Binary Dirty Paper Coding

Daniel CASTANHEIRA, Atilio GAMEIRO
Inst. de Telec./Aveiro Univ., Campo Universitário, Aveiro, 3810-193, Portugal
Tel: +351 234377900, Fax: + 351 234 377901
Email: dcastanheira@av.it.pt, amg@ua.pt

Abstract: This paper proposes a practical scheme for implementing binary dirty paper coding (DPC) using a low density generator matrix code (LDGM) concatenated with a high rate low density parity check (LDPC) code. We also propose a new algorithm, a modified version of the belief propagation algorithm (BP), for doing lossy source coding at the encoder, with linear complexity in the block length. In contrast to the superposition coding framework, where high order alphabet codes are used, we propose to implement binary DPC using only binary codes. Through application of approximate density evolution and linear programming we optimize the degree distribution of the proposed code. Simulation results show that our scheme achieves close to state-of-the-art performance with reduced complexity.

1. Introduction

Dirty Paper Coding is a non-linear coding scheme for canceling non-causally known interference at the transmitter. This name has been celebrated in a remarkable paper by Costa [1], in 1983, where it has been shown that the capacity of a Gaussian channel, where the transmitter knowns non-causally the interference, is the same of the corresponding interference free channel. Recently, driven by that results, in [2] the authors have shown that DPC is capacity achieving on the Gaussian broadcast channel. But the applications of DPC do not end here. DPC has found applications in information hiding, data embedding, watermarking and more recently on cooperative communications. The fundamental idea behind DPC is binning. Binning is not only an important concept for DPC. It is also fundamental in multi-user information theory, namely, for example, in cooperative communications. In [3] and [4], Peyman and Wei Yu, have proposed a bilayer LDPC code construction for efficient implementation of binning at the relay channel. The main idea behind their scheme is to design a LDPC code that is capable of working at two different rates: the one at the destination and the one at the relay. In this paper we follow a similar approach to attain the capacity of the binary dirty paper channel. However, instead of designing a dual rate LDPC code for channel coding, we now design a LDGM code, also able to work at two different rates, but where the global code works at the channel coding level and the corresponding sub-code at the lossy source coding level. LDGM codes have emerged as a subset of LDPC codes, but with lower encoding complexity. Although, LDGM codes are asymptotically bad, they exhibit an error floor that is independent of the considered block size. Even though, the poor distance properties of LDGM codes can be easily controlled with proper concatenation of two codes, while maintaining the low complexity advantages [5]. In [6] the authors have shown that, LDGM codes, as duals of LDPC, can saturate the rate-distortion bound of the dual of the binary erasure channel (BEC) coding problem, the binary erasure quantization problem (BEQ), in conjunction with a modified belief propagation (BP) algorithm. Since this pioneering work, LDGM codes have been also extensively used for the lossy source coding problem with very good practical results [7], [8]. As a result

of the good channel and lossy source coding performance of LDGM codes and also due to its inherent low complexity, LDGM codes are good candidates for the DPC problem.

This paper is organized as follows: section 2 presents the general framework of binary DPC. In section 3 we describe our proposed scheme, namely the code structure, respective encoding/decoding algorithms and how to optimize the proposed code, using the erasure channel and the dual code approximations [6]. Next in section 4 the performance of our scheme is accessed by numerical simulations and finally we conclude in section 5.

2.   Binary Dirty Paper Coding Framework

In this section we describe the binary dirty paper framework in more detail. For the binary dirty paper channel, under the binary input $\mathbf{x} \in \{0,1\}^n$ the output takes the form $\mathbf{y} = \mathbf{x} + \mathbf{s} + \mathbf{n}$, where[1] $\mathbf{s} \sim Ber(1/2)$ is the interference signal and $\mathbf{n} \sim Ber(p)$ is the channel noise. The channel input, $\mathbf{x}$, is a function of the interference $\mathbf{s}$ and of the information bearing symbol $\mathbf{d}$, $\mathbf{x} = f(\mathbf{s}, \mathbf{d})$. The objective of binary DPC is to maximize the transmission rate subject to the average input constraint $\mathbf{E}[\|\mathbf{x}\|_1] \leq \delta n$. The capacity of this channel is given by, [9]:

$$R(\delta, p) = \begin{cases} h(\delta) - h(p), & \text{if } \delta_0 \leq \delta \leq 1/2 \\ \alpha\delta, & \text{if } 0 \leq \delta \leq \delta_0 \end{cases} \qquad (1)$$

where $h(.)$ is the binary entropy function, $\delta_0 = 1 - 2^{-h(p)}$ and $\alpha = log((1-\delta_0)/\delta_0)$.

For this problem the binning strategy is to divide the set of all $2^{n(1-h(p))}$ possible[2] input binary sequences into $B = 2^{n(h(\delta)-h(p))}$ bins, indexed by the transmit message $\mathbf{d}$. The input set contains $2^{n(1-h(p))}$ elements draw randomly and thus can be viewed as a channel code with rate $R_C = 1 - h(p)$, for which arbitrary small probability of error can be achieved, for channel crossover probabilities up to p. Each randomly constructed bin contains $2^{n(1-h(p))}/B = 2^{n(1-h(\delta))}$ elements and thus can be viewed as a lossy code with rate $R_L = 1 - h(\delta)$, for which, by the rate-distortion theory [10], an average distortion down to δn can be achieved.

To encode message $\mathbf{d}$ the encoder must restrict itself to bin $\mathbf{d}$ and search for the codeword that is closer to $\mathbf{s}$. After finding the closest codeword to $\mathbf{s}$, $\mathbf{u}$, the encoder inputs to the channel $\mathbf{x} = \mathbf{u} - \mathbf{s}$. At the other end, the decoder receives $\mathbf{u} + \mathbf{n}$ and wants to recover message $\mathbf{d}$. To do that, the decoder can treat the decoding problem as a standard channel decoding problem, to find the sequence $\mathbf{u}$, and after that it only needs to match sequence $\mathbf{u}$ to his respective bin, to obtain $\mathbf{d}$.

3.   Proposed Scheme

In the following sections of this paper we design LDGM codes for the DPC problem and analyze numerically its performance. The proposed code structure is presented in Fig. 1(a). We call the proposed structure LDGM/LDPC code, since it is formed by the concatenation of a LDGM and a LDPC code. The main idea behind the proposed code structure is to use part of the information bits of a LDGM code to approximate the channel interference and to use the other part for real data transmission. However,

---

[1] By $\mathbf{x} \sim Ber(p)$ we mean a Bernoulli random variable with a probability p of being equal to '1'.

[2] By possible we mean a set of codewords for which arbitrary small probability of error, for channel crossover probabilities lower than p, is attainable.

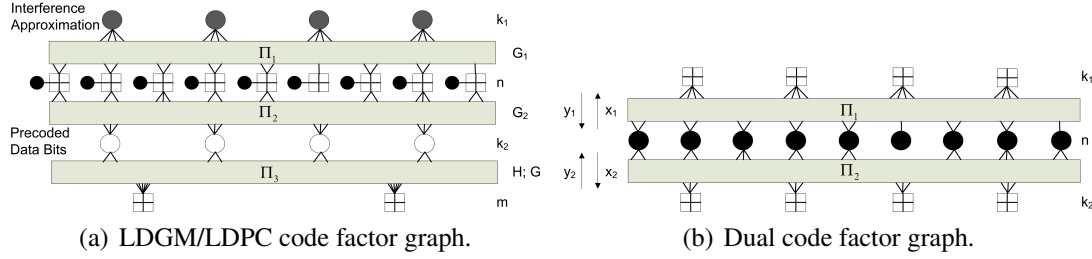(a) LDGM/LDPC code factor graph.   (b) Dual code factor graph.

Figure 1: LDGM/LDPC code and dual code factor graphs.

since a LDGM code is not a good channel code the data bits are firstly precoded with a high rate LDPC code to remove the small distance codewords. In Fig. 1(a) we use white circles to represent the precoded data bits and dark gray circles to denote the information bits used to approximate the channel interference. In addition black circles represent the channel interference and squares represent the check nodes. By $\prod_i, i \in \{1, 2, 3\}$, see Fig. 1(a), we mean a uniformly drawn permutation. It can be seen from Fig. 1(a) and Fig. 1(b) that, without precoding, our proposed code structure is dual of the Bilayer expurgated code, proposed by Peyman Razaghi and Wei Yu in [3], [4], for relay channels. As we will also see, in the following sections, the proposed scheme also fits under the superposition coding framework proposed in [9].

3.1 Modified Belief Propagation Algorithm

Let $\mathcal{N}_j$ denote the set of checks in which bit j participates, $\mathcal{M}_i$ denote the set of variables that participate in check i and $\mathcal{N}_j \backslash i$ denote the set $\mathcal{N}_j$ with node i excluded. Let also c_{ij}^n denote the message sent from check node i to variable node j and v_{ji}^n denote the message sent from variable node j to check node i, at iteration n. Then, the modified BP updating rules can be expressed as follows:

From variable to check:

$$v_{ji}^{n+1}(x_j) \propto \prod_{k \in \mathcal{N}_j \backslash i} c_{kj}^n(x_j) \tag{2}$$

From check to variable:

$$c_{ij}^{n+1}(x_j) = \xi v_{ji}^n(x_j) + \sum_{\sim x_j} f_i(s_i, \mathbf{x}_i) \prod_{k \in \mathcal{M}_i \backslash j} v_{ki}^n(x_k) \tag{3}$$

where s_i represents the channel interference bit i, $0 \leq \xi \leq 1$ and $f_i(s_i, \mathbf{x}_i)$ represents the check node i weight function, that we consider equal to $1.0 - \xi$ or ξ, for variable configurations that satisfy or do not satisfy the check node value, s_i, respectively, when the variables in the set $\mathcal{M}_i$ take the value $\mathbf{x}_i$. For $\xi = 0$ the previously stated equations simplify to the standard BP update equations. The interference is taken into account as an extra message, entering in each check node. The correspondent value, of this message, is considered to be equal to $(1, 0)$ or $(0, 1)$, for $s_i = 0$ and $s_i = 1$, respectively. Interference bit s_i is considered to belong to the set $\mathcal{M}_i$. The term $\xi v_{ji}^n(x_j)$ in equation (3) is used to constrain the message value sent by check i to variable j to the one sent by variable j to himself and, in that way, guide the algorithm through the iterations to a solution. Since, unlike in the channel coding problem, where the received signal

is typically at a short distance of the transmitted codeword, in the lossy source coding problem the interference sequence is likely to be equidistant from a large number of codewords [11], producing local conflicting information about the direction towards which BP should proceed. Thus the BP marginals can get very close to 1/2. Given that there is a need to reinforce them. The function $f_i(s_i, \mathbf{x}_i)$ is used to weight differently the configurations that satisfy or do not satisfy the corresponding check node, limiting in that way the message value sent by check node i. We define the marginal of variable j at iteration $n+1$, like in standard BP, as:

$$m_j^{n+1} \propto \prod_{k \in \mathcal{N}_j} c_{kj}^n(x_n) \tag{4}$$

3.2 Encoding and Decoding

Lets consider that the upper and lower subgraphs have k_1 and k_2 variable nodes, respectively. Denote by $\mathbf{G}, \mathbf{G}_1$ and $\mathbf{G}_2$ the generator matrices of the high rate LDPC code and of the upper and lower subgraphs of the LDGM code, respectively, as shown in Fig. 1(a). Denote by $\mathbf{H}$ the parity check matrix of the LDPC code. Thus the upper, lower and global code can be defined by $\mathcal{C}_0$, $\mathcal{C}_1$, and $\mathcal{C}$, respectively:

$$\mathcal{C}_0 = \{\mathbf{x} \in \{0,1\}^n | \; \mathbf{x} = \mathbf{G}_1 \mathbf{y}, \; \forall \; \mathbf{y} \in \{0,1\}^{k_1}\} \tag{5}$$

$$\mathcal{C}_1 = \{\mathbf{x} \in \{0,1\}^n | \mathbf{x} = \mathbf{G}_2 \mathbf{y}, \forall \; \mathbf{y} \in \{0,1\}^{k_2} : \mathbf{H}\mathbf{y} = 0\} \tag{6}$$

$$\mathcal{C} = \{\mathbf{x} \in \{0,1\}^n | \mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2, \forall \; (\mathbf{x}_1, \mathbf{x}_2) \in \mathcal{C}_1 \times \mathcal{C}_2\} \tag{7}$$

From this definitions one can easily see that the global code ($\mathcal{C}$) is the superposition of two codes ($\mathcal{C}_0$ and $\mathcal{C}_1$), like in [9]. The main difference between our scheme and the one proposed in [9] is that in ours only binary codes are used and instead of a convolutional code a LDGM code is used for doing lossy source coding. Another important difference is that we propose to optimize our code within the linear programming framework.

With the previous definitions, encoding and decoding with the LDGM/LDPC code, for the binary dirty-paper channel, is as follows:

ENCODING: In the encoding stage, a given message $\mathbf{d}$, is first encoded by the lower code $\mathcal{C}_1$ to obtain the lower codeword $\mathbf{c}_1 = \mathbf{G}_2 \mathbf{G} \mathbf{d}$. Then, the proposed modified version of BP is run on the upper subgraph to select the codeword $\mathbf{c}_0$, belonging to $\mathcal{C}_0$, that is closest to $\mathbf{c}_1 - \mathbf{s}$. Finally the encoder inputs to the channel the sequence $\mathbf{c}_0 + \mathbf{c}_1 - \mathbf{s}$.

DECODING: The decoding problem can be cast as a standard channel decoding problem, since the received signal is given by $\mathbf{c}_0 + \mathbf{c}_1 + \mathbf{n} = \mathbf{c} + \mathbf{n}$, where $\mathbf{c}$ is a codeword of the global code $\mathcal{C}$, and $\mathbf{n}$ is the added channel noise. Thus to decode the received data we can simply run the standard BP algorithm ($\xi = 0$, in equation (3)) over the overall factor graph, of code $\mathcal{C}$, to infer the transmitted data bits.

3.3 Code Optimization

In this section we describe how to optimize the degree distribution of the LDGM/LDPC code for the binary dirty paper channel. In Fig. 1(b) we present the dual of the LDGM/LDPC code, if the precoding part is not considered. For a more detailed description of dual codes please refer to [6]. Its not difficult to realize that the dual code is in fact the bilayer expurgated code proposed by Peyman Razaghi and Wei Yu in [3], for relay channels. This resemblance can be used in our favor to optimize the LDGM/LDPC

code. Since to optimize the lossy source coding part of the LDGM/LDPC code, $\mathcal{C}_0$, the dual code approximation [6] can be used and also due to the fact that the optimization of a LDGM code amounts to not much more than translating results obtained for LDPC ensembles [12]. Thus the optimization of the proposed code for the dirty paper channel can be approximated by the optimization of a bilayer expurgated LDPC code, for the relay channel, like in [3], [4]. However, here a further simplification is used: the binary erasure channel approximation. This choice, has been made, mainly to simplify things and in light of the connection between LDPC channel coding and LDGM source coding, in the erasure case [6]. Even if it is an approximation, the resultant performance is very good, as we will see in the next sections, showing, as a proof of concept, that the proposed scheme can be used for DPC.

3.3.1 Density Evolution

As stated in [3] and [4], the ensemble of bilayer expurgated LDPC codes can be characterized by a variable degree distribution $\lambda_{i,j}, i \geq 2, j \geq 0$ and by two (upper and lower) check node degree distributions, $\rho_i^U, i \geq 2$ and $\rho_i^L, i \geq 2$. Nevertheless, we consider regular check degree distributions, with degrees d_c^U and d_c^L, respectively for the upper and lower check nodes. Due to its evidenced good performance as will be seen in the results section and also due to its inherent simplicity.

Using conventional density evolution and the erasure channel approximation the densities of the messages sent by the check nodes, $\mathbf{y} = [y_1, y_2]$, and by the variable nodes, $\mathbf{x} = [x_1, x_2]$, can be expressed by, see Fig 1(b):

$$\mathbf{y} = \left[\begin{array}{c} 1 - (1 - x_1)^{d_c^U - 1} \\ 1 - (1 - x_2)^{d_c^L - 1} \end{array} \right] \qquad \mathbf{x} = \left[\begin{array}{c} \frac{\epsilon}{\eta} \sum_{i,j} i \frac{\lambda_{i,j}}{i+j} y_1^{i-1} y_2^j \\ \frac{\epsilon}{\gamma} \sum_{i,j} j \frac{\lambda_{i,j}}{i+j} y_1^i y_2^{j-1} \end{array} \right] \tag{8}$$

where η corresponds to the percentage of upper edges in the overall graph, $\gamma = 1 - \eta$, and ϵ is the channel erasure probability. Thus, the overall probability of erasure at a given variable node is given by:

$$P_e = \epsilon \sum_{i,j} \lambda_{i,j} y_1^i y_2^j \tag{9}$$

3.3.2 Code Design

To decode a codeword successfully, the overall erasure probability, equation (9), should decrease at each iteration and converge to zero. This can be enforced by constraining y_1 and y_2 to decrease at each iteration and converge to zero, in the large block lengh limit. Due to the one to one correspondence between y_1 and x_1 and between y_2 and x_2, this is equivalent to both x_1 and x_2 decrease, which can be formulated as:

$$\frac{\epsilon}{\eta} \sum_{i,j} i \frac{\lambda_{i,j}}{i+j} y_1^{i-1} y_2^j < x_1 \qquad \qquad \frac{\epsilon}{\gamma} \sum_{i,j} j \frac{\lambda_{i,j}}{i+j} y_1^i y_2^{j-1} < x_2 \tag{10}$$

where y_1 and y_2 should be replaced by their respective expressions, given by equation (8).

The design of a bilayer expurgated LDPC code involves finding $\lambda_{i,j}$, d_c^U, d_c^L and η, such that the overall code and respective upper sub-code are capacity approaching for a channel erasure probability ϵ and $\epsilon_U(< \epsilon)$, respectively. One way to formulate the design problem, like in standard LDPC code optimization, is to fix the check node

degree distribution, (d_c^U, d_c^L), and jointly optimize the parameters $\lambda_{i,j}$ and λ_i^U (upper code variable degree distribution).

Based on iterative linear programming, a rate maximization problem can be formulated, to optimize the bilayer LDPC code as follows. The rate of the overall code is given by $R = 1 - (\sum_i \rho_i/i)/(\sum_i \lambda_i/i)$, where ρ_i and λ_i are the degree distribution of the check nodes and of the variable nodes of the global code, respectively. Fixing the check node degrees, d_c^U and d_c^L, the rate of the global code depends on the $\lambda_{i,j}$ parameters. Thus the rate maximization problem is equivalent to the maximization of $\sum_i \lambda_i/i = \sum_{i,j} \lambda_{i,j}/(i+j)$. As a consequence, the following linear program can be formulated to optimize the proposed LDGM / LDPC code, for η, ϵ, ϵ_U, d_c^U, d_c^L and maximum upper and lower variable degrees known:

$$\max_{\lambda_{i,j},\lambda_i^U} \quad \sum_{i,j} \lambda_{i,j}/(i+j) \tag{11}$$

$$s.t. \quad \sum_{i,j} \lambda_{i,j} = 1; \quad 0 \le \lambda_{i,j}, \; i \ge 2, j \ge 0 \qquad\qquad \sum_{i,j} \frac{i}{i+j}\lambda_{i,j} = \eta \tag{12}$$

$$\sum_j \frac{i}{i+j}\lambda_{i,j} - \eta\lambda_i^U = 0 \qquad\qquad \frac{\epsilon}{\eta}\sum_{i,j} i\frac{\lambda_{i,j}}{i+j}y_1^{i-1}y_2^j < x_1 \tag{13}$$

$$\frac{\epsilon}{\gamma}\sum_{i,j} j\frac{\lambda_{i,j}}{i+j}y_1^i y_2^{j-1} < x_2 \qquad\qquad \epsilon_U\sum_i \lambda_i^U(1-(1-x)^{d_c^U}) < x \tag{14}$$

Equation (12), on the right side, enforces that the percentage of upper edges in the global code should be equal to η, equation (13), left side, expresses the relation between distribution λ_i^U and $\lambda_{i,j}$ and equation (14), on the right side, enforce that the error probability of the optimized upper sub-code should decrease at each iteration and converge to zero.

4. Results

To access the LDGM/LDPC code performance, we have optimized a code for a channel coding rate of $1/2$ (considering the high rate LDPC code) and for a lossy source coding rate of $1/8$, which gives a dirty paper rate of $3/8$. For the aforementioned rates the corresponding channel coding threshold is $p = 0.11$ and the lossy source coding threshold is $\delta = 0.295$, which is equivalent to an input constraint of 0.295. For the optimization process a maximum upper and lower variable degree of 7 and 10 have been considered, respectively, for the dual code. For the upper and lower check nodes a degree of 20 and a degree of 6 has been considered, respectively. The check node degrees have been obtained experimentally by testing several different pairs of values. However for the upper check degree, we can get an idea of the optimal value by looking to LDPC optimized degree distributions, for a code rate of $1 - 1/8 = 7/8$, since the upper sub-code will work "alone" at the encoding stage. Nevertheless its distribution will influence the global code degree distribution. To remove the low weight codewords of the LDGM code, a regular $(3, 60)$ LDPC code has been used. The optimized degree distribution, obtained from the code optimization, is shown in Table 1. As can be seen from that table, a small percentage of degree $(1, 0)$ check nodes has been added to the obtained optimized degree distribution, to help the iterative process to start, likewise for LDGM codes in the channel coding setting. The code used for simulations has been

Table 1: Optimized LDGM/LDPC check node degree distribution, for an upper and lower variable node degree of 20 and 6. An entry (i, j) corresponds to the percentage of edges connected to check nodes with upper degree i and lower degree j.

(i,j)	j = 0	j = 1	j = 2	j = 4	j = 5	j = 6	j = 7	j = 8	j = 9	j = 10
i = 1	0.0012	0	0	0	0	0	0	0	0	0
i = 2	0.0502	0.0179	0.3875	0.0048	0.0477	0.0033	0	0	0	0
i = 3	0.0322	0.0304	0.0452	0.0273	0.0683	0.0421	0.0182	0.0004	0	0
i = 4	0.0190	0	0	0	0.0180	0.0160	0.0080	0	0	0
i = 5	0.0096	0	0	0	0	0.0085	0	0	0	0.0231
i = 6	0	0	0	0	0	0	0	0	0.0007	0.0431
i = 7	0	0	0	0	0	0	0	0	0.0320	0.0453

generated randomly and the codeword length considered was 10^5. The optimal distortion value for a code with rate $1/8$ is 0.295. In our simulations, for 100 trials and for 400 iterations of the modified BP algorithm we have got an average distortion of 0.305 ($\xi = 9.25 \times 10^{-4}$), and for 100 iterations a distortion of 0.308 ($\xi = 0.002$). In Fig. 2 the bit-error rate (BER) performance of the LDGM/LDPC code is shown. These results were obtained for 100 iterations of the BP algorithm over the global code, having as a stop criterion, for each channel crossover probability, 200 received frames in error. One iteration of the global code corresponds to one iteration at the LDGM code and another at the LDPC code. As a reference the capacity limit for this code rate, $p = 0.11$, is also presented. As can be seen from that curve, at $\text{BER} = 2 \times 10^{-5}$ the threshold of our code is $p^* = 0.094$, with a gap to capacity of 0.016, similar to superposition coding proposed in [9]. However our system is only based on binary codes and instead of a convolutional code a LDGM code is used for implementing the lossy source coding part of DPC.

It is worth to note that in the optimization we constrained the check nodes to have regular degree distribution. Further improvement could be achieved allowing irregularity. However, due to its simplicity and evidenced good performance, we have focused only on regular check dual codes, as in standard LDPC code design. Another aspect that should be emphasized here is that instead of a high rate LDPC code a high rate LDGM code could be used to remove the inherent LDGM code error floor [5], reducing further the complexity of the proposed scheme.

5. Conclusion

In this paper we have proposed a practical scheme for implementing binary dirty paper coding with only binary low density codes. The main idea behind the proposed code is to divide the information bits of a LDGM code in two parts and to use one part to transmit data and the other part to approximate the channel interference. However, due to the inherent LDGM error floor, the data bits are firstly precoded by a high rate code, namely a LDPC code. To optimize the LDGM/LDPC code a simple linear programming approach has been proposed. We have also proposed a new algorithm for doing lossy source coding, with linear complexity in the block length. An important element, for code optimization, has been the analogy verified between the bilayer LDPC code approach, for the relay channel, and the proposed LDGM/LDPC code. This analogy has enabled the use of research that exists for bilayer LDPC code optimization,
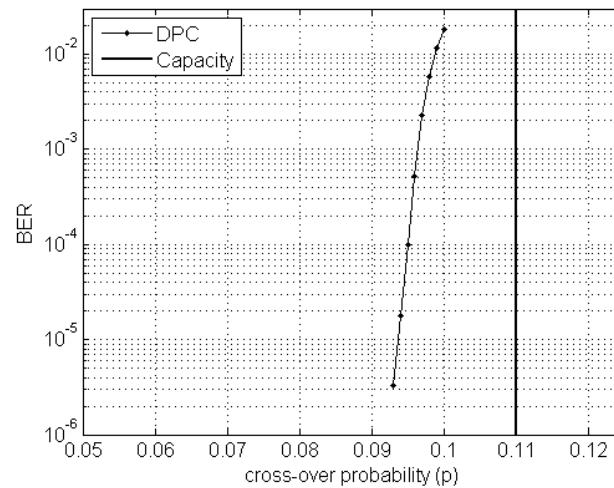
Figure 2: Bit-error rate performance.

simplifying our task significantly. Simulation results indicate reliable transition within 0.016 of the Shannon limit, for a rate of 0.375 and an input constraint of 0.295, showing close to state-of-the-art performance, with reduced complexity.

References

[1] M. Costa, "Writing on dirty paper (corresp.)," *Information Theory, IEEE Transactions on*, vol. 29, pp. 439–441, May 1983.

[2] H. Weingarten, Y. Steinberg, and S. Shamai, "The capacity region of the gaussian multiple-input multiple-output broadcast channel," *Information Theory, IEEE Transactions on*, vol. 52, pp. 3936–3964, Sept. 2006.

[3] P. Razaghi and W. Yu, "Bilayer ldpc codes for the relay channel," in *Communications, 2006. ICC '06. IEEE International Conference on*, vol. 4, pp. 1574–1579, June 2006.

[4] P. Razaghi and W. Yu, "Bilayer low-density parity-check codes for decode-and-forward in relay channels," *Information Theory, IEEE Transactions on*, vol. 53, pp. 3723–3739, Oct. 2007.

[5] J. Garcia-Frias and W. Zhong, "Approaching shannon performance by iterative decoding of linear codes with low-density generator matrix," *Communications Letters, IEEE*, vol. 7, pp. 266–268, June 2003.

[6] E. Martinian and J. S. Yedidia, "Iterative quantization using codes on graphs," *CoRR*, vol. cs.IT/0408008, 2004.

[7] M. J. Wainwright and E. N. Maneva, "Lossy source encoding via message-passing and decimation over generalized codewords of ldgm codes," *CoRR*, vol. abs/cs/0508068, 2005.

[8] T. Filler and J. J. Fridrich, "Binary quantization using belief propagation with decimation over factor graphs of ldgm codes," *CoRR*, vol. abs/0710.0192, 2007.

[9] A. Bennatan, D. Burshtein, G. Caire, and S. Shamai, "Superposition coding for side-information channels," *Information Theory, IEEE Transactions on*, vol. 52, pp. 1872–1889, May 2006.

[10] T. M. Cover and J. A. Thomas, *Elements of information theory*. New York, NY, USA: Wiley-Interscience, 1991.

[11] M. Wainwright, "Sparse graph codes for side information and binning," *Signal Processing Magazine, IEEE*, vol. 24, pp. 47–57, Sept. 2007.

[12] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, Mar. 2008.