Iterative Bounded Distance Decoding of Product Codes with Scaled Reliability

N.B. When citing this work, cite the original published paper.

(article starts on next page)

Iterative Bounded Distance Decoding of Product Codes with Scaled Reliability

Alireza Sheikh[(1)], Alexandre Graell i Amat[(1)], and Gianluigi Liva[(2)]

[(1)] Dept. of Electrical Engineering, Chalmers University of Technology, Sweden, asheikh@chalmers.se
[(2)] Institute of Communications and Navigation, German Aerospace Center (DLR), Germany

Abstract *We propose a modified iterative bounded distance decoding of product codes. The proposed algorithm is based on exchanging hard messages iteratively and exploiting channel reliabilities to make hard decisions at each iteration. Performance improvements up to 0.26 dB are achieved.*

Introduction

Product-like codes are a family of codes extensively used in fiber-optic communications, for example in the optical submarine standard ITU-T G.975.1. Product codes (PCs) are a specific class of these codes defined as the set of all rectangular arrays such that each row and each column of the array is a codeword of a given component code. PCs are used as stand-alone codes [1,2] or in a concatenation structure (c.f., ITU-T G.975.1, appendix I.5.1) in optical communication systems. As compared to low-density parity-check (LDPC) codes with soft decision decoding, PCs decoded using iterative bounded distance decoding (IBDD),[2] i.e., by iterating BDD decoding of the component codes, yield much lower decoding complexity and power consumption, yet achieving large net coding gains. PCs can be combined with higher order modulations in a so-called coded modulation (CM) scheme, to boost the spectral efficiency[3].

In this paper, we propose a novel decoding algorithm for PCs. The proposed algorithm is based on BDD of the component codes and the iterative exchange of binary (i.e., hard) messages between the component decoders. However, compared to conventional IBDD, the proposed algorithm exploits the channel reliabilities to perform the final hard decision at the output of the component decoders. The proposed algorithm can be seen as a modification of the conventional IBDD and we refer to it as IBDD with scaled reliability (IBDD-SR). We show that the proposed decoding algorithm yields superior performance compared to IBDD with no impact in the decoder data-flow, albeit at the expense of a slight increase in complexity. Gains up to 0.26 dB are observed for a CM system.

Preliminaries

We consider binary PCs with Bose-Chaudhuri-Hocquenghem (BCH) codes as component codes. Let $\mathcal{C}$ be an (n_c, k_c) shortened BCH code obtained from a BCH code constructed over the Galois field $\mathrm{GF}(2^\nu)$ with minimum Hamming distance $\mathsf{d}_{\min}$, error correction capability $t \triangleq \left\lfloor \frac{\mathsf{d}_{\min}-1}{2} \right\rfloor$. The (even) block length n_c, and information block length k_c are given by $n_c = 2^\nu - 1 - s$ and $k_c = 2^\nu - \nu t - 1 - s$, where s is the shortening parameter. The shortened BCH code is defined by the triplet (ν, t, s). A (two-dimensional) PC with parameters (n_c^2, k_c^2) can be defined as the set of all $n_c \times n_c$ matrices $\boldsymbol{C} = [c_{i,j}]$ such that each row and each column of the matrix is a codeword of the (n_c, k_c) component code.

We consider bit interleaved coded modulation (BICM) using quadrature amplitude modulation (QAM) over the additive white Gaussian noise (AWGN) channel. Accordingly, $\boldsymbol{C}$ is randomly interleaved and the rows of the resulting matrix are concatenated, forming a vector $\boldsymbol{c}' = (c'_1, \cdots, c'_{n_c^2})$. The binary reflected Gray mapping is used to map blocks of $m = \log_2(M)$ bits of $\boldsymbol{c}'$ to an M-QAM symbol. At the output of the bit-to-symbol mapper, a sequence of symbols $\boldsymbol{x} = (x_1, x_2, \cdots, x_p)$ of length $p = n_c^2/m$ is obtained. The channel observation at the output of the channel corresponding to the ith QAM symbol, is given by $y_i = x_i + n_i$, where $N_i \sim \mathcal{CN}(0, 2\sigma^2)$. Given y_i, the log-likelihood ratio (LLR) corresponding to the lth bit of the corresponding binary label of x_i is obtained as

$$L_i^l = \ln \left(\sum_{a \in \mathcal{S}_l^1} e^{-\frac{|y_i - a|^2}{2\sigma^2}} \Big/ \sum_{a \in \mathcal{S}_l^0} e^{-\frac{|y_i - a|^2}{2\sigma^2}} \right), \text{ where}$$

$\mathcal{S}_l^1$ and $\mathcal{S}_l^0$ are the sets of M-QAM symbols with a 1 and 0 as the lth bit of the corresponding labeling, respectively. The sequence of LLRs $L_1^1, \cdots, L_1^m, \cdots, L_p^1, \cdots, L_p^m$ is then reshuffled and de-interleaved forming the $n_c \times n_c$ matrix $\boldsymbol{L} = [L_{i,j}]$, where $L_{i,j}$ is the LLR corresponding to code bit $c_{i,j}$. Let $\boldsymbol{R} = [r_{i,j}]$ be the matrix of hard decisions at the output of the channel, i.e., $r_{i,j}$ is obtained taking the sign of $L_{i,j}$ and map-

ping $-1 \mapsto 0$ and $+1 \mapsto 1$. We denote by $\mathsf{B}(\cdot)$ the mapping $L_{i,j} \to r_{i,j}$, i.e., $r_{i,j} = \mathsf{B}(L_{i,j})$.

Bounded Distance Decoding

BDD is a hard decision decoding (HDD) algorithm for linear block codes that corrects all error patterns with Hamming weight up to the error correcting capability of the code, t. BDD can be implemented efficiently for BCH codes with small t. BDD may introduce decoding miscorrections if the error pattern has Hamming weight larger than t and there exists a codeword with Hamming distance from the received hard detected word less than or equal to t. If such a codeword does not exist, a failure is declared. We refer to iterative decoding of PCs based on BDD of the component codes as IBDD, which is accomplished by performing BDD of the row and column codes in an iterative fashion.

Iterative Bounded Distance Decoding of PCs With Scaled Reliability

The proposed IBDD-SR works as follows. Without loss of generality, assume that decoding starts with the decoding of the row codes and, in particular, consider the decoding of the ith row code at iteration ℓ. Let $\mathbf{\Psi}^{\mathsf{c},(\ell-1)} = [\psi_{i,j}^{\mathsf{c},(\ell-1)}]$ be the matrix of hard decisions on code bits $c_{i,j}$ after the decoding of the n_{c} column codes at iteration $\ell-1$. Row decoding is then performed based on $\mathbf{\Psi}^{\mathsf{c},(\ell-1)}$. We assume that the output of the BDD stage of the ith row component code corresponding to code bit $c_{i,j}$, denoted by $\mu_{i,j}^{\mathsf{r}}$, takes values on a ternary alphabet, $\mu_{i,j}^{\mathsf{r}} \in \{\pm 1, 0\}$, where -1 corresponds to the hard-decoded bit 0, $+1$ corresponds to the hard-decoded bit 1, and 0 corresponds to unsuccessful BDD of the ith component row code. Then, the hard decision on $c_{i,j}$ made by the ith row decoder is formed as

$$\psi_{i,j}^{\mathsf{r},(\ell)} = \mathsf{B}(w \cdot \mu_{i,j}^{\mathsf{r}} + L_{i,j}),$$

where ties can be broken with any policy. Parameter w is a scaling parameter that needs to be properly optimized. After decoding of the n_{c} row codes at decoding iteration ℓ, the matrix $\mathbf{\Psi}^{\mathsf{r},(\ell)} = [\psi_{i,j}^{\mathsf{r},(\ell)}]$ is passed to the n_{c} column decoders, and column decoding based on $\mathbf{\Psi}^{\mathsf{r},(\ell)}$ is performed. As before, we assume that the output of the BDD stage of the jth column component code corresponding to code bit $c_{i,j}$, denoted by $\mu_{i,j}^{\mathsf{c}}$, takes values on $\{\pm 1, 0\}$, where $+1$, -1 and 0 have the same meaning as before but for the jth component column decoder. Then, the hard decision on $c_{i,j}$ made by the jth column decoder at iteration ℓ
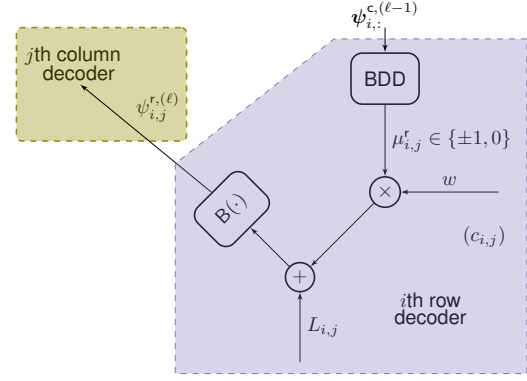

Fig. 1: Block diagram of IBDD-SR for the hard decision on bit $c_{i,j}$ by the ith row decoder at iteration ℓ. $\psi_{i,:}^{\mathsf{c},(\ell-1)}$ is the ith row of $\mathbf{\Psi}^{\mathsf{c},(\ell-1)}$.

is formed as

$$\psi_{i,j}^{\mathsf{c},(\ell)} = \mathsf{B}(w \cdot \mu_{i,j}^{\mathsf{c}} + L_{i,j}).$$

After decoding of the n_{c} column codes at decoding iteration ℓ, the matrix $\mathbf{\Psi}^{\mathsf{c},(\ell)} = [\psi_{i,j}^{\mathsf{c},(\ell)}]$ is passed to the n_{c} row decoders for the next decoding iteration. The schematic diagram of the proposed IBDD-SR is shown in Fig. 1. The crucial modification in IBDD-SR with respect to conventional IBDD is that the hard decisions passed between component decoders are not simply the result of the BDD of the component codes, but are made on the sum of a scaled version of the output of the BDD decoder and the channel LLR. Therefore, the channel reliabilities are exploited to make the final hard decisions at each row and column decoding stage. Intuitively, since the channel reliability is exploited in the hard decisions at each row and column decoding, in the case that the reliability of the channel is high ($|L_{i,j}|$ is large) and conventional IBDD introduces miscorrections, the modified algorithm may combat the possible miscorrections. We highlight that the proposed algorithm exchanges binary (hard) messages, hence the decoder data-flow is the same as that of conventional IBDD. This comes at a minor increase in complexity (a product and a sum for each code bit and iteration) and some memory increase as the channel LLRs need to be stored.

Simulation Results

In order to assess the performance of IBDD-SR, it is required to optimize the scaling parameter w. For a given component code (ν, t, s), we simulate the performance of the PC for $w = 1$ and find the corresponding threshold, $(E_{\mathsf{b}}/N_0)^*$ (with E_{b} being the energy per information bit and N_0 the single-sided noise spectral density), where the bit error rate (BER) is 10^{-4}. We search numerically for the value of w that minimizes $(E_{\mathsf{b}}/N_0)^*$ and use
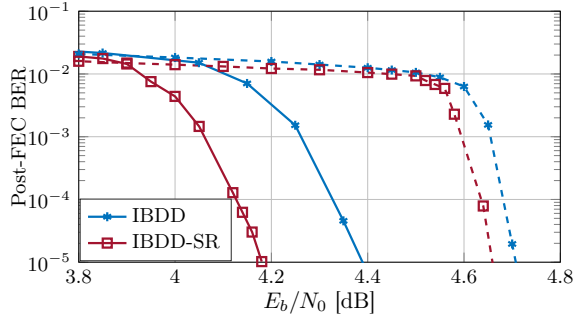
Fig. 2: BER for $\mathcal{C}_1$ (dashed, $w = 1.44$), and $\mathcal{C}_2$ (solid, $w = 1.13$) with BPSK.
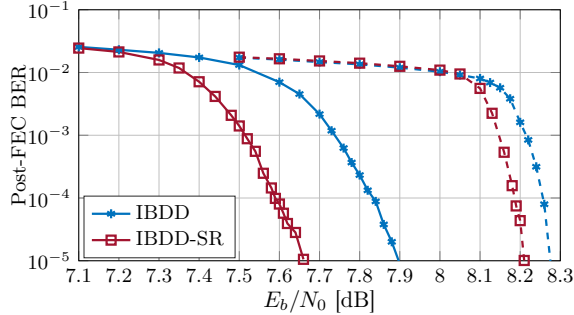
Fig. 3: BER for $\mathcal{C}_1$ (dashed, $w = 12.82$), and $\mathcal{C}_2$ (solid, $w = 8.88$) with 16-QAM.

Fig. 4: BER for $\mathcal{C}_2$ with 64-QAM.

Fig. 5: BER for $\mathcal{C}_2$ with 256-QAM.

it for performance evaluation in the entire range of E_b/N_0. We remark that in our simulations w is fixed across iterations, since by allowing the variation of w over iterations we found a negligible BER improvement. We consider two PCs, $\mathcal{C}_1$ and $\mathcal{C}_2$, with component codes $(9, 4, 7)$[4] and $(8, 3, 63)$[4], respectively, and corresponding code rates 0.8622 and 0.7656 and $t = 4$ and $t = 3$. In Fig. 2 and Fig. 3, the performance of CM for PCs $\mathcal{C}_1$ and $\mathcal{C}_2$ decoded using IBDD-SR and IBDD with 10 iterations for binary phase shift keying (BPSK) and 16-QAM is depicted. As can be seen, the performance gains of IBDD-SR compared to IBDD at BER of 10^{-5} are 0.05 dB and 0.21 dB for BPSK with $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively, while the gains are 0.065 dB and 0.23 dB for 16-QAM. For $t = 3$ IBDD-SR yields a significant gain, while for $t = 4$ the gain is very small. The intuition is that for larger t, the probability of miscorrection of the component code is reduced, hence, the performance gain of IBDD-SR compared to IBDD is reduced. In Fig. 4 and Fig. 5, we plot the performance of $\mathcal{C}_2$ decoded with 10 iterations for 64-QAM and 256-QAM, respectively. The performance gains of IBDD-SR compared to IBDD at a BER of 10^{-5} are 0.25 dB and 0.26 dB for 64-QAM and 256-QAM, respectively. The performance improvement of IBDD-SR over iterative IBDD is slightly higher than for BPSK. Our simulations show similar gains (~ 0.26 dB) for component codes with $t < 3$. Note that low values of t (e.g., $t = 3$) are interesting for optical communications, since a smaller t entails a lower decoding complexity (and higher throughput).
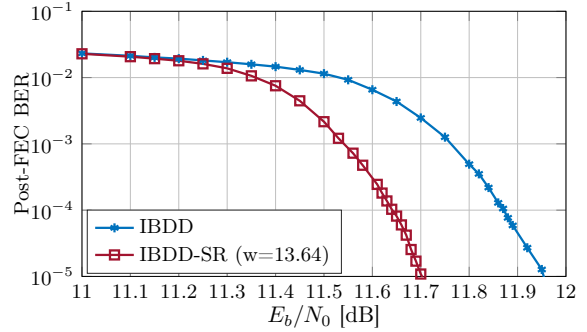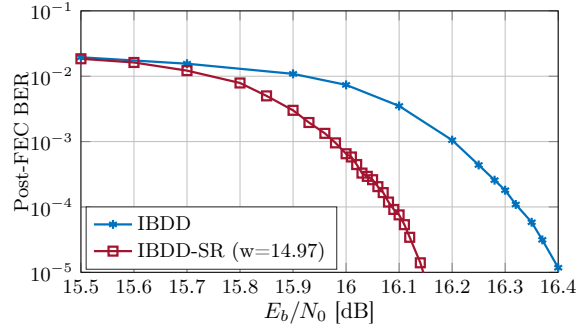
Conclusion

A novel decoding algorithm for product codes, based on BDD of the component codes, is proposed. As conventional IBDD, the proposed algorithm exchanges binary (hard) messages between component decoders. The key novelty with respect to conventional IBDD is that the hard decisions on the code bits at each decoding stage exploit the channel LLRs. The performance gains of the novel IBDD-SR over conventional IBDD are considerable for component codes with $t \leq 3$. Gains up to 0.26 dB are observed compared to IBDD for 256-QAM modulation. Currently, we are extending IBDD-SR to staircase codes.

Acknowledgment

References

[1] B. Li, K. J. Larsen, D. Zibar, and I. T. Monroy, "Over 10 dB net coding gain based on 20communication systems," in Proc. Eur. Conf. Opt. Commun., Geneva, Switzerland, Sep. 2011, pp. 1–3.

[2] J. Justesen, "Performance of product codes and related structures with iterated decoding," IEEE Trans. Commun., vol. 59, no. 2, pp. 407–415, Feb. 2011.

[3] A. Sheikh, A. Graell i Amat, and G. Liva, "Achievable information rates for coded modulation with hard decision decoding for coherent fiber-optic systems," IEEE/OSA J. Lightw. Technol., vol. 35, no. 23, pp. 5069–5078, Dec. 2017.

[4] C. Häger, A. Graell i Amat, H. D. Pfister, A. Alvarado, F. Brännström, and E. Agrell, "On parameter optimization for staircase codes," in Proc. Optical Fiber Commun. Conf., Los Angeles, CA, USA, Mar. 2015, pp. 1–3.