Low-Complexity Fixed-to-Fixed Joint Source-Channel Coding

Irina E. Bocharova[1], Albert Guillén i Fàbregas[234], Boris D. Kudryashov[1],
Alfonso Martinez[2], Adrià Tauste Campo[2], and Gonzalo Vazquez-Vilar[2]
[1]St. Petersburg Univ. of Information Technologies [2]Universitat Pompeu Fabra, [3]ICREA, [4]Univ. of Cambridge
Email: irina@eit.lth.se, guillen@ieee.org, boris@eit.lth.se {alfonso.martinez,atauste,gvazquez}@ieee.org

***Abstract*—A source-channel coding scheme in which source messages are assigned to two classes and encoded using a channel code that depends on the class index is studied. A low-complexity implementation with two quasi-cyclic LDPC codes with belief-propagation decoding achieves a better frame error rate than optimized separate coding. The coding gain obtained by simulation is consistent with the theoretical gain.**

I. Introduction

A variety of applications are related to transmission of compressed multimedia data over communication channels. Most of these applications impose strict restrictions on the delivery delay. It is well-known that the performance of joint source-channel coding (JSCC) can improve on the performance of systems based on Shannon's separation principle [2] under finite delay constraints. This potential improvement justifies the interest in practical joint source-channel codes.

Often, multimedia compression standards force the data to be processed in short units of constant length, e. g. frames, packets, slices, group-of-blocks. These units are then encoded with an error-correcting code for their transmission in packets of a fixed size. We refer to these schemes as "fixed-to-fixed" (FF) source-channel codes, since both the source messages and channel codewords have a constant, typically small, length. Numerous FF JSCC schemes have been proposed in the literature. For a detailed overview of previous work, see [3] and references therein. For example, in [3], Fresia el al. propose the use of low-density parity-check (LDPC) block codes in both FF source compression and FF channel coding.

In this paper we propose a novel FF source-channel code based on standard fixed-to-variable (FV) rate lossless compression followed by variable-to-fixed (VF) channel coding. We show that a simple combination of lossless compression and relatively short quasi-cyclic (QC) LDPC codes with belief-propagation (BP)-based decoding outperforms separate LDPC coding. As performance measure we use frame error rate (FER) and, for a fair comparison, both the proposed and separate schemes are optimized with respect to the FER. The decoding complexity of the proposed scheme is less than twice the decoding complexity of a separate source-channel code that uses either of the two sub-codes.

This work has been funded in part by the European Research Council under ERC grant agreement 259663, by the European Union's 7th Framework Programme (PEOPLE-2011-CIG) under grant agreement 303633 and by the Spanish Ministry of Economy and Competitiveness under grants RYC-2011-08150 and TEC2012-38800-C03-03.

The presented coding technique is based on recent results obtained in [1]. This work presented a random-coding analysis of a JSCC scheme where source messages are assigned to classes and encoded by different codes that depend on the class index. At the receiver, the channel output is processed in parallel for each class using a maximum likelihood (ML) decoder. The decoded message is then selected from the outputs of the ML decoders based on a maximum *a posteriori* (MAP) criterion. This joint coding scheme has better error exponent than separate coding [1]. While the implementation proposed in this paper uses QC LDPC codes with BP decoding instead of random codes with ML decoding, the gain over a separate scheme agrees with the theoretically predicted values.

The paper is organized as follows. Section II describes the system model. The proposed coding scheme is presented in Section III. Section IV treats the problem of optimizing the parameters of the QC LDPC codes. Finally, Section V presents simulation results and compares the simulated performance with the corresponding theoretical bounds.

II. System Model

We consider the transmission of a binary memoryless source (BMS) over an additive white Gaussian noise (AWGN) channel. A binary source $X \in \{0, 1\}$ generates a data sequence $\boldsymbol{x} \in \{0, 1\}^n$ of length n. A source-channel code maps the source message $\boldsymbol{x}$ to a binary codeword $\boldsymbol{c} \in \{0, 1\}^N$ of length N. We define the codebook $\mathcal{C}$ as the set of all possible codewords. We denote the source entropy as $H(X)$.

The codeword $\boldsymbol{c} \in \mathcal{C}$ is then transmitted over the AWGN channel using binary phase-shift keying (BPSK) modulation and coherent detection. The binary codeword $\boldsymbol{c} = (c_1, \ldots, c_N) \in \mathcal{C}_i$ is mapped to the transmitted signal $\boldsymbol{y} = (y_1, \ldots, y_N)$ as $y_t = (1 - 2c_t)\sqrt{E_\mathrm{s}}$, $t = 1, 2, \ldots, N$. Here E_s denotes the average symbol energy per channel use. The discrete-time AWGN channel model is given by

$$\boldsymbol{r} = \boldsymbol{y} + \boldsymbol{n} \tag{1}$$

where $\boldsymbol{r} = (r_1, \ldots, r_N)$ denotes the received signal, and the noise vector $\boldsymbol{n} = (n_1, \ldots, n_N)$ is a realization of i. i. d. Gaussian random variables. The variance of the noise is $N_0/2$ such that the ratio E_s/N_0 represents the signal-to-noise ratio per

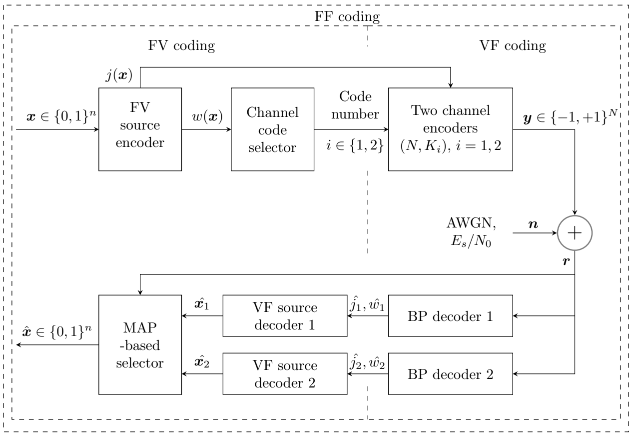

Fig. 1: Implementation of two-class JSCC system

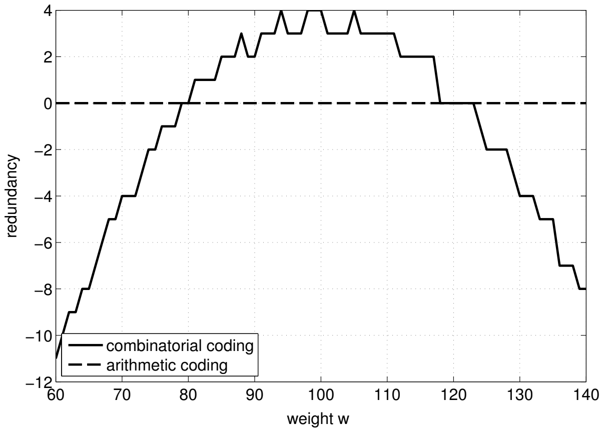

Fig. 2: Redundancy of enumerative coding with respect to arithmetic coding for the BMS with $p = 0.1$ and $n = 1000$.

channel use. We normalize E_s/N_0 with respect to the number of *information bits* transmitted, i. e., the source entropy $H(X)$. We define the signal-to-noise ratio per bit (SNRb) as

$$\frac{E_\mathrm{b}}{N_0} \triangleq \frac{N}{nh(p)} \frac{E_\mathrm{s}}{N_0} \tag{2}$$

where $p \triangleq \Pr\{X = 1\}$ and $h(p) = -p \log_2 p - (1-p) \log_2(1-p)$ denotes the binary entropy function, i. e., $H(X) = h(p)$. Without loss of generality we assume $E_\mathrm{s} = 1$.

III. JSCC Scheme

We consider the scheme shown in Fig. 1. In this scheme the channel code $\mathcal{C}$ is split into two linear (N, K_i)-codes, denoted by $\mathcal{C}_i$, $i = 1, 2$. We define the rate of each of those codes as $R_i = K_i/N$. Since the rates of these sub-codes are different, we may think of $\mathcal{C}$ as a variable-to-fixed channel code. As a source code we may use any fixed-to-variable coding scheme such that the most probable messages are assigned to the shortest codewords. The length of the source codeword $L(\boldsymbol{x})$ determines which one of two available codes will be used to encode each source message. If the source codeword length $L(\boldsymbol{x})$ is below a threshold K_1, then channel code $\mathcal{C}_1$ is used for transmission, otherwise, if $L(\boldsymbol{x}) \le K_2$ the second code, $\mathcal{C}_2$, is used. If $L(\boldsymbol{x}) > K_2$ a source coding error is reported.

A. Encoding

As FV source coding technique we have chosen enumerative encoding [4]. For a given source sequence $\boldsymbol{x} \in \{0,1\}^n$ the binary enumerative encoder first computes a pair of integer numbers (w, j), where $w \in \mathcal{W} = \{0, 1, ..., n\}$ is the Hamming weight of the source sequence, and $j \in \mathcal{J}_w = \{1, 2, ..., \binom{n}{w}\}$ is the index of the actual sequence $\boldsymbol{x}$ in the lexicographically ordered list of all possible sequences of type (weight) w.

Since the mapping $\boldsymbol{x} \to (w, j)$ is one-to-one, the entropy of the random variable (W, J) is $H(W, J) = nH(X)$. Thus, the pair (w, j) can be encoded instead of the actual $\boldsymbol{x}$ without loss of optimality. Moreover, with very small loss of optimality both w and j can be uniformly lossless encoded by codewords of lengths $L_w = \lceil \log_2(n+1) \rceil$ and $L_j = \lceil \log_2 \binom{n}{w} \rceil$, producing a codeword of overall length

$$L_\mathrm{C}(\boldsymbol{x}) = \lceil \log_2(n+1) \rceil + \left\lceil \log_2 \binom{n}{w} \right\rceil . \tag{3}$$

It is well-known that enumerative coding is asymptotically optimal (see, e.g., [5]), that is, the average length of the compressed sequence $\overline{L}_\mathrm{C}$ is such that

$$\frac{\overline{L}_\mathrm{C}}{n} = h(p) + \frac{\log_2 n}{2n} + \mathcal{O}\left(\frac{1}{n}\right). \tag{4}$$

In contrast, arithmetic coding (see, e.g., [6]) uses $-\log_2(1-p)$ and $-\log_2 p$ bits for encoding 0 and 1, respectively. Then, for any sequence $\boldsymbol{x}$ of length n and weight w an arithmetic encoder will require

$$L_\mathrm{A}(\boldsymbol{x}) = \lceil -w \log_2 p - (n-w) \log_2(1-p) \rceil + 1 \tag{5}$$

bits. The average length of the arithmetically encoded sequence is $\overline{L}_\mathrm{A}/n \le h(p) + 2/n$ which is better than that of enumerative coding. However, for enumerative coding the length of the codeword for $\boldsymbol{x}$ is close to $h(w(\boldsymbol{x})/n)$ for all $\boldsymbol{x}$, and if the weight $w(\boldsymbol{x})$ differs from the expected value np, then the codeword produced by the enumerative encoder can be much shorter than that for the arithmetic encoder.

In Fig. 2 we have plotted the redundancy of the enumerative encoder with respect to that of the arithmetic encoder $L_\mathrm{C}(\boldsymbol{x}) - L_\mathrm{A}(\boldsymbol{x})$ as a function of the weight w of the source sequence. In this figure we consider a BMS with $p = 0.1$ and source-sequence length $n = 1000$. Instead of prefix length $\lceil \log_2(n+1) \rceil$ in (3) we used the fixed length 7 that corresponds to restricting the range of encoded weights to 40–167. The probability of the source sequence being out of this range is 1.4×10^{-11}, that is, negligible for practical error probabilities. As follows from this example, when using enumerative coding we spend 3–5 extra bits for typical source sequences, but, on the other hand, we may save several bits for non-typical sequences. Notice also that enumerative coding can be implemented via arithmetic coding with updating symbol probabilities at each step of the encoding procedure.

As linear (N, K_i) block codes $\mathcal{C}_i$, $K_1 \leq K_2$, we use QC LDPC block codes. The corresponding encoding algorithm, based on enumerative encoding and QC LDPC block codes, is presented in Algorithm 1. We choose three weight thresholds: $w_{\min}$, $w_{\max}$, and a threshold w_{thr} that determines which code to use. If a source sequence has Hamming weight either below $w_{\min}$ or above $w_{\max}$, we round up or down its weight to lie in the required range $[w_{\min}, \dots, w_{\max}]$ as shown in Algorithm 1. Then, w_{thr} determines the highest weight for which the first channel code $\mathcal{C}_1$ is used for transmission. Enumerative coding is applied to compute the index j and the length L_j. The source message weight w, given in information bits, is transmitted as a prefix of length that depends on the selected code:

$$L_{\text{p1}} = \lceil \log_2(w_{\text{thr}} - w_{\min} + 1) \rceil \quad (6)$$

$$L_{\text{p2}} = \lceil \log_2(w_{\max} - w_{\text{thr}} + 1) \rceil. \quad (7)$$

Thus, in total $L_j + L_{\text{p}i}$ information bits are to be transmitted and the code dimensions are chosen to satisfy $K_i \geq L_j + L_{\text{p}i}$. If $K_i > L_j + L_{\text{p}i}$ then $l = K_i - L_j - L_{\text{p}i}$ bits are left unused. In our encoding scheme we use l leftover bits for error detection. To do this, we use a pseudo-random binary matrix $\mathcal{H}(w, j)$ of size $(L_j + L_{\text{p}i}) \times l$ and we assume that this matrix is known to the decoder. Multiplying the information vector of length $L_j + L_{\text{p}i}$ by $\mathcal{H}(w, j)$ we obtain a check sequence $\boldsymbol{c}$ of length l which is included into the information sequence of the selected code. The entire information sequence is encoded and transmitted over the communication channel.

B. Decoding

At the decoder two decoding attempts, one for each of the two channel codes, are performed. Instead of optimal ML decoding, we simulate two BP decoders with 50 iterations each. The decoding algorithm is shown in Algorithm 2. For each code we start with 50 iterations of BP decoding. If after 50 iterations the decoder succeeds in obtaining a zero syndrome we then verify that the weight of the decoded sequence is consistent with the code used and with the parity checks written in the leftover positions.

If only one of two codes —the typical case— passed the compatibility tests, then the corresponding data are used as the decoder output. Otherwise, if both decoders fail, we output a predetermined, for example all-zero, data sequence as shown in Algorithm 2. Finally, in case that both decoders report decoding success then we compute the *a posteriori* likelihoods

$$\lambda_j = \frac{2(\boldsymbol{y}_j, \boldsymbol{r})E_{\text{b}}}{N_0} + \ln(P_{w_j}),\ j = 1, 2 \quad (8)$$

where $\boldsymbol{y}_j \triangleq (y_1^{(j)}, \dots, y_N^{(j)})$, $P_w \triangleq p^w(1-p)^{(n-w)}$ is the probability of the source sequence $\boldsymbol{x}$ of weight w. As final decision, the decoder chooses $j = \arg\max\{\lambda_j\}$.

IV. Optimization of QC LDPC Codes for JSCC

An (Mc, Mb) QC LDPC code can be determined by a polynomial parity-check matrix $H(D)$ of its parent rate $R = b/c$ convolutional code

$$H(D) = \{h_{ij}(D)\},\ i = 1, ..., c-b,\ j = 1, ..., c \quad (9)$$

where $h_{ij}(D)$ is either zero or a monomial entry, that is, $h_{ij}(D) = D^{w_{ij}}$ with w_{ij} being a nonnegative integer, $w_{ij} \leq m$, m is syndrome former memory and $M > m$ denotes the tailbiting length.

The corresponding $(c-b) \times c$ base matrix follows as

$$B = H(D)|_{D=1} \quad (10)$$

The polynomial parity-check matrix $H(D)$ (9) can also be represented via its $(c-b) \times c$ *degree matrix*

$$W = \{w_{ij}\},\ i = 1, ..., c-b,\ j = 1, ..., c \quad (11)$$

with entries w_{ij} at the positions of the monomials $D^{w_{ij}}$ and $w_{ij} = -1$ at the zero positions. If each column of $H(D)$ contains J nonzero elements and each row contains K nonzero elements the QC LDPC convolutional code is (J, K)-regular and irregular otherwise.

Techniques of constructing QC LDPC block codes are typically based on choosing suitable base matrices B followed by finding degree matrices W which are good in terms of target criteria. This corresponds to finding proper labelings for the base Tanner graph $\mathcal{G}_{\text{B}}$ determined by B (see, for example, [7], [8] and references therein). In this way, we obtained an infinite Tanner graph determined by $H(D)$ (9) that can be

Input: Source sequence $\boldsymbol{x}$
Compute Hamming weight $w = w(\boldsymbol{x})$
if $w > w_{\max}$ **then**
  Set the final $w - w_{\max}$ ones to zeros
  $w \leftarrow w_{\max}$
end if
if $w < w_{\min}$ **then**
  Set the final $w_{\min} - w$ zeros to ones
  $w \leftarrow w_{\min}$
end if
Enumerative encoding: $\boldsymbol{x} \rightarrow j,\ L = \lceil \log_2 \binom{n}{w} \rceil$ % L is length of j
if $w \leq w_{\text{thr}}$ **then**
  $\mathcal{C} = \mathcal{C}_1$, $K = K_1$, $L_{\text{p}} = L_{\text{p1}}$ %Select (N, K_1) code and prefix length L_{p1}
else
  $\mathcal{C} = \mathcal{C}_2$, $K = K_2$, $L_{\text{p}} = L_{\text{p2}}$ % Select (N, K_2) code and prefix length L_{p2}
end if
$l \leftarrow K - L - L_{\text{p}}$
Select $(L + L_p) \times l$ matrix $\mathcal{H}(w, j)$,
Compute additional parity-check symbols $(w, j) \xrightarrow{\mathcal{H}(w,j)} \boldsymbol{c} = (c_{K-l+1}, ..., c_K)$
Channel encoding: $(w, j, \boldsymbol{c}) \xrightarrow{\text{code } \mathcal{C}} \boldsymbol{y}$
Output: Channel input $\boldsymbol{y} \in \{-1, +1\}^N$

Algorithm 1: JSCC encoding using enumerative source coding and two channel codes.

Input: Channel output sequence $\boldsymbol{r}$
for $i = 1, 2$ **do**
  Set error detection flag $F_i \leftarrow 0$
  BP decoding: $\boldsymbol{y} \xrightarrow{\text{code } \mathcal{C}_i} (\hat{w}, \hat{\jmath}, \hat{\boldsymbol{c}})$
  if $\hat{w}$ is not consistent with the code number **then**
    set $F_i \leftarrow 1$
  else
    Compute the length $\hat{L}$ of the source codeword for $(\hat{w}, \hat{\jmath})$
    $l \leftarrow K_i - \hat{L} - L_{\mathrm{p}i}$
    if $l \leq 0$ **then** $F_i \leftarrow 1$
    else
      Select $(\hat{L} + L_{\mathrm{p}i}) \times l$ matrix $\mathcal{H}(\hat{w}, \hat{\jmath})$,
      Compute l parity checks $(\hat{w}, \hat{\jmath}) \xrightarrow{\mathcal{H}(\hat{w}, \hat{\jmath})} \boldsymbol{c}$
      if $\boldsymbol{c} \neq \hat{\boldsymbol{c}}$ **then** $F_i \leftarrow 1$
      else
        Enumerative decoding $(\hat{w}, \hat{\jmath}) \rightarrow \hat{\boldsymbol{x}}_i$
      end if
    end if
  end if
end for
switch (F_1, F_2)
  case (1,1): $\boldsymbol{x} \leftarrow \boldsymbol{0}$;
  case (0,1): $\boldsymbol{x} \leftarrow \boldsymbol{x}_1$;
  case (1,0): $\boldsymbol{x} \leftarrow \boldsymbol{x}_2$;
  case (0,0):
    for $i = 1, 2$ **do**
      Compute λ_i using (8)
    end for
    if $\lambda_1 > \lambda_2$ **then**
      $\boldsymbol{x} \leftarrow \boldsymbol{x}_1$;
    else
      $\boldsymbol{x} \leftarrow \boldsymbol{x}_2$;
    end if
end switch
Output: Estimated channel input $\boldsymbol{x} \in \{0, 1\}^n$

Algorithm 2: JSCC decoding using two BP decoders and enumerative decoding.

interpreted as the Tanner graph $\mathcal{G}$ of the tailbiting LDPC block code unwrapped and extended to infinity in the time domain. When searching for good QC LDPC codes the girth g [9]–[11], as well as the girth profile [12], the sliding window girth of the code Tanner graph, and the column degree distribution of the base matrix are used as target parameters.

It is well known that for LDPC coding not only the decoding complexity but also the encoding complexity should be taken into account to choose the best code. To facilitate a low-complexity implementation of encoding, we selected codes with bi-diagonal structure of the parity-check part of their parity-check matrices. In other words, all matrices in our experiments had the form $H = \begin{pmatrix} H_{\mathrm{bd}} & H_0 \end{pmatrix}$, where H_{bd} is a binary matrix of size $(c-b) \times (c-b-1)$ with ones only on the main diagonal and straight below it and zeros on all

TABLE I: Code parameters

Rate	M	Girth profile	Degree distribution
$\frac{12}{24}$	42	16(14),21(8),24(6)	11(2), 9(3), 4(11)
$\frac{13}{24}$	42	21(8), 24(6)	10(2), 10(3), 4(10)
$\frac{14}{24}$	42	20(8), 24(6)	9(2), 10(3), 5(8)
$\frac{15}{24}$	42	19(8), 24(6)	8(2), 11(3), 5(8)

other positions, and the matrix H_0 has no restrictions except that one of its columns (let it be the first one) is of weight 3, and contains one nonzero-degree element and two zero-degree elements. For example,

$$H(D) = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & D^7 & 1 & 0 \\ 0 & 1 & 1 & 0 & D^5 & 0 & D & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & D^3 \\ 0 & 0 & 0 & 1 & 1 & D^5 & D & D^{14} \end{pmatrix} \tag{12}$$

is the parity-check matrix of rate $R = 3/8$ $(8M, 3M)$-QC LDPC code with low encoding complexity [13], [14].

We assume that for each SNRb value a pair of rate-optimized QC LDPC codes is used for encoding. For our JSCC scheme for the BMS with $p = 0.1$ we chose QC LDPC codes with $c = 24$ and rates $R = 12/24, 13/24, ..., 16/24$. For constructing these parity-check matrices we used the optimization algorithm from [12]. The only exception is the code of rate $R = 18/24 = 2/3$ which is borrowed from [14], code A. Parameters of the newly constructed QC LDPC codes are presented in Table I where $n_i(g_i)$ stands for the girth g_i of the Tanner graph determined by the submatrix consisting of the first n_i columns of the code parity-check matrix and column degree distribution $n_c(w_c)$ means that there are n_c columns of weight w_c in B.

For each SNRb value we have chosen the best pair of QC LDPC codes $(\mathcal{C}_1, \mathcal{C}_2)$ providing the best FER performance of joint source-channel decoding by optimizing the cost function

$$\begin{aligned} K(\mathcal{C}_1, \mathcal{C}_2) = P(\mathcal{C}_1)\,(P_{\mathrm{ch1}} + P_{\mathrm{mix12}}) \\ + P(\mathcal{C}_2)\,(P_{\mathrm{ch2}} + P_{\mathrm{mix21}}) \\ + \Pr\{w(\boldsymbol{x}) \notin \{w_{\min}, ..., w_{\max}\}\} \end{aligned} \tag{13}$$

where $P(\mathcal{C}_i)$, $i = 1, 2$, denotes the probability of the source sequence to be in class i, i. e. , used with code $\mathcal{C}_i$, $P_{\mathrm{ch}i}$ is the probability of decoding error occurred in the ith BP decoder, and $P_{\mathrm{mix}ij}$ is the probability of an error occurred due the ith encoder being identified instead of the jth encoder.

V. Numerical Results and Discussion

The FER performance of the optimized two-code construction is shown under the name "Joint" in Fig. 3, for $n = 1000$ and $N_1 = N_2 = N = 1008$. For comparison we also include the FER performance of separate coding ("Separate") and that of BP decoding of a QC LDPC code with rate $R = H(X)$ ("Infinite delay"). The separate scheme has been optimized for each SNRb value, as shown in Fig. 4. The error floors correspond to the probability that $L(\boldsymbol{x})$ exceeds the available

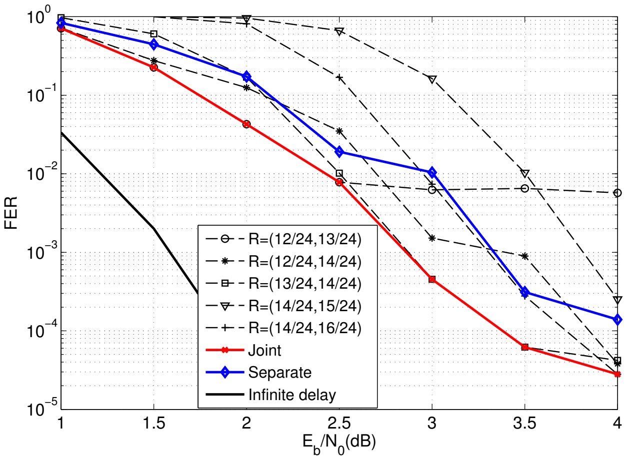

Fig. 3: FER performance for the optimized separate source-channel coding, for pairs of QC LDPC codes in FF source-channel coding scenario, and for the optimized joint source-channel coding using two QC LDPC code, $N = 1008$, $n = 1000$.

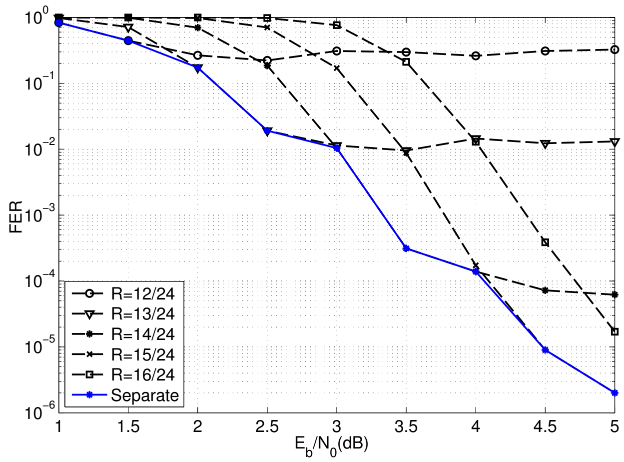

Fig. 4: FER performance for separate QC LDPC codes in FF source-channel coding scenario and for the optimized separate source-channel coding, $N = 1008$, $n = 1000$.

code dimension. The infinite-delay curve can be considered as an empirical lower bound on FER performance achievable by JSCC. From Fig. 3 we can see that the proposed JSCC scheme presents 0.5-0.7 dB gain with respect to separate coding. Also, the SNRb loss of the JSCC scheme with respect to the infinite delay curve is about 1 dB.

Fig. 5 shows the error exponents for JSCC with 2 classes and separate coding [1]. The asymptotic gain of JSCC with two classes with respect to separate coding is about 0.5-1.0 dB. This values are consistent with the simulated FER performance of the suboptimal JSCC with two QC LDPC codes in Fig. 3.

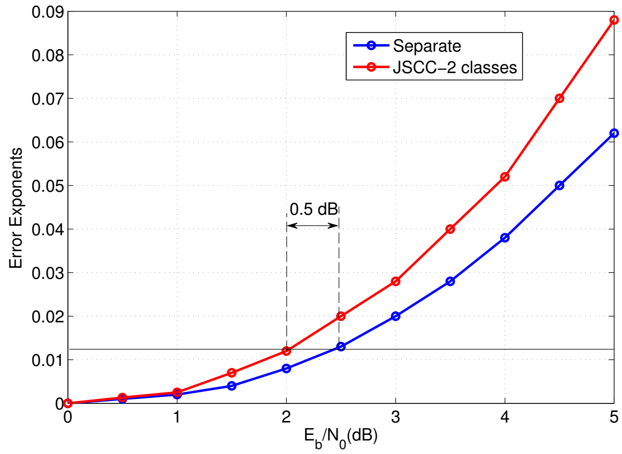

Fig. 5: Error exponent lower bounds [1], $n = N$.

REFERENCES

[1] I. E. Bocharova, A. Guillén i Fàbregas, B. D. Kudryashov, A. Martinez, A. Tauste Campo, and G. Vazquez-Vilar, "Source-channel coding with multiple classes," in *2014 IEEE Int. Symp. on Inf. Theory*, July 2014.

[2] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 1, pp. 379–423, 1948.

[3] M. Fresia, F. Peréz-Cruz, H. V. Poor, and S. Verdú, "Joint source and channel coding," *IEEE Signal Process. Mag.*, vol. 27, no. 6, pp. 104–113, November 2010.

[4] T. Cover, "Enumerative source encoding," *IEEE Trans. Inf. Theory"*, vol. 19, no. 1, pp. 73–77, 1973.

[5] V. F. Babkin, "A universal encoding method with nonexponential work expenditure for a source of independent messages," *Probl. Peredachi Inf.*, vol. 7, no. 4, pp. 13–21, 1971.

[6] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Comm. of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[7] M. Esmaeili and M. Gholami, "Structured quasi-cyclic LDPC codes with girth 18 and column-weight $J \geq 3$," *Int. J. Electron. Comm. (AEÜ)*, vol. 64, no. 3, pp. 202–217, Mar. 2010.

[8] I. Bocharova, R. Johannesson, F. Hug, B. Kudryashov, and R. Satyukov, "Searching for voltage graph-based LDPC tailbiting codes with large girth," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, Apr. 2012.

[9] M. P. Fossorier, "Quasi-cyclic low-density parity-check codes from circulant permutation matrices," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1788–1793, 2004.

[10] M. E. O'Sullivan, "Algebraic construction of sparse matrices with large girth," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 718–727, Feb. 2006.

[11] I. E. Bocharova, F. Hug, R. Johannesson, B. D. Kudryashov, and R. V. Satyukov, "Searching for voltage graph-based LDPC tailbiting codes with large girth," *IEEE Trans. Inf. Theory"*, vol. 58, no. 4, pp. 2265–2279, 2012.

[12] I. Bocharova, B. Kudryashov, and R. Johannesson, "Combinatorial optimization for improving QC LDPC codes performance," in *2013 IEEE Int. Symp. on Inf. Theory*, 2013, pp. 2651–2655.

[13] S. Myung, K. Yang, and J. Kim, "Quasi-cyclic LDPC codes for fast encoding," *IEEE Trans. Inf. Theory"*, vol. 51, no. 8, pp. 2894–2901, 2005.

[14] *Air Interface for Fixed and Mobile Broadband Wireless Access Systems*, IEEE P802.16e/D12 Draft, Oct 2005.