REPORTS IN INFORMATICS

ISSN 0333-3590

On the Design of Bit-Interleaved Turbo-Coded Modulation with Low Error Floors

Eirik Rosnes and Øyvind Ytrehus

REPORT NO 267 **March 2004**

Department of Informatics

UNIVERSITY OF BERGEN

Bergen, Norway

This report has URL http://www.ii.uib.no/publikasjoner/texrap/ps/2004-267.ps

Reports in Informatics from Department of Informatics, University of Bergen, Norway, is available at http://www.ii.uib.no/publikasjoner/texrap/.

Requests for paper copies of this report can be sent to:

Department of Informatics, University of Bergen, Høyteknologisenteret,
P.O. Box 7800, N-5020 Bergen, Norway

On the Design of Bit-Interleaved Turbo-Coded Modulation with Low Error Floors

Eirik Rosnes and Øyvind Ytrehus*
Department of Informatics
University of Bergen
Norway
{eirik, oyvind}@ii.uib.no

Abstract

In this report we introduce an algorithm to optimize the performance in the error floor region of bit-interleaved turbo-coded modulation (BITCM) on the additive white Gaussian noise (AWGN) channel. The key ingredient is an *exact* turbo code weight distribution algorithm producing a list of all codewords in the underlying turbo code of weight less than a given threshold. In BITCM, the information sequence is turbo-encoded, bit-interleaved, and mapped to signal points in a signal constellation. Using the union bounding technique, we show that a well-designed bit-interleaver is crucial to have a low error floor. Furthermore, the error rate performance in the waterfall region depends on the bit-interleaver, since the level of protection from channel noise on the bit-level depends on the bit-position and the neighboring bit-values within the same symbol in the transmitted sequence. We observe a trade-off between error rate performance in the waterfall and error floor regions as illustrated by an extensive case study of a high-rate BITCM scheme. The reported case study shows that it is possible to design bit-interleavers with our proposed algorithm with equal or better performance in the waterfall region and superior performance in the error floor region compared to randomly generated bit-interleavers. In particular, we were able to design BITCM schemes with maximum-likelihood decoding frame error rates of 10^{-12} and 10^{-17} at 2.6 dB and 3.8 dB away from unconstrained channel capacity at spectral efficiencies of 3.10 and 6.20 b/s/Hz using square 16 and 256-QAM signal constellations, respectively.

1 Introduction

Bit-interleaved coded modulation (BICM) is a bandwidth-efficient coding scheme based on serial concatenation of a binary error-correcting code, bit-interleaving, and high-order modulation [1]. In more detail, the information sequence is encoded by the error-correcting code, after which the encoded bits are interleaved and mapped to a signal constellation. The decoder first computes log-likelihood ratio (LLR) values of the coded bits from the soft output of the channel. These LLR values are then sent to a soft-input binary decoder. In particular, when employing state-of-the-art codes, such as turbo codes [2] and low-density parity-check codes [3], BICM is a very powerful technique.

In general, it is possible to design BICM schemes using any lattice-based two-dimensional signal constellation. In [4], the problem of finding the most suitable signal sets for designing power-efficient BICM schemes, on the AWGN channel, was addressed. In particular, for 2^{2m} ($m \geq 1$) signal points, the square 2^{2m}-QAM constellation with Gray labeling gives good performance.

*This work was supported by Nera Research and the Norwegian Research Council (NFR) Grants 156712/220 and 146874/420. Parts of this work have been accepted for presentation at the 2004 IEEE International Conference on Communications (ICC) and at the 2004 IEEE International Symposium on Information Theory (ISIT).

In this work we consider square 2^{2m}-QAM constellations with distance-preserving labeling, meaning that the signal points are labeled such that the squared Euclidean distance between any two points is lower bounded by the Hamming distance of the corresponding signal point labels. The distance-preserving property is satisfied for square 2^{2m}-QAM constellations when the signal points are labeled independently with a Gray code in each direction. However, for standard 2^{2m+1}-CROSS constellations, a distance-preserving labeling does not exist [5]. The importance of a distance-preserving labeling is discussed in Section 2.3.

The error floor of BITCM on the AWGN channel is our main interest in this work. Recently, several algorithms have been introduced to compute the first few terms of the weight distribution of both parallel and serial turbo codes. Both exact algorithms (e.g. [6, 7, 8, 9]) and approximate algorithms (e.g. [10]) have been introduced. For some channels, estimates of the minimum Hamming distance and the corresponding multiplicity may be sufficient to get a rough estimate of the performance in the error floor region. However, for BITCM an accurate list of low-weight codewords is essential for estimating the error floor. Moreover, such a list in conjunction with a well-designed bit-interleaver could lower the error floor substantially, as illustrated in the following sections. We will present a greedy algorithm to construct a bit-interleaver from a given list of low-weight codewords and compare the constructed bit-interleavers with randomly generated bit-interleavers.

This report is organized as follows: In Section 2 we introduce the transmission model, discuss constellation labeling, and give the union upper bound on the frame error rate (FER) under a maximum-likelihood decoding assumption. Section 3 introduces a greedy bit-interleaver design algorithm to lower the error floor. We further discuss the different levels of protection on the bit-level from channel noise due to high-order modulation, and the impact of the bit-interleaver on the performance in the waterfall region. An extensive case study of a high-rate BITCM scheme is reported in Section 4. Conclusions and a discussion of future work are given in Section 5.

2 Preliminaries

2.1 Transmission scheme

We consider the transmission scheme depicted in Fig. 1 which follows the principle of BICM. The transmission scheme contains a turbo code, a bit-interleaver, and a modulator that maps binary sequences to sequences of signal points from a two-dimensional constellation with a distance-preserving labeling. The receiver computes symbol-wise LLR values from the received signal sequence. These symbol-wise LLR values are converted to bit-wise LLR values, deinterleaved, and sent to a binary turbo decoder. In this study we will focus on high-rate turbo codes composed of high-rate non-punctured constituent convolutional codes. To reduce decoding complexity, the maximum *a posteriori* (MAP) decoding algorithm can be applied to the *dual* codes of the constituent codes [11].

2.2 Constellation labeling

Let $\mathbf{b} = (b_0 \cdots b_{M-1}) \in \{0,1\}^M$, for a positive integer M, denote the binary label of some signal point $s = \rho(\mathbf{b})$ in a two-dimensional signal constellation S of cardinality 2^M, where ρ denotes the bijective mapping $\rho : \{0,1\}^M \to S$ between signal points and their labels. We define the mapping φ of $\mathbf{b} \neq (0 \cdots 0)$ and a positive integer d, which gives the fraction of all pairs of distinct signal points of binary label *difference* $\mathbf{b}$ which also correspond to a squared Euclidean distance of d. Let $w(\mathbf{b})$ be the Hamming weight of $\mathbf{b}$. With a distance-preserving labeling, as briefly described in Section 1, $d \geq w(\mathbf{b})$ when $\varphi(\mathbf{b}, d) \neq 0$.

A 2^m-PAM constellation, denoted by $S_{\mathrm{PAM}}(m)$, of *size* 2^m is the sequence

$$\left(s_m^{(0)}, s_m^{(1)}, \ldots, s_m^{(2^m-1)}\right) = (-2^m + 1/2, -2^m + 3/2, \ldots, -1/2, 1/2, \ldots, 2^m - 1/2) \qquad (1)$$

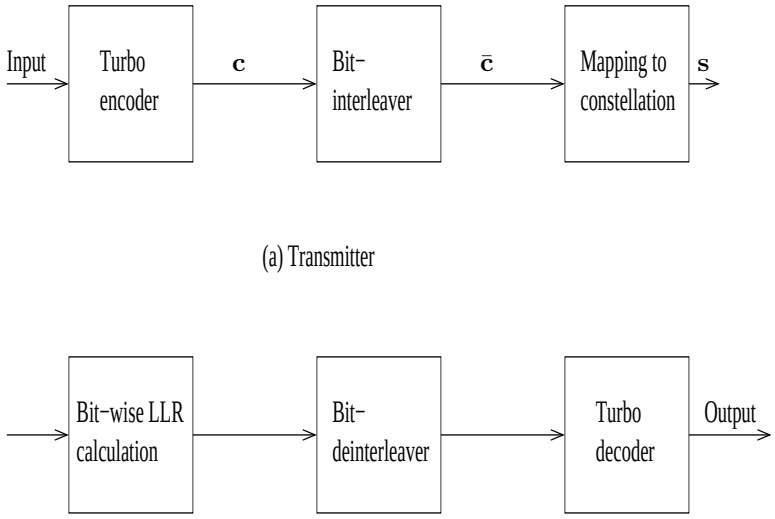

Fig. 1. Block diagram of system model.

of rational numbers, for a positive integer m. Furthermore, a square 2^{2m}-QAM constellation, denoted by $S_{\mathrm{QAM}}(m)$, of *size* 2^{2m} is the sequence of complex numbers where the real and imaginary parts are both taken from $S_{\mathrm{PAM}}(m)$, for a positive integer m. A 2^m-PSK constellation, denoted by $S_{\mathrm{PSK}}(m)$, of *size* 2^m is the sequence

$$\left(s_m^{(0)}, s_m^{(1)}, \ldots, s_m^{(2^m-1)}\right) = \left(A_m, A_m \exp\left(\frac{\sqrt{-1}\,\pi}{2^{m-1}}\right), \ldots, A_m \exp\left(\frac{\sqrt{-1}\,(2^m-1)\pi}{2^{m-1}}\right)\right)$$

of complex numbers with $A_m = 1/(2\sin(\pi/2^m))$, for a positive integer m.

A binary Gray code $\mathbf{g}_m = (\mathbf{g}_m^{(0)}, \ldots, \mathbf{g}_m^{(2^m-1)})$, $m \geq 1$, is a sequence of *all* distinct binary m-tuples for which any two consecutive m-tuples differ in exactly one position. A well-known way to recursively construct Gray codes is by "reflection" as described next. Define $\mathbf{g}_1 = (0, 1)$. Let $m \geq 1$ and construct a Gray code $\mathbf{g}_{m+1}$ of *length* 2^{m+1} recursively from $\mathbf{g}_m$ as follows

$$\mathbf{g}_{m+1} = (\mathbf{g}_m^{(0)}0, \ldots, \mathbf{g}_m^{(2^m-1)}0, \mathbf{g}_m^{(2^m-1)}1, \ldots, \mathbf{g}_m^{(0)}1) \tag{2}$$

where $\mathbf{g}_m^{(i)}0$ and $\mathbf{g}_m^{(i)}1$ denote the concatenation of the m-tuple $\mathbf{g}_m^{(i)}$, $0 \leq i \leq 2^m - 1$, with 0 and 1, respectively.

Example 1 *The two sequences*

$$(000, 001, 011, 111, 101, 100, 110, 010) \textit{ and } (000, 100, 110, 010, 011, 111, 101, 001)$$

are both Gray codes of which the latter is constructed recursively using the above procedure.

Consider a Gray code $\mathbf{g}_m$, $m \geq 1$, of length 2^m and the constellation labeling ρ_{PAM} (ρ_{PSK}) of $S_{\mathrm{PAM}}(m)$ ($S_{\mathrm{PSK}}(m)$) defined by $\mathbf{g}_m^{(i)} \mapsto s_m^{(i)}$ for all i, $0 \leq i \leq 2^m - 1$. This type of labeling of $S_{\mathrm{PAM}}(m)$ ($S_{\mathrm{PSK}}(m)$) is called *Gray labeling*. In the case that the Gray code is constructed recursively as described in (2) we use the term *reflected Gray labeling*.

Table 1. The nonzero values of $\varphi_{\mathrm{PAM}}(b_0b_1b_2, d)$ with the first and the second Gray code of Example 1 when $w(b_0b_1b_2) = 1$ are displayed in (a) and (b), respectively. Furthermore, the nonzero values of $\varphi_{\mathrm{PSK}}(b_0b_1b_2, d)$ with the second Gray code of Example 1 when $w(b_0b_1b_2) = 1$ are displayed in (c).

		d			
		1	9	25	49
b	001	1/2	1/4	1/4	0
	010	3/4	0	0	1/4
	100	1/2	1/4	1/4	0

(a)

		d			
		1	9	25	49
b	001	1/4	1/4	1/4	1/4
	010	1/2	1/2	0	0
	100	1	0	0	0

(b)

		d	
		1	$(1+\sqrt{2})^2$
b	001	1/2	1/2
	010	1/2	1/2
	100	1	0

(c)

Suppose two constellation labelings of $S_{\mathrm{PAM}}(m)$, denoted by $\rho^{(1)}_{\mathrm{PAM}}$ and $\rho^{(2)}_{\mathrm{PAM}}$, are given. Consider the constellation labeling ρ_{QAM} of $S_{\mathrm{QAM}}(m)$ defined by $(b_0 \cdots b_{2m-1}) \mapsto a + \sqrt{-1}\, b$ for all $(b_0 \cdots b_{2m-1}) \in \{0,1\}^{2m}$ where $\rho^{(1)}_{\mathrm{PAM}}(b_0 \cdots b_{m-1}) = a$ and $\rho^{(2)}_{\mathrm{PAM}}(b_m \cdots b_{2m-1}) = b$. This type of labeling of $S_{\mathrm{QAM}}(m)$ is called *independent labeling*. In the case that $\rho^{(1)}_{\mathrm{PAM}} = \rho^{(2)}_{\mathrm{PAM}} = \rho_{\mathrm{PAM}}$ and ρ_{PAM} is a (reflected) Gray labeling, ρ_{QAM} is said to be a *(reflected) Gray independent labeling* of $S_{\mathrm{QAM}}(m)$.

Note that a (reflected) Gray independent labeling is a distance-preserving labeling, and that any distance-preserving independent labeling is a Gray independent labeling. Consider an arbitrary distance-preserving labeling ρ of $S_{\mathrm{QAM}}(m)$. For $m = 1$ and 2, it is relatively easy to prove that there exists a Gray independent labeling $\bar{\rho} = \rho \circ \kappa_\pi$ where $\circ$ denotes composition of mappings, π is a permutation of the indexes $0, \ldots, 2m-1$, and κ_π is defined by $(b_0 \cdots b_{2m-1}) \mapsto (b_{\pi(0)} \cdots b_{\pi(2m-1)})$. Thus, due to the bit-interleaver, we can say without loss of generality that any distance-preserving labeling is a Gray independent labeling for $m = 1$ and 2.

When we consider $S_{\mathrm{QAM}}(m)$ in the rest of this work, we choose (reflected) Gray independent labeling due to the distance-preserving property and the fact that an independent labeling reduces the complexity of calculating bit-wise LLR values at the decoder. Furthermore, for $S_{\mathrm{QAM}}(m)$ with independent labeling, the union upper bound on the FER on the AWGN channel that will be presented in Section 2.3 uses only on the mapping φ for the constituent labelings of $S_{\mathrm{PAM}}(m)$, and not the mapping φ for $S_{\mathrm{QAM}}(m)$, i.e., sequences of symbols from S_{QAM} can be regarded as sequences of pairs of symbols from $S_{\mathrm{PAM}}(m) \times S_{\mathrm{PAM}}(m)$.

In the rest of this work, we use the notation $\varphi_{\mathrm{PAM}}(b_0 \cdots b_{m-1}, d)$, $\varphi_{\mathrm{PSK}}(b_0 \cdots b_{m-1}, d)$, and $\varphi_{\mathrm{QAM}}(b_0 \cdots b_{2m-1}, d)$ when it is important to highlight the underlying constellation.

Example 2 *Consider two Gray labelings of* $S_{\mathrm{PAM}}(3)$ *using the two Gray codes of Example 1. The nonzero values of* $\varphi_{\mathrm{PAM}}(b_0b_1b_2, d)$ *when* $w(b_0b_1b_2) = 1$ *are tabulated in Tables 1(a) and (b), respectively. As an example, we have also considered a Gray labeling of* $S_{\mathrm{PSK}}(3)$ *using the second Gray code from Example 1. The nonzero values of* $\varphi_{\mathrm{PSK}}(b_0b_1b_2, d)$ *when* $w(b_0b_1b_2) = 1$ *are tabulated in Table 1(c). As a remark, note that we get a table with the same* structure *if we use the first instead of the second Gray code from Example 1. As shown in Tables 1(a) and (b) all Gray codes are not equal (which was also observed in [5] in the context of the* edge profile*).*

Lemma 1 *For $S_{\mathrm{PAM}}(m)$, $m \geq 1$, with reflected Gray labeling it holds, for $0 \leq x \leq m-1$, that*

$$\varphi\left(0^x 10^{m-1-x}, d\right) = \begin{cases} \left(\frac{1}{2}\right)^x & \text{if } d = (2i+1)^2 \text{ for all } i \in \{0, 1, \ldots, 2^x - 1\}, \\ 0 & \text{otherwise}, \end{cases} \tag{3}$$

where 0^x is a shorthand notation for a sequence of x zeros.

Proof: We prove the result by induction on m. For $m = 1$, (3) reduces to

$$\varphi(1, d) = \begin{cases} 1 & \text{if } d = 1, \\ 0 & \text{otherwise}, \end{cases}$$

which is obviously true. Assume that the result in Lemma 1 holds for some $m = l$, $l \geq 1$. Consider the case of $m = l + 1$. It follows from the induction hypotheses and the recursive construction of Gray codes in (2) that (3) is also true for $m = l+1$ when $0 \leq x \leq l-1$. It remains to prove that (3) holds for $m = l+1$ and $x = l$. For $m = l+1$ and $x = l$, it follows from the recursive construction of Gray codes in (2) that the pair of signal points (s_1, s_2) in which the index of s_1 in $S_{\mathrm{PAM}}(l+1)$ is i and the index of s_2 in $S_{\mathrm{PAM}}(l+1)$ is $2^{l+1}-1-i$ for some i, $0 \leq i \leq 2^l - 1$, has binary label difference $(0^l 1)$ and squared Euclidean distance $(2(2^l - 1 - i) + 1)^2$. In fact, this is the only pair of signal points with binary label difference $(0^l 1)$ and squared Euclidean distance $(2(2^l - 1 - i) + 1)^2$ for a fixed i, $0 \leq i \leq 2^l - 1$. Furthermore, there does not exist pairs of signal points with binary label difference $(0^l 1)$ and squared Euclidean distance different from $(2(2^l - 1 - i) + 1)^2$, $0 \leq i \leq 2^l - 1$. The result in (3) for $m = l + 1$ and $x = l$ follows immediately. □

Example 3 *Consider the square 16-QAM signal constellation with reflected Gray independent labeling as depicted in Fig. 2. In this case, as an example,*

$$\varphi_{\mathrm{QAM}}(0001, 1) = 0.5,\ \varphi_{\mathrm{QAM}}(0001, 9) = 0.5,\ \varphi_{\mathrm{QAM}}(0101, 2) = 0.25,$$
$$\varphi_{\mathrm{QAM}}(0101, 10) = 0.5,\ \varphi_{\mathrm{QAM}}(1000, 1) = 1,\ \textit{and}\ \varphi_{\mathrm{QAM}}(1110, 5) = 1.$$

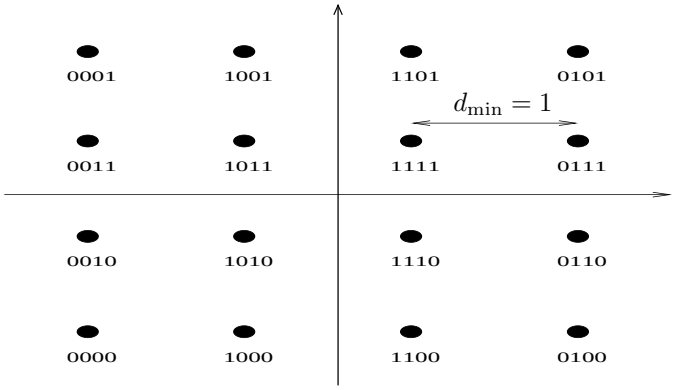

Fig. 2. Square 16-QAM signal constellation with reflected Gray independent labeling.

Consider sequences $\boldsymbol{\alpha} = (\alpha_0, \ldots, \alpha_{l-1})$ and $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_{l-1})$ of rational numbers of length l. We say that $\boldsymbol{\alpha}$ *dominates* $\boldsymbol{\beta}$ if and only if there exists an integer i, $0 \leq i \leq l-1$, such that $\alpha_j = \beta_j$ for all j, $0 \leq j < i$, and $\alpha_i < \beta_i$. Furthermore, for a given constellation labeling ρ of $S_{\mathrm{PAM}}(m)$ order the nonzero values of $\varphi(b_0 \cdots b_{m-1}, 1)$ when $w(b_0 \cdots b_{m-1}) = 1$ in an increasing fashion. The resulting ordered sequence is denoted by $\boldsymbol{\varphi}_\rho$.

Lemma 2 *There exists a Gray labeling $\bar{\rho}$ of $S_{\mathrm{PAM}}(m)$, $m \geq 1$, such that the ith component of $\boldsymbol{\varphi}_{\bar{\rho}}$ is equal to $(1/2)^{m-i-1}$, $0 \leq i \leq m-1$. Furthermore, there does not exist a different Gray labeling ρ of $S_{\mathrm{PAM}}(m)$ such that $\boldsymbol{\varphi}_\rho$ dominates $\boldsymbol{\varphi}_{\bar{\rho}}$.*

Proof: The existence of a Gray labeling $\bar{\rho}$ with the property of having the ith component of $\boldsymbol{\varphi}_{\bar{\rho}}$ equal to $(1/2)^{m-i-1}$, $0 \leq i \leq m-1$, follows directly from Lemma 1, i.e., $\bar{\rho}$ is a reflected Gray labeling. The second part of Lemma 2 is proved by contradiction. Assume that there exists a Gray labeling ρ of $S_{\mathrm{PAM}}(m)$ such that $\boldsymbol{\varphi}_{\rho}$ dominates $\boldsymbol{\varphi}_{\bar{\rho}}$. From the properties of a Gray code there exists at least one integer j, $0 \leq j \leq 2^m - 2$, such that $w(\rho^{-1}(s_m^{(j)}) \oplus \rho^{-1}(s_m^{(j+1)})) = (0^x 10^{m-1-x})$ for *all* integers x, $0 \leq x \leq m-1$, where $\oplus$ denotes modulo 2 addition. Since $\boldsymbol{\varphi}_{\rho}$ dominates $\boldsymbol{\varphi}_{\bar{\rho}}$, we can find an integer x, $0 \leq x \leq m-1$, and a unique integer j, $0 \leq j \leq 2^m - 2$, such that $w(\rho^{-1}(s_m^{(j)}) \oplus \rho^{-1}(s_m^{(j+1)})) = (0^x 10^{m-1-x})$, which imply 1) $j = 2^{m-1} - 1$, 2) that the leftmost 2^{m-1} points have to agree in the xth bit-position, and 3) that the rightmost 2^{m-1} points have to agree in the xth bit-position. Applying the argumentation above on both $s_m^{(0)}, \ldots, s_m^{(j)}$ and $s_m^{(j+1)}, \ldots, s_m^{(2^m-1)}$, and recursively on each new non-empty subset of consecutive signal points encountered during the process result in a sequence $\boldsymbol{\varphi}_{\rho}$ which is equal to $\boldsymbol{\varphi}_{\bar{\rho}}$. Thus, $\boldsymbol{\varphi}_{\rho}$ does not dominate $\boldsymbol{\varphi}_{\bar{\rho}}$ and a contradiction occurs. □

2.3 Bounding the error rate performance

Consider a signal constellation S of cardinality 2^M, $M \geq 1$, with some constellation labeling ρ. Under a maximum-likelihood decoding assumption, the union bounding technique gives an accurate estimate of the error rate performance in the high signal-to-noise ratio region. The contribution from each signal sequence $\mathbf{s} = (s_0, \ldots, s_{n/M-1})$ (n being the code length) of Hamming weight $w(\mathbf{s})$ and *support* $\{i_1, \ldots, i_{w(\mathbf{s})}\}$ to the union bound on the FER on an AWGN channel is

$$\sum_{\substack{d_1 \in \mathbb{N}: \\ \varphi(\rho^{-1}(s_{i_1}), d_1) \neq 0}} \cdots \sum_{\substack{d_{w(\mathbf{s})} \in \mathbb{N}: \\ \varphi(\rho^{-1}(s_{i_{w(\mathbf{s})}}), d_{w(\mathbf{s})}) \neq 0}} \left[\prod_{j=1}^{w(\mathbf{s})} \varphi(\rho^{-1}(s_{i_j}), d_j) \right] Q\left(\sqrt{\frac{d_1 + \cdots + d_{w(\mathbf{s})}}{2N_0}} \right) \tag{4}$$

where $\mathbb{N} = \{1, 2, \ldots\}$ is the natural numbers, $N_0/2$ is the noise variance in each direction (real and imaginary) of the complex additive white Gaussian noise, and $Q(\cdot)$ is the standard Q-function. The bit energy E_b, needed when plotting the union bound as a function of E_b/N_0, for $S_{\mathrm{QAM}}(m)$, is $(2^{2m}-1)d_{\min}^2/(12mR)$ where R is the turbo code rate and $d_{\min}$ is the minimum Euclidean distance between any two signal points in the signal constellation [12] (see Fig. 2). In our case we have $d_{\min} = 1$ from (1). The overall bound is obtained by summing the above expression over all signal sequences.

In practice, the summation is truncated and only signal sequences $\mathbf{s}$ with $w(\rho^{-1}(\mathbf{s}))$ less than a given threshold $\tau \geq d_{\mathrm{free}}$ (d_{free} being the free distance of the turbo code) are considered. Note that when applied to a sequence $\mathbf{s}$ of symbols from a constellation S, ρ^{-1} is applied to each component individually. When the summation is truncated, only terms in (4) with $d_1 + \cdots + d_{w(\mathbf{s})} \leq \tau$ are needed. This reduces the number of terms in the summation significantly and allows an efficient recursive computation of (4).

A distance-preserving labeling guarantees that the contribution of every signal sequence pair $(\mathbf{s}, \mathbf{s}')$ with $d_{\mathrm{E}}^2(\mathbf{s}, \mathbf{s}') \leq \tau$ is taken into account in a truncated union bound when all signal sequences $\tilde{\mathbf{s}}$ with $w(\rho^{-1}(\tilde{\mathbf{s}})) \leq \tau$ are summed using (4). Here, $d_{\mathrm{E}}(\cdot, \cdot)$ denotes Euclidean distance. In particular, with a non-distance-preserving labeling there could exist signal sequences $\tilde{\mathbf{s}}$, $\mathbf{s}$, and $\mathbf{s}'$ such that $\tilde{\mathbf{s}}$ corresponds to a binary turbo codeword of rather large Hamming weight $> \tau$, $d_{\mathrm{E}}^2(\mathbf{s}, \mathbf{s}') \leq \tau$, and $\rho^{-1}(\mathbf{s}) \oplus \rho^{-1}(\mathbf{s}') = \rho^{-1}(\tilde{\mathbf{s}})$. The contribution of the signal sequence pair $(\mathbf{s}, \mathbf{s}')$ will not be taken into account in a truncated union bound, since by assumption we do not know binary turbo codewords of weight $> \tau$.

In the case of $S_{\mathrm{QAM}}(m)$ with a Gray independent labeling ρ, we may look at the signal sequences as sequences of length n/m of symbols from $S_{\mathrm{PAM}}(m)$. The contribution from each signal sequence $\mathbf{s} = (s_0, \ldots, s_{n/m-1})$ to the union bound on the FER on an AWGN channel is given by the expression in (4). Note that in this case all variables are related to $S_{\mathrm{PAM}}(m)$ and $\mathbf{s}$ as a sequence

of symbols from $S_{\mathrm{PAM}}(m)$. For instance, the mapping φ is computed for $S_{\mathrm{PAM}}(m)$ with Gray labeling.

3 Design of bit-interleaver

In this section we propose a greedy bit-interleaver design algorithm for improved performance in the error floor region that uses a set L consisting of all codewords of weight less than a given threshold τ obtained from the algorithm in [8]. We further discuss the different levels of protection on the bit-level from channel noise due to high-order modulation, and the impact of the bit-interleaver on the performance in the waterfall region.

3.1 Greedy bit-interleaver design

For any codeword $\mathbf{c} = (c_0, \ldots, c_{n-1}) \in L$, let $X(\mathbf{c})$ be the support of $\mathbf{c}$, i.e., $X(\mathbf{c}) = \{i : c_i = 1\}$, and let $X = X(L) = \cup_{\mathbf{c}\in L} X(\mathbf{c})$, i.e., the support of L. A subset H of X of minimum cardinality such that $|X(\mathbf{c}) \cap H| \geq l(\mathbf{c})$, $0 \leq l(\mathbf{c}) \leq |X(\mathbf{c})|$, for all $\mathbf{c} \in L$ is called a *minimum (generalized) hitting set*, and the values $\{l(\mathbf{c}) : \mathbf{c} \in L\}$ constitute a prescribed *hitting distribution*. If we choose $l(\mathbf{c}) = 1$ for all $\mathbf{c} \in L$, then the problem of finding a minimum hitting set is equivalent to finding a *minimum set cover* [13] which is an $\mathcal{NP}$-hard problem. In the following we choose $l(\mathbf{c}) = l'(w(\mathbf{c}))$, i.e., the value depends only on the Hamming weight of $\mathbf{c}$. For each position $p \in X$, let $N(p) = |\{\mathbf{c} \in L : p \in X(\mathbf{c})\}|$. A hitting set can be constructed by the greedy algorithm below. Note that in general a minimum hitting set is *not* constructed.

Greedy Hitting Set$(L, \{l(\mathbf{c}) : \mathbf{c} \in L\})$**:**

/* Find a hitting set H with a target hitting distribution
$\{l(\mathbf{c}) : \mathbf{c} \in L\}$. */
Compute $X = X(L)$ and $N(p)$ for all $p \in X$, and
start with an empty set H.
(*) **If** $|X(\mathbf{c}) \cap H| \geq l(\mathbf{c})$ for all $\mathbf{c} \in L$ or $X = \emptyset$,
terminate the process.
Otherwise,
set $p_{\max} = \arg\max_{p\in X} N(p)$.
If $\exists\, \mathbf{c} \in L : X(\mathbf{c}) \cap \{p_{\max}\} \neq \emptyset$ and $|X(\mathbf{c}) \cap H| < l(\mathbf{c})$,
set $H = H \cup \{p_{\max}\}$.
Remove $p_{\max}$ from X.
Proceed from (*).

Consider $S_{\mathrm{QAM}}(m)$ with a reflected Gray independent labeling. We construct a bit-interleaver randomly under the constraint that the *first* n/m positions in the hitting set H are mapped to positions $\equiv m-1$ modulo m. If $n/m < |H| \leq 2n/m$, $m \geq 2$, then the *last* $|H| - n/m$ positions in H are mapped to positions $\equiv m-2$ modulo m. This principle is motivated by Lemmas 1 and 2 and equation (4). In more detail, the contribution to the multiplicity of the Q-function in (4) is *smallest* (equal to $(1/2)^{m-1}$) when the nonzero symbols $\{s_{i_j}\}$ have labels with a single one in position $m-1$. Note that in this case the nonzero symbols $\{s_{i_j}\}$ are symbols from $S_{\mathrm{PAM}}(m)$. This bit-interleaver construction algorithm will be denoted by Greedy Interleaver Design (GID). When the target hitting distribution is important we use the notation GID$(\{l(\mathbf{c}) : \mathbf{c} \in L\})$.

In the context of turbo codes, a large fraction of the positions in the hitting set are parity positions, since the low-weight codewords have low input weight, which will be mapped to positions $\equiv m-1$ modulo m.

Consider a turbo codeword $\mathbf{c}$ of Hamming weight $w(\mathbf{c}) \geq 1$. We may look at the signal sequences as sequences of length n/m of symbols from $S_{\mathrm{PAM}}(m)$. Suppose x, $1 \leq x \leq w(\mathbf{c})$, 1-positions in the codeword are mapped into a symbol $s \in S_{\mathrm{PAM}}(m)$. From the properties of a Gray code, it follows that $\varphi\left(\rho^{-1}(s), d\right) = 0$ for all integers $d \leq x_1^2 - 1$. Since $x_1^2 \geq x_1$ with equality if and only

if $x_1 = 1$, it follows that the event of mapping the 1-positions from a given codeword into distinct symbols constitutes a worst case scenario in terms of squared Euclidean distance. Thus, we do not have to worry if more than a single 1-position in some low-weight codeword is mapped to a single symbol. Note that this could happen if $n/m < |H| \leq 2n/m$, $m \geq 2$.

3.2 Level of protection from channel noise

For a given high-order modulation scheme with a specific labeling ρ between signal points and binary labels, the level of protection (from channel noise) of a transmitted bit depends on both its position (modulo M) in the transmitted binary sequence and its neighboring bits within the same symbol. In this subsection we give a reliability measure for the LLR values on the bit-level, as seen by the binary turbo decoder, at high signal-to-noise ratios to quantify the different levels of protection.

Consider $S_{\mathrm{QAM}}(m)$ with a reflected Gray independent labeling (see Fig. 2 for an example when $m = 2$). Any symbol $\tilde{s} \in S = S_{\mathrm{QAM}}(m)$ with binary label $(\tilde{b}_0 \cdots \tilde{b}_{2m-1})$ is transmitted across an AWGN channel with noise variance $N_0/2$ in each direction. The received value is $r = \tilde{s} + \epsilon$ where ϵ denotes additive white Gaussian noise. The LLR values on the bit-level are $(0 \leq i \leq m-1)$

$$\begin{aligned}
\log\left(\frac{\Pr(\tilde{b}_i = 1|r)}{\Pr(\tilde{b}_i = 0|r)}\right) &= \log\left(\frac{\Pr(r|\tilde{b}_i = 1)}{\Pr(r|\tilde{b}_i = 0)}\right) + \log\left(\frac{\Pr(\tilde{b}_i = 1)}{\Pr(\tilde{b}_i = 0)}\right) \underset{(a)}{=} \log\left(\frac{\Pr(r|\tilde{b}_i = 1)}{\Pr(r|\tilde{b}_i = 0)}\right) \\
&= \log\left(\frac{\sum_{\substack{\forall\, s\in S:\\ b_i=1}} \Pr(r|\tilde{s} = s)}{\sum_{\substack{\forall\, s\in S:\\ b_i=0}} \Pr(r|\tilde{s} = s)}\right) = \log\left(\frac{\sum_{\substack{\forall\, \Re(s), s\in S:\\ b_i=1}} \Pr(\Re(r)|\Re(\tilde{s}) = \Re(s))}{\sum_{\substack{\forall\, \Re(s), s\in S:\\ b_i=0}} \Pr(\Re(r)|\Re(\tilde{s}) = \Re(s))}\right) \\
&\underset{(b)}{\approx} -\frac{1}{N_0}\left(\min_{\forall\, \Re(s), s\in S: b_i=1} (\Re(r) - \Re(s))^2 - \min_{\forall\, \Re(s), s\in S: b_i=0} (\Re(r) - \Re(s))^2\right) \\
&\underset{(c)}{\approx} \frac{2\tilde{b}_i - 1}{N_0}\left(\tilde{d}_i^2(\tilde{s}) + 2\tilde{d}_i(\tilde{s})\Re(\epsilon)\right) \sim \mathcal{N}\left(\frac{(2\tilde{b}_i - 1)\tilde{d}_i^2(\tilde{s})}{N_0}, \frac{2\tilde{d}_i^2(\tilde{s})}{N_0}\right)
\end{aligned} \tag{5}$$

where $\Re(\cdot)$ denotes the real part of its argument, $\mathcal{N}(\cdot,\cdot)$ is a Gaussian distribution where the first and second arguments are the mean and variance, respectively, and where

$$\tilde{d}_i(\tilde{s}) = \min_{\forall\, \Re(s), s\in S: b_i \neq \tilde{b}_i} d_{\mathrm{E}}(\Re(s), \Re(\tilde{s})). \tag{6}$$

In (5), the equality in (a) follows from the linearity of the underlying code and the assumption of no *a priori* information, the approximation in (b) is the standard "max-log" approximation

$$\log(e^{a_1} + e^{a_2} + \cdots + e^{a_j}) \approx \max(a_1, a_2, \ldots, a_j),\ \forall j > 1,$$

and the approximation in (c) is a high E_b/N_0 approximation. For $m \leq i \leq 2m-1$, (5) and (6) are valid with $\Im(\cdot)$ (the imaginary part) instead of $\Re(\cdot)$. In general, $\Re(\tilde{s})$ and $\Im(\tilde{s})$ take on values from the ordered set (1)

$$\left\{-2^{m-1} + 1/2, -2^{m-1} + 3/2, \ldots, -1/2, 1/2, \ldots, 2^{m-1} - 1/2\right\} \tag{7}$$

of cardinality 2^m. The corresponding values of $\tilde{d}_i(\tilde{s})$, for $0 \leq i \leq m-1$, are contained in the ordered set

$$\left\{\underbrace{2^i, 2^i - 1, \ldots, 1, 1, 2, \ldots, 2^i}_{2^{i+1}}, \ldots, \underbrace{2^i, 2^i - 1, \ldots, 1, 1, 2, \ldots, 2^i}_{2^{i+1}}\right\}. \tag{8}$$

Note that for $m \leq i \leq 2m-1$, $\tilde{d}_i(\tilde{s}) = \tilde{d}_{i-m}(\tilde{s})$.

As argued above the LLR values on the bit-level, at high E_b/N_0, are approximately Gaussian random variables with mean and variance dependent on the sent symbol $\tilde{s}$. A signal-to-noise ratio for the LLR random variable can be defined as the ratio of the square of the mean over the variance [14], which, at position i, is

$$\frac{\left[(2\tilde{b}_i - 1)\tilde{d}_i^2(\tilde{s})/N_0\right]^2}{2\tilde{d}_i^2(\tilde{s})/N_0} = \frac{\tilde{d}_i^2(\tilde{s})}{2N_0}. \tag{9}$$

This signal-to-noise ratio gives a measure for the quality of the LLR values on the bit-level as seen by the binary turbo decoder. For the underlying turbo code, linearity implies that all symbols have the same probability of being sent, and using (8) and the fact that $\tilde{d}_i(\tilde{s}) = \tilde{d}_{i-m}(\tilde{s})$ for $m \leq i \leq 2m-1$, the average signal-to-noise ratio, at position i, becomes

$$\frac{1}{2^{r+1}N_0}\sum_{j=1}^{2^r} j^2 = \frac{(2^r+1)(2^{r+1}+1)}{12N_0} \tag{10}$$

where $i = q \cdot m + r$, $q \in \{0,1\}$ and $r \in \{0,\ldots,m-1\}$, which shows that transmitted bits in positions $\equiv 0$ modulo m ($r = 0$) have the lowest level of protection, while the transmitted bits in positions $\equiv m-1$ modulo m ($r = m-1$) have the highest level of protection.

The iterative turbo decoding algorithm iterates on extrinsic values of information bits only. Consequently, information bits should have the highest level of protection if performance in the waterfall region is the primary concern.

3.3 Discussion

A bit-interleaver construction algorithm giving any bit-interleaver with equal probability will be denoted by RND. RND with the additional constraint of mapping parity positions in a random fashion to the least protected positions, i.e., to positions $\equiv 0$ modulo m will be denoted by RND-PC.

For improved performance in the error floor region we use GID($\{l(\mathbf{c}) : \mathbf{c} \in L\}$). For turbo codes in general, a large fraction of their 1-positions in low-weight codewords are parity positions, since the input weight is low. Consequently, a bit-interleaver from GID($\{l(\mathbf{c}) : \mathbf{c} \in L\}$) will map a larger fraction of parity positions to positions $\equiv m-1$ modulo m (see Section 3.1), which are the most reliable positions (see Section 3.2), compared to a bit-interleaver from RND. This will degrade the performance in the waterfall region as discussed in Section 3.2.

To compensate we could impose a restriction upper bounding the number of parity positions in the hitting set when using Greedy Hitting Set. GID($\{l(\mathbf{c}) : \mathbf{c} \in L\}$) with an upper bound of δ on the number of parity positions in the hitting set and with the additional constraint of mapping the parity positions in the complement of the hitting set in a random fashion to the least protected positions will be denoted by GID-PC($\{l(\mathbf{c}) : \mathbf{c} \in L\}; \delta$). As we will show in a case study in the next section, such a restriction will give bit-interleavers with at least as good performance in the waterfall region and significant performance improvement in the error floor region compared to the empirical average bit-interleaver from RND.

4 Case study

We have looked at a rate $R = 7/9$ parallel turbo code with non-punctured rate 7/8, constraint length $\nu = 4$ constituent convolutional codes with free distance $d_{\text{free}} = 4$ defined by the (*canonical* [15]) parity check matrix

$$\mathbf{H}(D) = \begin{pmatrix} 17 & 23 & 25 & 27 & 33 & 35 & 37 & 31 \end{pmatrix}. \tag{11}$$

The entries in the matrix in (11) are given in octal notation in the sense that $23 = 010011 = 1 + D + D^4$. We choose the last polynomial which is irreducible and primitive as the parity

Table 2. The initial part of the IOWEF of the example turbo code. As an example there are 42 codewords of weight 16 with input weight 4.

		Input weight								
		1	2	3	4	5	6	7	8	9
Output weight	13	-	-	4	2	-	-	-	-	-
	14	-	-	7	6	-	-	-	-	-
	15	1	1	21	22	-	-	-	-	-
	16	1	5	26	42	15	1	-	-	-
	17	-	15	55	75	36	19	2	1	-
	18	-	27	58	168	115	55	6	1	-
	19	1	24	93	283	287	126	32	9	1

polynomial making the constituent encoders recursive. The information bits from the second constituent encoder are punctured to obtain the overall rate of 7/9, and both constituent encoders are terminated. The UMTS termination scheme [16] is used, i.e., both parity and information bits from both encoders are transmitted for the termination sections. The information block size is 4004, which results in codewords of length 5164. The turbo code interleaver is a high spread interleaver [17] with additional constraints [18, 19], and a maximum of 20 iterations are used in the iterative turbo decoding algorithm. The initial part of the code's input-output weight enumerating function (IOWEF), computed by the algorithm in [8], is tabulated in Table 2.

The bit-interleavers are generated according to either RND or RND-PC, or GID($\{l(\mathbf{c}) : \mathbf{c} \in L\}$) or GID-PC($\{l(\mathbf{c}) : \mathbf{c} \in L\}; \delta$) with L composed of all codewords of weight ≤ 17. Both 16-QAM ($m = 2$) and 256-QAM ($m = 4$) signal sets with reflected Gray independent labeling are considered. In the following let $\mathrm{FER}_{\mathrm{UB},\leq\tau}(E_b/N_0)$ denote the truncated union bound on the FER calculated using (4) and all codewords of weight $\leq \tau$ at a target signal-to-noise ratio E_b/N_0. We use the notation $(x_{13}, x_{14}, x_{15}, x_{16}, x_{17})$ with GID and GID-PC for a given target hitting distribution where the value of x_i $(0 \leq x_i \leq i)$, $13 \leq i \leq 17$, indicates that at least x_i 1-positions in every codeword of weight i are used in the design.

In Fig. 5 the probability density functions (pdfs) of $\log(\mathrm{FER}_{\mathrm{UB},\leq\mathrm{x}}(E_b/N_0 = 7.0 \text{ dB}))$, $x = 17$ and 19, are estimated from a large number of bit-interleavers generated independently according to the indicated algorithms for $m = 2$. In particular, the estimated pdf with the algorithm GID$(13, 14, 15, 16, 17)$ is displayed. We observe that there is a very high probability that the truncated union bound on the FER at $E_b/N_0 = 7.0$ dB is at a probability of 10^{-13}. Note that codewords of weight 18 and 19 are taken into account through (4), but not in the design of the bit-interleavers. The narrow shape of the pdf indicates that the computation using (4) has actually converged. The narrow shape of the pdf is a direct consequence of the fact that almost every parity position is contained in the hitting set. Additional codewords of weight $18, 19, 20, \cdots$ will have a large parity weight and thus some of the positions will map to positions with a high protection level, and their influence on the union bound through (4) is limited. As discussed in Section 3.3 these bit-interleavers will probably display a performance worse than the empirical average bit-interleaver from RND in the waterfall region, and indeed this is the case as illustrated in Fig. 6.

Furthermore, the pdf of $\log(\mathrm{FER}_{\mathrm{UB},\leq 17}(E_b/N_0 = 7.0 \text{ dB}))$ is estimated from a large number of bit-interleavers generated independently according to RND and RND-PC. The constraint of mapping all parity positions to positions with the lowest protection level will raise the error floor compared to the empirical average bit-interleaver from RND, but the performance will improve in the waterfall region as illustrated in Fig. 6. The pdf of $\log(\mathrm{FER}_{\mathrm{UB},\leq 19}(E_b/N_0 = 7.0 \text{ dB}))$, estimated from bit-interleavers generated independently according to GID-PC$(13, 14, 15, 5, 4; 644)$, is also displayed in Fig. 5. In this context note that the total number of parity positions (not counting termination bits) is 1144. Note that the pdf has its probability mass at a significantly lower level than the estimated pdf of $\log(\mathrm{FER}_{\mathrm{UB},\leq 17}(E_b/N_0 = 7.0 \text{ dB}))$ of bit-interleavers from RND. The performance in the waterfall region is identical as illustrated in Fig. 6.

Figs. 3 and 4 contain the same information as Fig. 5 but with $m = 4$. In more detail, in Fig. 3 the pdf of $\log(\mathrm{FER}_{\mathrm{UB},\leq 19}(E_b/N_0 = 14.5 \text{ dB}))$ is estimated from a large number of bit-interleavers generated independently according to GID$(13, 14, 15, 16, 17)$ and GID$(13, 14, 15, 12, 11)$. Note that there is a very high probability that the truncated union bound on the FER at $E_b/N_0 = 14.5$ dB is at a probability of 10^{-17} when using the most greedy algorithm, i.e., GID$(13, 14, 15, 16, 17)$. Again, as for the case of $m = 2$, codewords of weight 18 and 19 are taken into account through (4), but not in the design of the bit-interleavers. As noted above the narrow shape of the pdf indicates that the computation using (4) has actually converged. The estimated pdf when using GID$(13, 14, 15, 12, 11)$ illustrates the effect of being less greedy.

In Fig. 4 the pdf of $\log(\mathrm{FER}_{\mathrm{UB},\leq 17}(E_b/N_0 = 14.5 \text{ dB}))$ is estimated from a large number of bit-interleavers generated independently according to RND and RND-PC. We observe that the constraint of mapping all parity positions to positions with the lowest protection level will raise the error floor substantially compared to the empirical average bit-interleaver from RND, but the performance will improve in the waterfall region as illustrated in Fig. 7. Furthermore, the pdf of $\log(\mathrm{FER}_{\mathrm{UB},\leq 19}(E_b/N_0 = 14.5 \text{ dB}))$ estimated from bit-interleavers generated independently according to GID-PC(13,14,15,5,4;644) is also displayed in Fig. 4. From Figs. 4 and 7 we observe that the empirical average bit-interleaver from GID-PC(13,14,15,5,4;644) gives slightly *better* performance in the waterfall region and superior performance in the error floor region compared to the empirical average bit-interleaver from RND.

Figs. 6 and 7 show simulation results and truncated union bounds, calculated from (4) using all codewords of the underlying turbo code of weight ≤ 19, on the FER of BITCM using several different bit-interleavers generated according to the algorithms in the legends for $m = 2$ and 4, respectively. The tag BAD in the legend of Fig. 7 corresponds to a BITCM scheme with a very high error floor. Note that the truncated union bound is in excellent agreement with the simulation results.

The unconstrained E_b/N_0 channel capacity in dB is $10 \log[(2^\eta - 1)/\eta]$ where η is the spectral efficiency. In our case study $\eta = 3.10$ and 6.20 b/s/Hz for $m = 2$ and 4, respectively. If the error floor is the primary concern, we observe from Figs. 6 and 7 that maximum-likelihood decoding frame error rates of 10^{-12} and 10^{-17} at 2.6 dB and 3.8 dB away from unconstrained channel capacity at spectral efficiencies of 3.10 and 6.20 b/s/Hz using square 16 and 256-QAM signal constellations are achievable, respectively.

5 Conclusion and future work

In this work we have introduced a greedy algorithm to lower the error floor of BITCM in AWGN through a detailed design of the bit-interleaver. The input to the algorithm is an accurate list of low-weight codewords computed from an exact weight distribution algorithm. For high-order modulation, the level of protection from channel noise is dependent on the bit-position in the transmitted sequence and the neighboring bit-values within the same symbol. For improved performance in the waterfall region, the information bits should have the highest level of protection, since the iterative turbo decoding algorithm iterates on extrinsic values of information bits only.

An extensive case study of a high-rate BITCM scheme using square QAM signal constellations shows that there is a trade-off between performance in the waterfall and the error floor regions, since a low error floor requires that the parity bits should have the highest level of protection. The reported case study shows that it is possible to design bit-interleavers with our proposed algorithm with equal or better performance in the waterfall region and superior performance in the error floor region compared to randomly generated bit-interleavers. Moreover, if the error floor is the primary concern, we were able to design BITCM schemes with maximum-likelihood decoding frame error rates of 10^{-12} and 10^{-17} at 2.6 dB and 3.8 dB away from unconstrained channel capacity at spectral efficiencies of 3.10 and 6.20 b/s/Hz using square 16 and 256-QAM signal constellations, respectively.

The bit-interleaver design algorithm described in Section 3.1 is a one-dimensional approach. As future work we would like to investigate a higher-dimensional approach as indicated below. Let

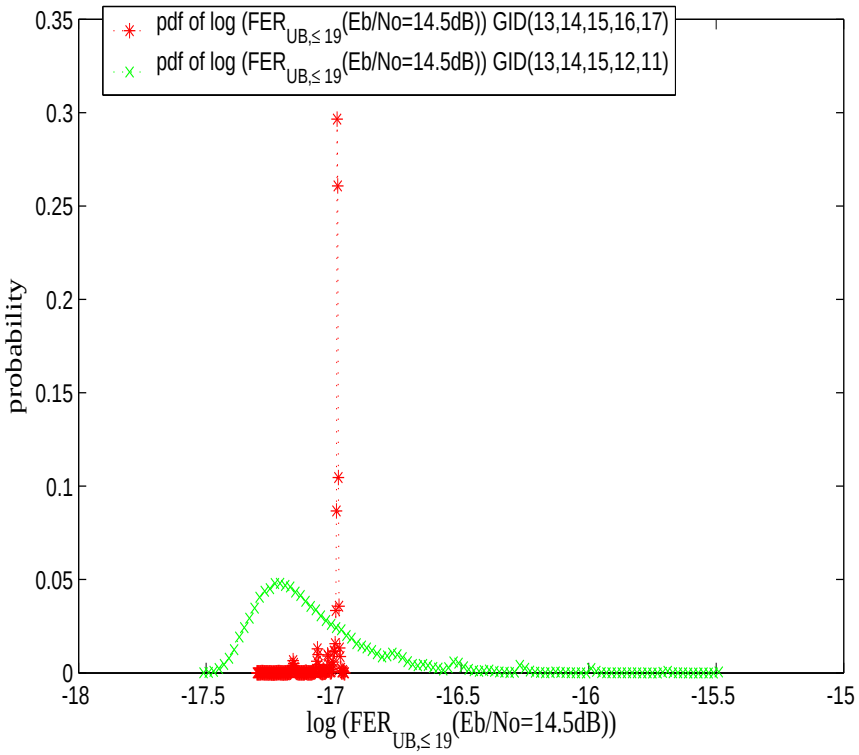

Fig. 3. Estimated probability density functions with $m = 4$ of $\log(\mathrm{FER}_{\mathrm{UB},\leq 19}(E_b/N_0 = 14.5 \text{ dB}))$ from 100000 bit-interleavers generated independently according to $\mathrm{GID}(13, 14, 15, 16, 17)$ and $\mathrm{GID}(13, 14, 15, 12, 11)$ with L composed of all codewords of weight ≤ 17.

$X^{(i)} = X^{(i)}(L)$, $1 \leq i \leq d_{\mathrm{free}}$ (d_{free} being the free distance of the underlying turbo code), be the set of distinct i-tuples of 1-positions of all the codewords in L. For any codeword $\mathbf{c} \in L$, the set of i-tuples of 1-positions in the codeword is a subset of $X^{(i)}$ denoted by $X^{(i)}(\mathbf{c})$. A subset $H^{(i)}$ of $X^{(i)}$ of minimum cardinality such that $|X^{(i)}(\mathbf{c}) \cap H^{(i)}| \geq l^{(i)}(\mathbf{c})$, $0 \leq l^{(i)}(\mathbf{c}) \leq |X^{(i)}(\mathbf{c})|$, for all $\mathbf{c} \in L$ is called a *minimum i-dimensional (generalized) hitting set*, and the values $\{l^{(i)}(\mathbf{c}) : \mathbf{c} \in L\}$ constitute a prescribed *i-dimensional hitting distribution*. For each i-tuple $(p_1 \cdots p_i) \in X^{(i)}$, let $N^{(i)}(p_1 \cdots p_i) = |\{\mathbf{c} \in L : (p_1, \ldots, p_i) \in X^{(i)}(\mathbf{c})\}|$. A slightly modified Greedy Hitting Set is then used with parameters L and $\{l^{(i)}(\mathbf{c}) : \mathbf{c} \in L\}$.

We note that the extension to PSK constellations is straightforward. However, extensions to more *energy-efficient* signal constellations, e.g., 2^{2m+1}-CROSS constellations with $m \geq 1$ are *not* straightforward, and this is a topic of future research. The fact that a distance-preserving labeling does not exist for such constellations causes some immediate difficulties.

As a final remark we note that the general approach described here is applicable to other turbo-like BICM schemes and channels, e.g., serial concatenated convolutional codes and flat-fading channels.

References

[1] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inform. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes. 1," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Geneva, Switzer-

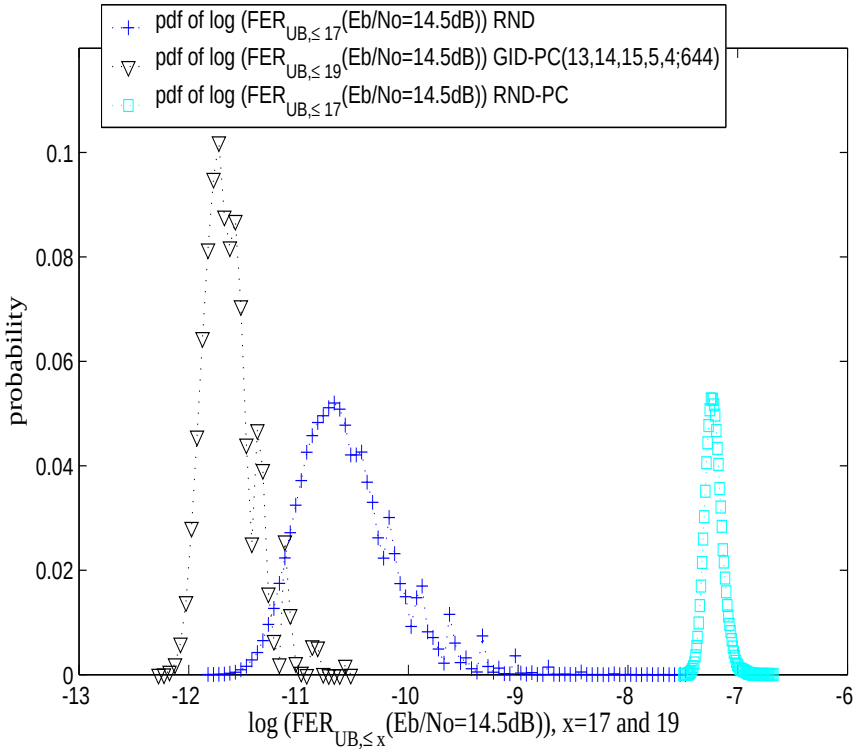

Fig. 4. Estimated probability density functions with $m = 4$ of $\log(\mathrm{FER}_{\mathrm{UB},\leq \mathrm{x}}(E_b/N_0 = 14.5 \text{ dB}))$, $x = 17$ and 19, from 500000 and 100000 bit-interleavers, respectively, generated independently according to the indicated algorithms with L composed of all codewords of weight ≤ 17.

land, May 1993, pp. 1064–1070.

[3] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[4] S. Y. Le Goff, "Signal constellations for bit-interleaved coded modulation," *IEEE Trans. Inform. Theory*, vol. 49, no. 1, pp. 307–313, Jan. 2003.

[5] R. D. Wesel, X. Liu, J. M. Cioffi, and C. Komninakis, "Constellation labeling for linear encoders," *IEEE Trans. Inform. Theory*, vol. 47, no. 6, pp. 2417–2431, Sept. 2001.

[6] L. C. Perez, J. Seghers, and D. J. Costello, Jr., "A distance spectrum interpretation of turbo codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 6, pp. 1698–1709, Nov. 1996.

[7] R. Garello, P. Pierleoni, and S. Benedetto, "Computing the free distance of turbo codes and serially concatenated codes with interleavers: Algorithms and applications," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 800–812, May 2001.

[8] E. Rosnes and Ø. Ytrehus, "Improved algorithms for the determination of turbo code weight distributions," 2004, *IEEE Trans. Commun.*, accepted for publication.

[9] ——, "An efficient algorithm for tailbiting turbo code weight distribution calculation," in *Proc. 3rd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, Sept. 2003, pp. 439–442.

[10] C. Berrou and S. Vaton, "Computing the minimum distances of linear codes by the error impulse method," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, Lausanne, Switzerland, July 2002, p. 5.

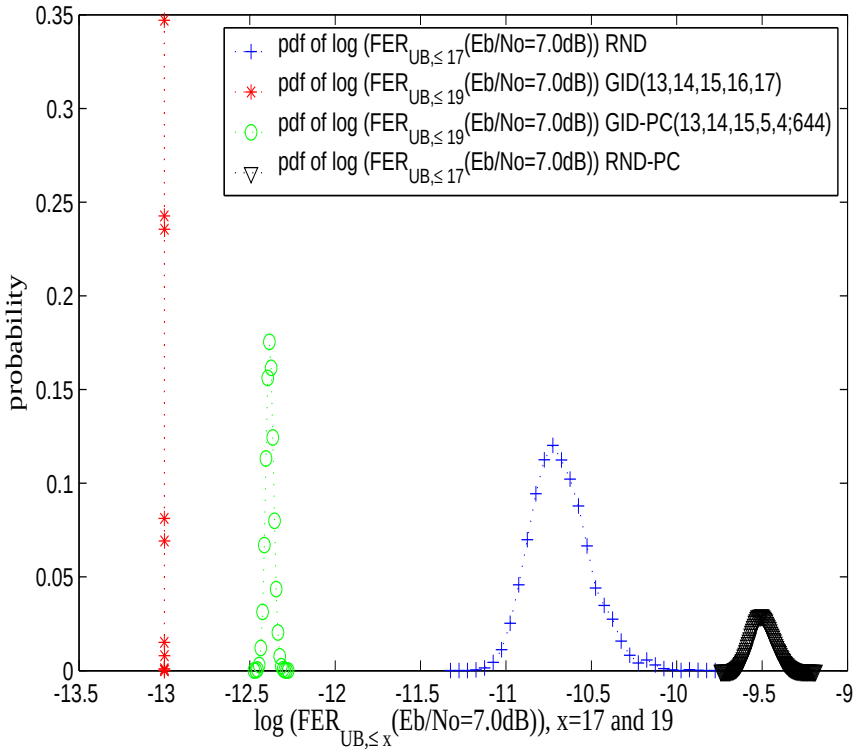

Fig. 5. Estimated probability density functions with $m = 2$ of $\log(\mathrm{FER}_{\mathrm{UB},\leq \mathrm{x}}(E_b/N_0 = 7.0 \text{ dB}))$, $x = 17$ and 19, from 500000 and 100000 bit-interleavers, respectively, generated independently according to the indicated algorithms with L composed of all codewords of weight ≤ 17.

[11] S. Riedel, "MAP decoding of convolutional codes using reciprocal dual codes," *IEEE Trans. Inform. Theory*, vol. 44, no. 3, pp. 1176–1187, May 1998.

[12] J. G. Proakis, *Digital Communications*. McGraw-Hill, 1995.

[13] G. Ausiello, A. D'Atri, and M. Protasi, "Structure preserving reductions among convex optimization problems," *J. Comput. Syst. Sci.*, vol. 21, no. 1, pp. 136–153, Aug. 1980.

[14] D. Divsalar, S. Dolinar, and F. Pollara, "Iterative turbo decoder analysis based on density evolution," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 891–907, May 2001.

[15] R. J. McEliece, "The algebraic theory of convolutional codes," in *Handbook of Coding Theory*, V. S. Pless and W. C. Huffman, Eds. North-Holland, Amsterdam: Elsevier, 1998, ch. 12, pp. 1065–1138.

[16] *Multiplexing and channel coding (FDD)*, 3rd Generation Partnership Project, June 1999, 3G TS 25.212.

[17] S. N. Crozier, "New high-spread high-distance interleavers for turbo-codes," in *Proc. 20th Biennial Symp. Commun.*, Kingston, Ontario, Canada, May 2000, pp. 3–7.

[18] K. S. Andrews, C. Heegard, and D. Kozen, "Interleaver design methods for turbo codes," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, Boston, MA, Aug. 1998, p. 420.

[19] M. Breiling, S. Peeters, and J. Huber, "Interleaver design using backtracking and spreading methods," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, Sorrento, Italy, June 2000, p. 451.

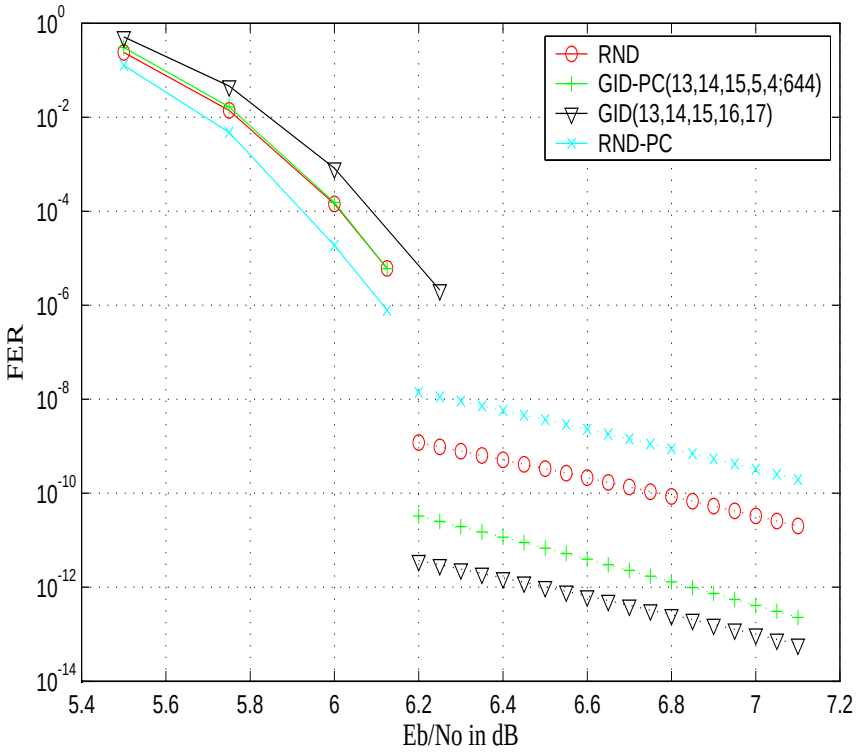

Fig. 6. Simulation results and truncated union bounds calculated from (4) using all codewords of weight ≤ 19 on the FER of BITCM with different bit-interleavers generated according to the algorithms in the legends with $m = 2$. The spectral efficiency is 3.10 b/s/Hz.

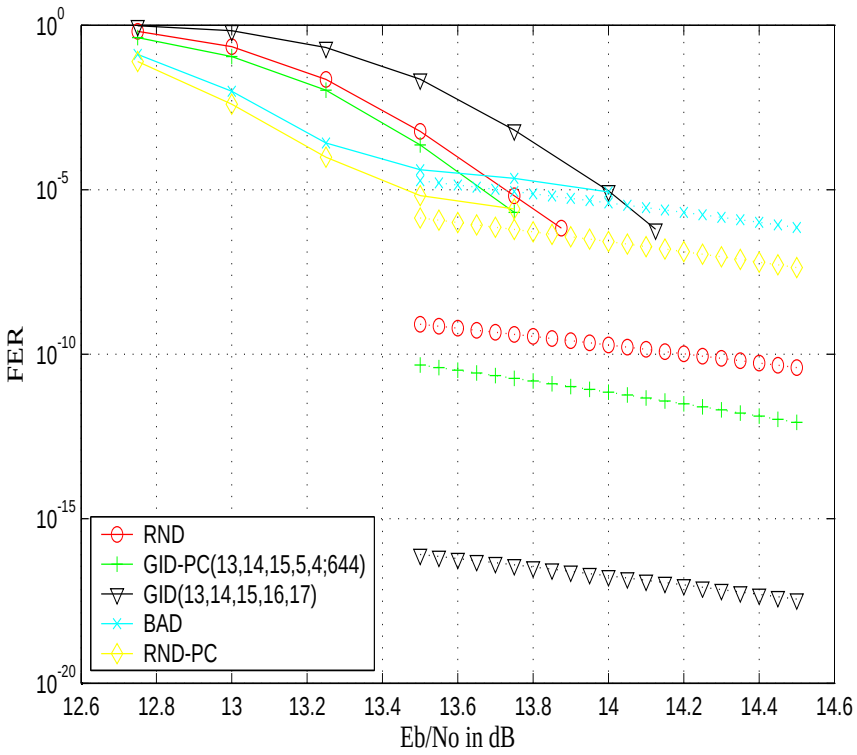

Fig. 7. Simulation results and truncated union bounds, calculated from (4) using all codewords of weight ≤ 19, on the FER of BITCM with different bit-interleavers generated according to the algorithms in the legends with $m = 4$. The spectral efficiency is 6.20 b/s/Hz.