y=y)}\left[-\log P(X|Y=y)\right]. \tag{85}$$

Second,

$$H(U|W_1,W_2) = \sum_{w_1\in \mathrm{GF}(q), w_2\in\mathcal{Y}} E_{P(U|W_1=k,W_2=y)}\left[-\log P(U|W_1=k,W_2=y)\right] \tag{86}$$

Moreover,

$$P(U=u|W_1=k,W_2=y) = \frac{P(W_1=k,W_2=y|U=u)P(U=u)}{\sum_{u'\in\mathrm{GF}(q)} P(W_1=k,W_2=y|U=u)P(U=u)} \tag{87}$$

$$= \frac{P(W_1=k\oplus u,W_2=y|U=0)}{\sum_{u'\in\mathrm{GF}(q)} P(W_1=k\oplus u,W_2=y|U=0)} \tag{88}$$

by the fact that U is distributed uniformly and the channel symmetry. Then, from the channel definition,

$$P(U=u|W_1=k,W_2=y) = \frac{P(X=k\oplus u,Y=y)}{P(Y=y)} = P(X=k\oplus u|Y=y). \tag{89}$$

As a consequence, (86) becomes

$$H(U|W_1,W_2) = \sum_{y\in\mathcal{Y}} E_{P(X|Y=y)}\left[-\log P(X|Y=y)\right] = H(X|Y) \tag{90}$$

showing the entropy equality.

REFERENCES

[1] A. Bennatan and D. Burshtein. Design and analysis of nonbinary LDPC codes for arbitrary discrete-memoryless channels. *IEEE Transactions on Information Theory*, 52(2):549–583, 2006.

[2] R.K. Bhattar, K.R. Ramakrishnan, and K.S. Dasgupta. Density Evolution Technique for LDPC Codes in Slepian-Wolf Coding of Nonuniform Sources. *International Journal of Computer Applications IJCA*, 7(8):1–7, 2010.

[3] J. Chen and M. Fossorier. Density evolution for BP-based decoding algorithms of LDPC codes and their quantized versions. In *Global Telecommunications Conference, GLOBECOM*, volume 2, pages 1378–1382. IEEE, 2002.

[4] J. Chen, D.K. He, and A. Jagmohan. The equivalence between Slepian-Wolf coding and channel coding under density evolution. *IEEE Transactions on Communications*, 57(9):2534–2540, 2009.

[5] J. Chou, S. Pradhan, and K. Ramchandran. Turbo and trellis-based constructions for source coding with side information. In *Proc. Data Compression Conference*, pages 33–42. IEEE, 2003.

[6] S.Y. Chung, T.J. Richardson, and R.L. Urbanke. Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation. *IEEE Transactions on Information Theory*, 47(2):657–670, 2001.

[7] T.P. Coleman, A.H. Lee, M. Medard, and M. Effros. On some new approaches to practical slepian-wolf compression inspired by channel coding. In *Data Compression Conference*, pages 282–291. IEEE, 2004.

[8] T.P. Coleman, M. Medard, and M. Effros. Towards practical minimum-entropy universal decoding. In *Data Compression Conference*, pages 33–42. IEEE, 2005.

[9] T.M. Cover and J.A. Thomas. *Elements of information theory, second Edition*. Wiley, 2006.

[10] L. Cui, S. Wang, S. Cheng, and M. Yeary. Adaptive binary Slepian-Wolf decoding using particle based belief propagation. *IEEE Transactions on Communications*, 59(9):2337–2342, 2011.

[11] M.C. Davey and D.J.C. MacKay. Low Density Parity Check codes over GF (q). In *Information Theory Workshop*, pages 70–71. IEEE, 1998.

[12] D. Declercq and M. Fossorier. Decoding algorithms for nonbinary LDPC codes Over GF(q). *IEEE Transactions on Communications*, 55(4):633–643, 2007.

[13] E. Dupraz, A. Roumy, and M. Kieffer. Practical coding scheme for universal source coding with side information at the decoder. *Proceedings of the Data Compression Conference*, pages 401–410, 2013.

[14] M. Gorgoglione, V. Savin, and D. Declercq. Optimized puncturing distributions for irregular non-binary LDPC codes. In *Proc. International Symposium on Information Theory and its Applications (ISITA)*, pages 400–405. IEEE, 2010.

[15] A. Goupil, M. Colas, G. Gelle, and D. Declercq. FFT-based BP decoding of general LDPC codes over Abelian groups. *IEEE Transactions on Communications*, 55(4):644–649, 2007.

[16] X. Hu, E. Eleftheriou, and D. Arnold. Regular and irregular progressive edge-growth tanner graphs. *IEEE Transactions on Information Theory*, 51(1):386–398, 2005.

[17] G. Lechner and C. Weidmann. Optimization of binary LDPC codes for the q-ary symmetric channel with moderate q. In *Proceedings of the International Symposium on Turbo Codes and Related Topics*, pages 221–224, 2008.

[18] G. Li, I.J. Fair, and W.A. Krzymien. Density evolution for nonbinary LDPC codes under Gaussian approximation. *IEEE Transactions on Information Theory*, 55(3):997–1015, 2009.

[19] A. Liveris, Z. Xiong, and C. Georghiades. Compression of binary sources with side information at the decoder using LDPC codes. *IEEE Communications Letters*, 6:440–442, 2002.

[20] F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error-correcting Codes*, volume 16. Elsevier, 1977.

[21] T. Matsuta, T. Uyematsu, and R. Matsumoto. Universal Slepian-Wolf source codes using Low-Density Parity-Check matrices. In *IEEE International Symposium on Information Theory, Proceedings.*, pages 186–190, june 2010.

[22] C. Poulliat, M. Fossorier, and D. Declercq. Design of regular (2, d/sub c/)-LDPC codes over GF (q) using their binary images. *IEEE Transactions on Communications*, 56(10):1626–1635, 2008.

[23] R. Puri and K. Ramchandran. PRISM: A new robust video coding architecture based on distributed compression principles. In *Annual Allerton Conference on Communications Control and Computing, Proceedings.*, volume 40, pages 586–595, 2002.

[24] T.J. Richardson, M.A. Shokrollahi, and R.L. Urbanke. Design of capacity-approaching irregular Low-Density Parity-Check codes. *IEEE Transactions on Information Theory*, 47(2):619–637, 2001.

[25] T.J. Richardson and R.L. Urbanke. The capacity of Low-Density Parity-Check codes under message-passing decoding. *IEEE Transactions on Information Theory*, 47(2):599–618, 2001.

[26] V. Savin. Non binary LDPC codes over the binary erasure channel: density evolution analysis. In *First International Symposium on Applied Sciences on Biomedical and Communication Technologies*, pages 1–5. IEEE, 2008.

[27] D. Slepian and J. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 19(4):471–480, July 1973.

[28] V. Stankovic, A.D.Liveris, Z. Xiong, and C.N. Georghiades. On code design for the Slepian-Wolf problem and lossless multiterminal networks. *IEEE Transactions on Information Theory*, 52(4):1495 –1507, april 2006.

[29] V. Stankovic, A.D. Liveris, Z. Xiong, and C.N. Georghiades. Design of Slepian-Wolf codes by channel code partitioning. In *Data Compression Conference, Proceedings.*, pages 302 – 311, march 2004.

[30] R. Storn and K. Price. Differential evolution– a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[31] C.C. Wang, S.R. Kulkarni, and H.V. Poor. Density evolution for asymmetric memoryless channels. *IEEE Transactions on Information Theory*, 51(12):4216–4236, 2005.

[32] Z. Wang, X. Li, and M. Zhao. Distributed coding of Gaussian correlated sources using non-binary LDPC. In *Congress on Image and Signal Processing*, volume 2, pages 214–218. IEEE, 2008.

[33] Z-L. Wang, X-M Li, and Y. Xu. An Improved Decoding Algorithm for Distributed Video Coding. In *2nd International Congress on Image and Signal Processing, 2009. CISP'09.*, pages 1–4. IEEE, 2009.

[34] N. Wiberg. *Codes and decoding on general graphs*. Citeseer, 1996.

[35] Z. Xiong, A.D. Liveris, and S. Cheng. Distributed source coding for sensor networks. *IEEE Signal Processing Magazine*, 21(5):80–94, Sep 2004.