A novel soft-aided bit-marking decoder for product codes

A NOVEL SOFT-AIDED BIT-MARKING DECODER FOR PRODUCT CODES

Gabriele Liga[1]* , Alireza Sheikh[1], and Alex Alvarado[1]

[1]*Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands*
*E-mail: g.liga@tue.nl

Abstract

We introduce a novel soft-aided hard-decision decoder for product codes adopting bit marking via updated reliabilities at each decoding iteration. Gains up to 0.8 dB vs. standard iterative bounded distance decoding and up to 0.3 dB vs. our previously proposed bit-marking decoder are demonstrated.

1 Introduction

As optical line rates are quickly shifting towards 400 Gbit/s and beyond, implementing soft-decision forward-error correction (SD-FEC) decoders becomes increasingly challenging due to their high decoding complexity and internal data-flow. For this reason, lately, considerable attention has been devoted to hard-decision FEC (HD-FEC) schemes whose decoding complexity and data-flow can be order of magnitudes less than SD-FEC.

In the context of HD-FEC, product codes (PCs) [1] (and generalisation thereof, such as staircase codes [2]) offer at least at high coding rates a very good compromise between complexity and performance. This is due to the use of iterative algebraic decoding which is well known to be capacity-approaching at high rates in a binary-symmetric channel. A product code constructed on a (n, k) component code is a set of square arrays of size $n \times n$ where any row and column of each array is a valid codeword in the component code. Typically, Bose-Chaudhuri-Hocquenghem (BCH) or Reed-Solomon codes are used as a component code.

PCs based on BCH codes are typically decoded using iterative bounded distance decoding (iBDD) which can lead to substantial coding gains. However, the main performance limitation of BDD is due to so-called decoding *failures* and *miscorrections*. A decoding failure is declared when the received vector of bits is not within the correction capability of the component code. A miscorrection instead occurs when the received vector is succesfully decoded, but it is mapped into a codeword different from the transmitted one.

A number of new decoding strategies have recently been proposed to tackle both failures and miscorrections, and thus, improve the performance of iBDD decoders whilst preserving an iterative algebraic decoding structure [3–8]. Except from [3], all these strategies somehow require the use of soft information from the channel. For instance in [4, 6], the channel bit reliabilities are updated at each iteration based on the previous BDD output and are used to produce new hard decisions for the next BDD decoding stage.

In our previous work [5, 9], we showed how marking the input bits using the log-likelihood ratios (LLRs) can effectively prevent failures and miscorrections resulting in a significant improvement of the iBDD performance. We referred to this approach as *soft-aided bit-marking* (SABM). SABM was first introduced in [5] for staircase codes, and later also applied to PCs [9].

In this contribution, we extend the SABM algorithm to include scaled reliabilities (SRs) in the decoding process. Differently from [4, 6], the SRs are used to update the bit-marking process at each decoding iteration. The proposed design efficiently merges SABM and SRs to optimise the decoding of PCs. By doing that, we achieve up to 0.8 dB gain compared to standard iBDD and 0.3 dB compared to the previous implementation of SABM.

2 Soft-aided bit marking decoder

The SABM algorithm was shown to be an effective way to mitigate failures and miscorrections in iBDD [9]. The algorithm workflow is illustrated in Fig. 1. In the standard BDD (top part of Fig. 1), the received codeword r is succesfully decoded if there exists a codeword $\hat{c}$ that differs by at most t positions from r, where t is the correction power of the component code. Alternatively, a failure is declared and the decoder simply outputs $c' = r$. A successful BDD decoding does not always guarantee that the transmitted codeword is correctly recovered. If more than t errors are introduced by the channel, and r is within t positions from a codeword $c' \neq c$, the decoder will select c' as opposed to c, resulting in a so called *miscorrection*. Miscorrections often result in the decoder adding more errors than the ones introduced by the channel, thus leading to a deterioration of the iBDD performance.

SABM tackles both failures and miscorrections via a twofold action: *miscorrection detection* and *bit flipping*. This approach is enabled by the use of *soft quantities* or *reliabilities* extracted from the channel. In SABM, LLRs are used as a bit reliability measure. By setting a threshold in the LLR magnitude, bits are marked as highly reliable bits (HRBs) when they fall above
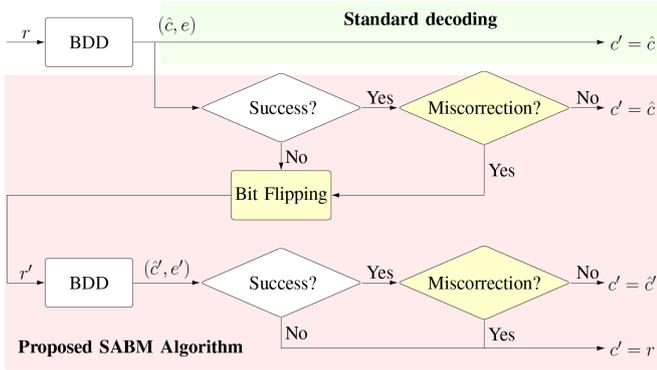
Fig. 1: Workflow of the SABM algorithm presented in [5].

Fig. 2: Schematic diagram of one iteration of the proposed SABM-SR algorithm.

such a threshold. Alternatively, they are labelled as highly unreliable bits. Two rules are adopted to prevent a miscorrection: i) the BDD decoder can never flip a bit from a previously successfully decoded codeword; ii) HRBs can never be flipped. When either a failure occurs or a miscorrection is detected (red area in Fig. 1) a new decoding attempt is performed after *bit flipping*.

Bit flipping is used to facilitate the success of an additional decoding attempt. The LLRs are ordered based on their magnitude and the least reliable bit is flipped in the case of failure. In the case of a detected miscorrection, the least $d_{min} - t - w_H(e)$ bits are flipped, where d_{min} and $w_H(e)$ are the minimum Hamming distance of the component code and the Hamming weight of the error vector e, respectively. If after an additional decoding attempt on the flipped received vector r' a further failure or miscorrection is detected, the output is set to $c' = r$ and passed to the next BDD decoding stage. This process is performed for a few iterations. For PCs the number of these iterations is lower than the number of total iBDD iterations and can be optimised in terms of post-FEC bit-error rate (BER) performance (see Sec. 4).

3 SABM with scaled reliabilities

In SABM the bit marking occurs only once (based on the channel LLRs) and is then used over a certain number of iterations. This approach is, however, sub-optimal as the marking mask loses its validity after bits are updated at every iteration by the BDD decoders. This problem can be solved by re-marking the output bits at each BDD iteration by defining a reliability measure for the outbound BDD decisions. The SR measure proposed in [4, 6] linearly combines the BDD decisions with the channels LLRs. We call this new approach SABM-SR.

The SABM-SR algorithm is illustrated in Fig. 2. The BDD output can provide at each iteration quantised information by assigning values -1 and +1 to successfully decoded bits 0 and 1, respectively. When a failure occurs the BDD output will instead be identically set to 0. The green blocks in Fig. 2 show how the reliabilities are updated at each iteration for both row and column decoding. Let $u_{i,j}^r \in \{-1, 0, +1\}$ be the quantised reliability information provided by the row BDD decoder for the bit in row i and column j. The SR after row decoding at each iteration is defined as [4]
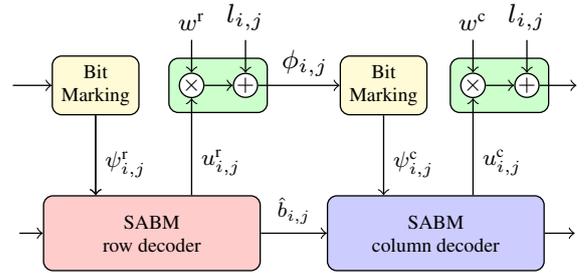
$$\phi_{i,j} = w^r u_{i,j}^r + l_{i,j} \quad \forall \, i, j = 1, 2, ...n \quad (1)$$

where $w^r \in \mathbb{R}^+$ are optimised weights and $l_{i,j}$ are the LLRs.

As illustrated in Fig. 2 (yellow blocks), the SRs $\phi_{i,j}$ can be used to update the SABM marking stage. The bit-marking block takes as an input the SRs from the previous (column) decoding iteration and yields a new set of marked bits $\psi_{i,j}^r$. The new marked bits are then passed to the SABM row decoder which produces a block of output bits $\hat{b}_{i,j}$ and a corresponding block $u_{i,j}^r$. This information is used to update again the reliabilities $\phi_{i,j}$ according to (1). A marking update is then performed via the $\phi_{i,j}$ for the SABM column decoder. Differently from [4, 6], here the reliability information at each iteration is used merely to re-mark the bits at the input of the next BDD decoder, and the BDD output $\hat{b}_{i,j}$ is still used as the input of the next BDD decoder.

4 Results and Discussion

The two decoders described in this paper were numerically assessed in an additive white Gaussian noise channel with binary antipodal modulation (2PAM) and noise power spectral density $N_0/2$. Ten decoding iterations were used for all the implemented decoders. For SABM, and SABM-SR, the reliability threshold and the number of iterations where miscorrection detection (and bit flipping) is performed was optimised and set to 5 and 5 (out of 10) iterations, respectively. The SR weights were numerically optimised and it was found $w^r \approx w^c$. The optimised vector $\boldsymbol{w} \triangleq [w_1, w_2, ..., w_N]$, where w_k is the SR weight at iteration k and N is the number of marking iterations, was found to be $\boldsymbol{w} = [3.42, 3.87, 4.08, 4.27, 4.49]$.

Fig. 3 shows the post-PC bit-error rate (BER) as a function of E_b/N_0, where E_b is the energy per information bit. The component code is in this case a BCH $(7,2,1)^*$, yielding a PC rate $R = 0.78$. Other than the decoding algorithms described in Secs. 2, 3, the performance of an ideal (miscorrection-free) iBDD (dash-dotted curve) and a *fully-fledged* SD decoder such as the turbo product decoder (TPD) based on the Chase-Pyndiah algorithm [10], are also shown as a reference.

---

*We use the common notation (ν, t, e) for BCH codes, where ν is the order of the extended Galois field and e number of bits of extension.

Fig. 3: BER vs. E_b/N_0 for the decoding schemes analysed and for a code rate R=0.78.
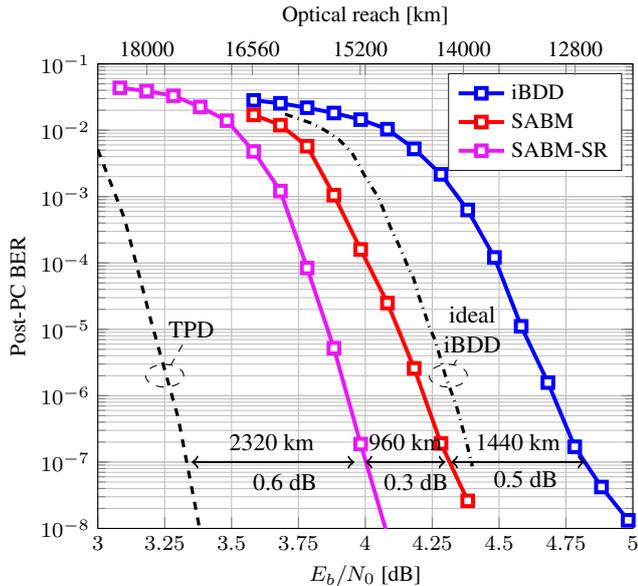

Fig. 4: BER vs. E_b/N_0 for the decoding schemes analysed and for a code rate R=0.87.

The SABM decoder (red curve) shows a 0.5 dB coding gain compared to iBDD (blue line) at a BER of 10^{-7}. SABM also outperforms ideal iBDD (dashed-dotted black line), due to its capability to also mitigate failures. SABM-SR gains 0.3 dB over SABM and an overall 0.8 dB compared to iBDD. This substantial coding gain comes at expense of some additional decoding complexity, due to the SR calculation and the repeated marking operation. However, as noted in [4] and also illustrated in Fig. 2, the SRs are not passed between the SABM decoding stages and can be calculated and stored in place, considerably reducing the decoder data flow. The only messages passed to the decoders are a mask of marked bits and the previous decoder output (hard bits). Finally, TPD (dashed black line) gains 0.6 dB over SABM-SR. Thus, SABM-SR performance is closer to an SD decoder than to conventional HD decoding for PCs (iBDD).

Fig. 3 also shows the performance in terms of achievable optical reach (x axis at the top). In order to link E_b/N_0 to a given transmission distance, a closed-form for the enhanced Gaussian noise model proposed in [11] was used. For our analysis, we assumed a long-haul optical transmission system (parameters in Table 1) using polarisation-multiplexed quadrature phase-shift keying (PM-QPSK) and Erbium-doped fibre amplifiers (EDFAs). SABM yields a 1440 km reach increase, whilst SABM-SR gains an additional 960 km and an overall 2400 km compared to iBDD. Using an SD decoder (TPD) increases the reach by 2320 km compared to SABM-SR.

Fig. 4 shows the performance of a PC using a BCH (8,2,1) with rate $R = 0.87$. It can be observed that the gains are slightly reduced compared to Fig. 3, due to the increased code rate. SABM (red line) gain over iBDD (blue line) is now reduced to 0.4 dB, equivalent to a 960 km reach increase. SABM-SR gains 0.23 dB over SABM and 0.63 dB compared to iBDD, equivalent to a 640 km and 1600 km reach extension, respectively. Despite being a *full* SD decoder, TPD shows *only* a 0.45 dB improvement over SABM-SR, which translates to a 1340 km reach increase.

5 Conclusions

We presented a novel soft-aided hard-decision decoder for product codes based on the principle of marking bits using soft information from the channel. Updating the bit marking at each iteration via scaled reliabilities yields up to 0.3 dB gain (or 960 km reach increase) compared to the standard SABM algorithm. SABM-SR outperforms iBDD by up to 0.8 dB (2400 km reach increase) while preserving its core algebraic structure. We believe this new scheme is a strong candidate for 400G and beyond optical transponders.

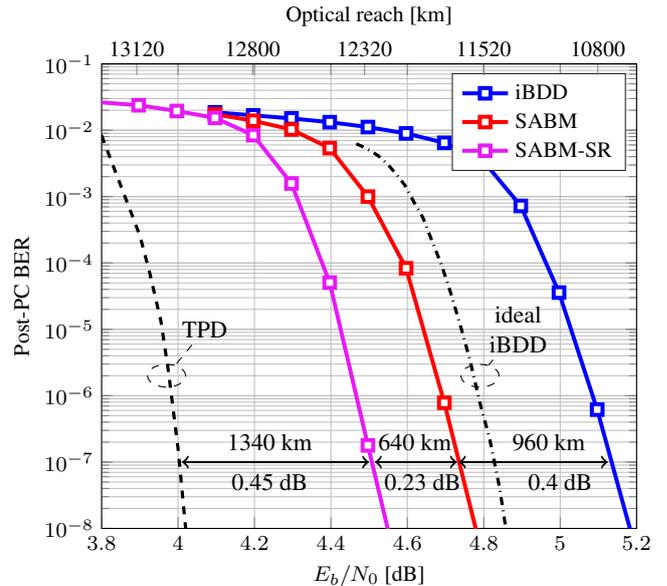| Parameter Name	Value
Modulation Format	PM-QPSK
Symbol rate	33 GBaud
Channel Spacing	33 GHz
No. of Channels	295
Roll-off factor	0
Fibre Type	Standard single-mode fibre
Span Length	80 km
EDFA noise figure	4.5 dB

Table 1: Optical system paramaters.

3

References

[1] P. Elias: 'Error-Free Coding', Trans. IRE Professional Group on Information Theory, 1954, **4**, (4), pp 29–37

[2] B. P. Smith, A. Farhood, A. Hunt, et al.: 'Staircase Codes: FEC for 100 Gb/s OTN', IEEE Journal of Lightwave Technology, 2012, **30**, (1), pp 110–117

[3] C. Häger, and H. D. Pfister: 'Approaching Miscorrection-Free Performance of Product Codes With Anchor Decoding', IEEE Transactions on Communications, 2017, **66**, (7), pp 2797–2808

[4] A. Sheikh, A. G. i Amat, and G. Liva: 'Binary Message Passing Decoding of Product Codes Based on Generalized Minimum Distance Decoding', 2019, available online: http://arxiv.org/abs/1901.02914

[5] Y. Lei, A. Alvarado, B. Chen, et al.: 'Decoding Staircase Codes with Marked Bits,' Proc. International Symposium on Turbo Codes and Iterative Information Processing (ISTC), Hong Kong, China, 2018

[6] A. Sheikh, A. G. i. Amat, G. Liva: 'Iterative Bounded Distance Decoding of Product Codes with Scaled Reliability', Proc. European Conference on Optical Communications, Rome, Italy, 2018

[7] A. Sheikh, A. G. i. Amat, G. Liva, et al.: 'On Low-complexity Decoding of Product Codes for High-throughput Fiber Optic Systems', Proc. IEEE 10th International Symposium on Turbo Codes and Iterative Information Processing (ISTC), Hong Kong, China, 2018

[8] C. Fougstedt, A. Sheikh, A. Graell i Amat, et al.: 'Energy-efficient soft-assisted product decoders', Proc. Optical Fiber Communciations Conference and Exposition (OFC), San Diego, USA, 2019

[9] Y. Lei, B. Chen, G. Liga, et al.: 'Improved Decoding of Staircase Codes: The Soft-aided Bit-marking (SABM) Algorithm', 2019, available online: http://www.arxiv.org/abs/1902.01178

[10] R. M. Pyndiah: 'Near–Optimum Decoding of Product Codes: Block Turbo Codes', IEEE Transactions on Communications, 1998, **46**, (8), 1003–1010

[11] P. Poggiolini, G. Bosco, A. Carena, et al.: 'A Simple and Effective Closed-form GN Model Correction Formula Accounting for Signal Non-Gaussian Distribution', Journal of Lightwave Technology, 2015, **33**, (2), 459–473