Chalmers Publication Library

Pseudocodewords of linear programming decoding of 3-dimensional turbo codes

(article starts on next page)

Pseudocodewords of Linear Programming Decoding of 3-Dimensional Turbo Codes

Eirik Rosnes[†], Michael Helmling[‡], and Alexandre Graell i Amat[§]

[†]Department of Informatics, University of Bergen, N-5020 Bergen, Norway
Email: eirik@ii.uib.no
[‡]Department of Mathematics, University of Kaiserslautern, 67663 Kaiserslautern, Germany
Email: helmling@mathematik.uni-kl.de
[§]Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden
Email: alexandre.graell@chalmers.se

Abstract—In this work, we consider pseudocodewords of (relaxed) linear programming (LP) decoding of 3-dimensional turbo codes (3D-TCs), recently introduced by Berrou *et al.*. Here, we consider binary 3D-TCs while the original work of Berrou *et al.* considered double-binary codes. We present a relaxed LP decoder for 3D-TCs, which is an adaptation of the relaxed LP decoder for conventional turbo codes proposed by Feldman in his thesis. The vertices of this relaxed polytope are the pseudocodewords. We show that the support set of any pseudocodeword is a stopping set of iterative decoding of 3D-TCs using maximum *a posteriori* constituent decoders on the binary erasure channel. Furthermore, we present a numerical study of small block length 3D-TCs, which shows that typically the minimum pseudoweight (on the additive white Gaussian noise (AWGN) channel) is smaller than both the minimum distance and the stopping distance. In particular, we performed an exhaustive search over all interleaver pairs in the 3D-TC based on quadratic permutation polynomials over integer rings with a quadratic inverse. The search shows that the best minimum AWGN pseudoweight is significantly smaller than the best minimum distance.

I. INTRODUCTION

Turbo codes (TCs) have gained considerable attention since their introduction by Berrou *et al.* in 1993 [1] due to their near-capacity performance and low decoding complexity. The conventional TC is a parallel concatenation of two identical recursive systematic convolutional encoders separated by a pseudorandom interleaver. To improve the performance of TCs in the error floor region, hybrid concatenated codes (HCCs) can be used. In [2], a powerful HCC nicknamed 3-dimensional turbo code (3D-TC) was introduced. The coding scheme consists of a conventional turbo encoder and a *patch*, where a fraction λ of the parity bits from the turbo encoder are post-encoded by a third rate-1 encoder. The value of λ can be used to trade-off performance in the waterfall region with performance in the error floor region. As shown in [2], this coding scheme is able to provide very low error rates for a wide range of block lengths and code rates at the expense

of a small increase in decoding complexity with respect to conventional TCs. In a very recent work [3], an in-depth performance analysis of 3D-TCs was conducted. Stopping sets for 3D-TCs were treated in [4, 5].

Feldman *et al.* [6] recently considered the use of linear programming (LP) to decode binary linear codes. The obvious polytope for LP decoding is the convex hull of all codewords, in which case LP decoding is equivalent to maximum-likelihood (ML) decoding. However, in general, the convex hull has a description complexity that is exponential in the codeword length. Thus, Feldman *et al.* [6] proposed a relaxed polytope which contains all valid codewords as vertices, but also additional noncodeword vertices. The vertices of the relaxed polytope are basically what the authors called *pseudocodewords* in [6].

Relaxed LP decoding of conventional TCs was first described by Feldman in his thesis [7]. In [8], it was shown that the set of points from the relaxed polytope (as described by Feldman in [7]) where all entries are rational numbers is equal to the set of all pseudocodewords of all finite graph covers of the TC factor graph. Also, a many-to-one correspondence between pseudocodewords and stopping sets for turbo decoding on the binary erasure channel (BEC), as originally defined in [9], was observed. The many-to-one correspondence can be described as follows. The support set of any (graph-cover) pseudocodeword, i.e., the set of nonzero coordinates, is a stopping set, as defined in [9], and for any stopping set there is a pseudocodeword with support set equal to the stopping set.

Expressions for the *pseudoweight* on the BEC, the binary symmetric channel (BSC), and the additive white Gaussian noise (AWGN) channel are given in [10, 11]. The pseudoweight is the equivalent of the Hamming weight on the BEC and the BSC, and the equivalent of the squared Euclidean distance on the AWGN channel, when LP decoding and *not* ML decoding is performed. Computing the exact value of the minimum pseudoweight on any channel (possibly besides the BEC) is very hard, and the exact value is not known even for the well-known $(155, 64, 20)$ Tanner code from [12]. The best upper bound is 16.4037, found using the search algorithm from [13].
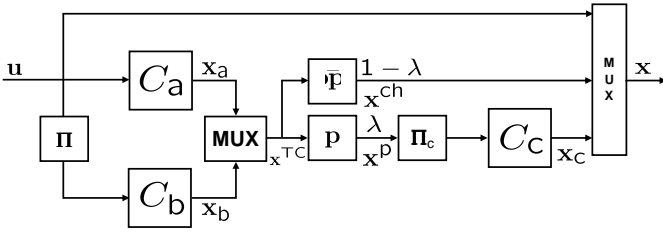
Fig. 1. 3D turbo encoder. A fraction λ of the parity bits from both constituent encoders C_a and C_b are grouped by a parallel/serial multiplexer, permuted by interleaver Π_c, and encoded by the rate-1 post-encoder C_c.

In this work, we adapt the definition of a pseudocodeword for a conventional TC from [8] to 3D-TCs. Furthermore, we show by several examples and by computer search that typically the minimum AWGN pseudoweight, denoted by $w_{\min}^{\text{AWGN}}$, is smaller than both the minimum distance $d_{\min}$ and the stopping distance $h_{\min}$ for these codes.

II. CODING SCHEME

A block diagram of the 3D-TC is depicted in Fig. 1. The information data sequence $\mathbf{u}$ of length K bits is encoded by a binary conventional turbo encoder. By a conventional turbo encoder we mean the parallel concatenation of two identical rate-1 recursive convolutional encoders, denoted by C_a and C_b, respectively. Here C_a and C_b are 8-state recursive convolutional encoders with generator polynomial $g(D) = (1 + D + D^3)/(1 + D^2 + D^3)$, i.e., the 8-state constituent encoder specified in the 3GPP/UMTS standard [14]. The code sequences of C_a and C_b are denoted by $\mathbf{x}_a$ and $\mathbf{x}_b$, respectively. We also denote by $\mathbf{x}^{\text{TC}}$ the codeword obtained by alternating bits from $\mathbf{x}_a$ and $\mathbf{x}_b$. A fraction λ ($0 \le \lambda \le 1$), called the *permeability rate*, of the parity bits from $\mathbf{x}^{\text{TC}}$ are permuted by interleaver Π_c (of length $N_c = 2\lambda K$), and encoded by an encoder of unity rate C_c, called the *patch* or the *post-encoder* [2]. This can be properly represented by a puncturing pattern $\mathbf{p}$ applied to $\mathbf{x}^{\text{TC}}$ (see Fig. 1) of period N_p containing λN_p ones (where a one means that the bit is not punctured). The fraction $1 - \lambda$ of parity bits which are not encoded by C_c is sent directly to the channel. Equivalently, this can be represented by a puncturing pattern $\bar{\mathbf{p}}$, the complement of $\mathbf{p}$. We denote by $\mathbf{x}_c$ the code sequence of C_c. Also, we denote by $\mathbf{x}_a^{\text{ch}}$ and $\mathbf{x}_b^{\text{ch}}$ the *sub-codewords* of $\mathbf{x}_a$ and $\mathbf{x}_b$, respectively, sent directly to the channel, and by $\mathbf{x}^{\text{ch}}$ the codeword obtained by alternating bits from $\mathbf{x}_a^{\text{ch}}$ and $\mathbf{x}_b^{\text{ch}}$. Likewise, we denote by $\mathbf{x}_a^{\text{p}}$ and $\mathbf{x}_b^{\text{p}}$ the *sub-codewords* of $\mathbf{x}_a$ and $\mathbf{x}_b$, respectively, encoded by C_c, and by $\mathbf{x}^{\text{p}}$ the codeword obtained by alternating bits from $\mathbf{x}_a^{\text{p}}$ and $\mathbf{x}_b^{\text{p}}$. Finally, the information sequence and the code sequences $\mathbf{x}^{\text{ch}}$ and $\mathbf{x}_c$ are multiplexed to form the code sequence $\mathbf{x}$, of length N bits, transmitted to the channel. Note that the overall nominal code rate of the 3D-TC is $R = K/N = 1/3$, the same as for the conventional TC without the patch. Higher code rates can be obtained either by puncturing $\mathbf{x}^{\text{ch}}$ or by puncturing the output of the patch, $\mathbf{x}_c$.

In [2], regular puncturing patterns of period $2/\lambda$ were considered for $\mathbf{p}$. For instance, if $\lambda = 1/4$, every fourth bit from each of the encoders of the outer TC are encoded by encoder C_c. The remaining bits are sent directly to the channel, and it follows that $\mathbf{p} = [11000000]$ and $\bar{\mathbf{p}} = [00111111]$.

III. STOPPING SETS FOR 3D-TCS

In this section, we review the definition of stopping sets for 3D-TCs from [4,5]. First, we need to define some concepts related to the support set of a vector and of a set of vectors.

The support set $\chi(\mathbf{x})$ of a binary vector $\mathbf{x} = (x_0, \ldots, x_{N-1})$ (of length N) is the set of nonzero coordinates. As an example, with $\mathbf{x} = (0, 1, 1, 0, 1)$, $\chi(\mathbf{x}) = \{1, 2, 4\}$. The support set $\chi(C)$ of a binary linear code C is the union of the support sets of each codeword in C, and the support vector of C, denoted by $\psi(C)$, is the unique binary vector whose support set is equal to $\chi(C)$. In the following, the notation $C^{-1}(\mathbf{c})$ with a binary vector $\mathbf{c}$ will denote the output of a maximum *a posteriori* decoder for code C when decoding $\mathbf{c}$ on the BEC where the 1-symbols in $\mathbf{c}$ are treated as erasures.

We start by reviewing the definition of a stopping set for a conventional TC, i.e., the 3D-TC without the patch. The following definition is taken from [9].

Definition 1 ([9])**.** *Let C denote a given conventional TC with interleaver Π. A subset $\mathcal{S} = \mathcal{S}(\Pi) = (\mathcal{S}_{\mathbf{u}}, \mathcal{S}_{\mathbf{x}_a}, \mathcal{S}_{\mathbf{x}_b}) \subseteq \{0, \ldots, N-1\}$ of the coordinates of the output sequence (or codeword), where $\mathcal{S}_{\mathbf{u}}$, $\mathcal{S}_{\mathbf{x}_a}$, and $\mathcal{S}_{\mathbf{x}_b}$ are subsets of the codeword coordinates corresponding to systematic bits, parity bits from C_a, and parity bits from C_b, respectively, is a stopping set if and only if the following conditions are satisfied.*

1) *There exist linear subcodes $\hat{C}_a \subseteq C_a$ and $\hat{C}_b \subseteq C_b$, such that $\mu_a(\chi(\hat{C}_a)) = \mathcal{S}_{\mathbf{x}_a}$ and $\mu_b(\chi(\hat{C}_b)) = \mathcal{S}_{\mathbf{x}_b}$,*
2) $\Pi(\chi(C_a^{-1}(\psi(\hat{C}_a)))) = \chi(C_b^{-1}(\psi(\hat{C}_b)))$, *and*
3) $\mu_s(\chi(C_a^{-1}(\psi(\hat{C}_a)))) = \mathcal{S}_{\mathbf{u}}$.

$\mu_x(\cdot)$ *is the mapping of indices of parity bits from C_x to indices of the turbo codeword, $x = a, b$, and $\mu_s(\cdot)$ is the mapping of indices of systematic bits to indices of the turbo codeword.*

The definition above can be generalized to 3D-TCs in a fairly straightforward manner. The following definition is taken from [5].

Definition 2 ([5])**.** *Let C denote a given 3D-TC with interleavers Π and Π_c. A set $\mathcal{S} = \mathcal{S}(\Pi, \Pi_c) = (\mathcal{S}_{\mathbf{u}}, \mathcal{S}_{\mathbf{x}_a^{\text{ch}}}, \mathcal{S}_{\mathbf{x}_b^{\text{ch}}}, \mathcal{S}_{\mathbf{x}_c}) \subseteq \{0, \ldots, N-1\}$ of the coordinates of the output sequence (or codeword), where $\mathcal{S}_{\mathbf{u}}$, $\mathcal{S}_{\mathbf{x}_a^{\text{ch}}}$, $\mathcal{S}_{\mathbf{x}_b^{\text{ch}}}$, and $\mathcal{S}_{\mathbf{x}_c}$ are subsets of the codeword coordinates corresponding to systematic bits, parity bits from C_a sent directly to the channel, parity bits from C_b sent directly to the channel, and parity bits from C_c, respectively, is a stopping set if and only if the following conditions are satisfied.*

1) *There exists a linear subcode $\hat{C}_c \subseteq C_c$, such that $\mu_c(\chi(\hat{C}_c)) = \mathcal{S}_{\mathbf{x}_c}$, and*
2) $(\mathcal{S}_{\mathbf{u}}, \mathcal{S}_{\mathbf{x}_a^{\text{p}}} \cup \mathcal{S}_{\mathbf{x}_a^{\text{ch}}}, \mathcal{S}_{\mathbf{x}_b^{\text{p}}} \cup \mathcal{S}_{\mathbf{x}_b^{\text{ch}}})$ *is a stopping set of the outer TC, as defined in Definition 1, where $\mathcal{S}_{\mathbf{x}_x^{\text{p}}}$ is the*

set of indices from $\Pi_{\rm c}^{-1}(\chi(C_{\rm c}^{-1}(\psi(\hat{C}_{\rm c}))))$ *corresponding to parity bits from $C_{\rm x}$, $\rm x = a, b$.*

$\mu_{\rm c}(\cdot)$ *is the mapping of indices of bits from the output of the inner encoder to indices of codewords of the 3D-TC.*

The size *of a stopping set $\mathcal{S}$ is its cardinality, and the size of the smallest nonempty stopping set is the stopping distance $h_{\min}$.*

IV. LP Decoding of 3D-TCs

In this section, we consider relaxed LP decoding of 3D-TCs, adapting the relaxation proposed in [7] for conventional TCs to 3D-TCs.

Let $T_{\rm x} = T_{\rm x}(V_{\rm x}, E_{\rm x})$ denote the *information bit-oriented trellis* of $C_{\rm x}$, $\rm x = a, b, c$, where the vertex set $V_{\rm x}$ partitions as $V_{\rm x} = \cup_{t=0}^{I_{\rm x}} V_{{\rm x},t}$, which also induces the partition $E_{\rm x} = \cup_{t=0}^{I_{\rm x}-1} E_{{\rm x},t}$ of the edge set $E_{\rm x}$, where $I_{\rm x}$ is the trellis length of $T_{\rm x}$. In the following, the encoders $C_{\rm x}$ (and their corresponding trellises $T_{\rm x}$) are assumed (with some abuse of notation) to be systematic, in the sense that the output bits are prefixed with the input bits. Thus, $C_{\rm x}$ is regarded as a rate-$1/2$ encoder, and the trellis $T_{\rm x}$ has an output label containing two bits, for $\rm x = a, b, c$. Now, let $e \in E_{{\rm x},t}$ be an arbitrary edge from the tth trellis section. The ith bit in the output label of e is denoted by $c_i(e)$, $i = 0, \ldots, n_{{\rm x},t} - 1$, the starting state of e as $s^S(e)$, and the ending state of e as $s^E(e)$. Define the polytope

$$
\mathcal{Q}_{\rm x} = \Big\{ (\tilde{\mathbf{y}}, \mathbf{f}) \in [0,1]^{N+2\lambda K} \times [0,1]^{|E_{\rm x}|} :
$$
$$
\sum_{e \in E_{\rm x}: s^S(e)=v} f_e = \sum_{e \in E_{\rm x}: s^E(e)=v} f_e, \ \forall v \in V_{{\rm x},t},
$$
$$
t = 1, \ldots, I_{\rm x} - 1, \ \sum_{e \in E_{{\rm x},0}} f_e = 1, \ \text{and}
$$
$$
\tilde{y}_{\rho_{\rm x}(\phi_{\rm x}(t,i))} = \sum_{e \in E_{{\rm x},t}: c_i(e)=1} f_e, \ t = 0, \ldots, I_{\rm x} - 1,
$$
$$
i = 0, \ldots, n_{{\rm x},t} - 1 \Big\} \tag{1}
$$

where $\phi_{\rm x}(t,i) = \sum_{j=0}^{t-1} n_{{\rm x},j} + i$, and $\rho_{\rm x}(\cdot)$ denotes the mapping of codeword indices of constituent encoder $C_{\rm x}$ to codeword indices of the overall codeword of the 3D-TC appended with the $2\lambda K$ parity bits from encoders $C_{\rm a}$ and $C_{\rm b}$ which are sent to the patch.

Next, define the following projection onto the first N variables of $\tilde{\mathbf{y}}$

$$
\dot{\mathcal{Q}}_{\rm x} = \Big\{ \mathbf{y} \in [0,1]^N : \exists\, \mathbf{f} \in [0,1]^{|E_{\rm x}|}, \hat{\mathbf{y}} \in [0,1]^{2\lambda K}, \ \text{and}
$$
$$
((\mathbf{y}, \hat{\mathbf{y}}), \mathbf{f}) \in \mathcal{Q}_{\rm x} \Big\}
$$

and the polytope

$$
\dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}} = \dot{\mathcal{Q}}_{\rm a} \cap \dot{\mathcal{Q}}_{\rm b} \cap \dot{\mathcal{Q}}_{\rm c}.
$$

Relaxed LP decoding (on a binary-input memoryless channel) of 3D-TCs can be described by the linear program

$$
\text{minimize} \ \sum_{l=0}^{N-1} \lambda_l y_l \ \text{subject to } \mathbf{y} \in \dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}} \tag{2}
$$

where

$$
\lambda_l = \log\left(\frac{\Pr\{r_l | c_l = 0\}}{\Pr\{r_l | c_l = 1\}} \right), \ l = 0, \ldots, N-1,
$$

c_l is the lth codeword bit, and r_l is the lth component of the received vector. If instead of $\dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}}$ we use the convex hull of the codewords of the 3D-TC, then solving the linear program in (2) is equivalent to ML decoding.

The notion of a *proper* and C-*symmetric* polytope was introduced in [7, Ch. 4] where the author proved that the probability of error of LP decoding is independent of the transmitted codeword on a binary-input output-symmetric memoryless channel when the underlying code is linear and the polytope is proper and C-symmetric.

Lemma 1. *The polytope $\dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}}$ is proper, i.e., $\dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}} \cap \{0,1\}^N = C$ and C-symmetric, i.e., for any $\mathbf{y} \in \dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}}$ and $\mathbf{c} \in C$ it holds that $|\mathbf{y} - \mathbf{c}| \in \dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}}$.*

Lemma 1 can be proved using the same arguments as for a conventional TC. The formal proof is omitted for brevity.

When decoding 3D-TCs by solving the linear program in (2), a vertex point of the polytope $\dot{\mathcal{Q}}_{\Pi,\Pi_{\rm c}}$ is always returned by the decoder. Furthermore, any vertex point is a pseudocodeword, and the pseudoweight on the AWGN channel of a nonzero pseudocodeword $\boldsymbol{\omega}$ is defined as [10, 11]

$$
w^{\rm AWGN}(\boldsymbol{\omega}) = \frac{\left(\sum_{l=0}^{N-1} \omega_l \right)^2}{\sum_{l=0}^{N-1} \omega_l^2}.
$$

We can prove the following lemma.

Lemma 2. *Let C denote a given 3D-TC with interleavers Π and $\Pi_{\rm c}$. For any pseudocodeword $\boldsymbol{\omega}$, the support set $\chi(\boldsymbol{\omega})$ of $\boldsymbol{\omega}$ is a stopping set according to Definition 2. Conversely, for any stopping set $\mathcal{S} = \mathcal{S}(\Pi, \Pi_{\rm c})$ of the 3D-TC there exists a pseudocodeword $\boldsymbol{\omega}$ with support set $\chi(\boldsymbol{\omega}) = \mathcal{S}$.*

As a consequence of Lemma 2, it follows that $w_{\min}^{\rm AWGN}$ of C is upper-bounded by the $h_{\min}$ of C.

We remark that the $d_{\min}$ can be computed exactly by solving the (integer) program in (2) with $\lambda_l = 1$ for all l, with integer constraints on all the flow variables $\mathbf{f}$ in (1), i.e., $f_l \in \{0, 1\}$ for all l, and with the constraint $\sum_{l=0}^{N-1} y_l \geq 1$ to avoid obtaining the all-zero codeword. The exact $d_{\min}$ of 3D-TCs has not been computed before in the literature. For instance, in [3], only estimates of the $d_{\min}$ were provided. Finally, note that the exact $h_{\min}$ can be computed in a similar manner using *extended trellis modules* for $T_{\rm x}$ (see [9] for details).

V. Numerical Results

In this section, we present some numerical results when the interleaver pair $(\Pi, \Pi_{\rm c})$ is taken from the set of all possible interleaver pairs, and when it is taken from the set of pairs of quadratic permutation polynomials (QPPs) over integer rings. Permutation polynomial based interleavers over integer rings for conventional TCs were first proposed by Sun and Takeshita in [15]. These interleavers are fully algebraic
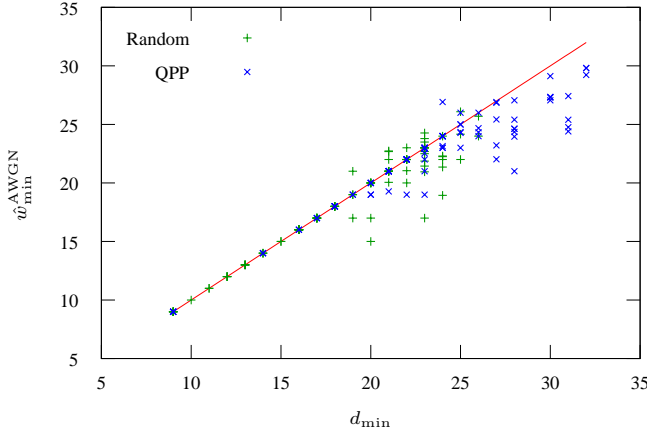
Fig. 2. *Estimated $w_{\min}^{\text{AWGN}}$, using the algorithm from [13], adapted to 3D-TCs, and exact $d_{\min}$ for 3D-TCs with 100 randomly selected pairs of interleavers (green plus signs) and with 100 randomly selected pairs of QPP-based interleavers (blue x-marks). $K = 128$ and $R = 1/3$.*

Fig. 3. *Exact $d_{\min}$ (red circles), exact $h_{\min}$ (green x-marks), and $\hat{w}_{\min}^{\text{AWGN}}$ (blue plus signs) of the 75 best (in terms of $\hat{w}_{\min}^{\text{AWGN}}$) QPP-based interleaver pairs for the 3D-TC with input block length $K = 128$ and code rate $R = 1/3$.*

and *maximum contention-free* [16], which makes them very suitable for parallel implementation in the turbo decoder. QPP-based interleavers for conventional TCs were recently adopted for the 3GPP LTE standard [17]. Finally, we remark that for all the results below, $\lambda = 1/4$ and the regular puncturing pattern $\mathbf{p} = [11000000]$ are assumed. As shown in [3], $\lambda = 1/4$ gives a suitable trade-off between performance in the waterfall and error floor regions.

A. Ensemble-Average Results for $K = 128$ and $R = 1/3$

In Fig. 2, we present the exact $d_{\min}$ and an estimate of $w_{\min}^{\text{AWGN}}$, denoted by $\hat{w}_{\min}^{\text{AWGN}}$, of unpunctured 3D-TCs with $K = 128$ and with 100 randomly selected pairs of interleavers (Π, Π_c) (green plus signs). The corresponding results with QPP-based interleaver pairs (and with no constraints on the inverse polynomials) are also displayed (blue x-marks). For all codes, except 11, the estimated $w_{\min}^{\text{AWGN}}$ is at most equal to the $d_{\min}$. The minimum AWGN pseudoweight $w_{\min}^{\text{AWGN}}$ was estimated using the algorithm from [13], adapted to 3D-TCs, with an SNR of 2.0 dB and 500 evaluations of the algorithm, while the $d_{\min}$ was computed exactly by solving (2), augmented with the constraint $\sum_{l=0}^{N-1} y_l \geq 1$ (to avoid obtaining the all-zero codeword), with integer constraints on the flow variables $\mathbf{f}$ in (1). Note that when the $d_{\min}$ is strictly smaller than the estimated $w_{\min}^{\text{AWGN}}$, the estimation algorithm from [13] was unsuccessful, since $w_{\min}^{\text{AWGN}}$ is upper-bounded by the $d_{\min}$ (see Lemma 2). From the figure, we observe that QPPs give in general better codes, and that $w_{\min}^{\text{AWGN}}$ is strictly lower than the $d_{\min}$ for most codes when the $d_{\min}$ is large.

B. Exhaustive/Random Search Optimizing $w_{\min}^{\text{AWGN}}$

In this subsection, we present the results of a computer search for pairs of QPPs with a quadratic inverse for $K = 128$ and 256 for unpunctured $R = 1/3$ 3D-TCs. The objective of the search was to find pairs of QPPs giving a large $w_{\min}^{\text{AWGN}}$, estimated using the algorithm from [13], adapted to 3D-TCs, with an SNR of 1.7 dB and 500 evaluations of
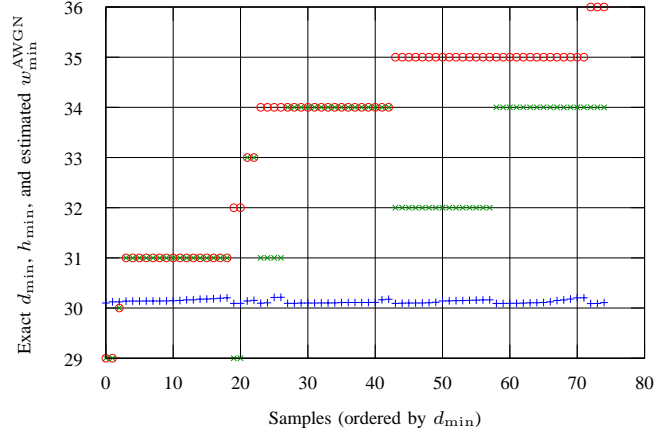
the algorithm. To speed up the search, an adaptive threshold on the minimum AWGN pseudoweight $w_{\min}^{\text{AWGN}}$ was set in the search, in the sense that if a pseudocodeword of AWGN pseudoweight smaller than the threshold was found, then this particular candidate pair of QPPs was rejected.

For $K = 128$, we performed an exhaustive search over the entire class of pairs of QPPs (with a quadratic inverse). In Fig. 3, we plot the exact $d_{\min}$ (red circles), the exact $h_{\min}$ (green x-marks), and $\hat{w}_{\min}^{\text{AWGN}}$ (blue plus signs) of the 75 3D-TCs with the best $\hat{w}_{\min}^{\text{AWGN}}$. For each point in the figure, the x-coordinate is the sample index (the results are ordered by increasing $d_{\min}$), while the y-coordinate is either the exact $d_{\min}$, the exact $h_{\min}$, or $\hat{w}_{\min}^{\text{AWGN}}$. From the figure, we observe that the best $w_{\min}^{\text{AWGN}}$ (which is at most 30.2139), is strictly smaller than the best possible $d_{\min}/h_{\min}$. The best possible $d_{\min}$ was established to be 38 (exhaustive search), and for this particular code the exact $h_{\min}$ is 36.

For $K = 256$, only a partial search has been conducted. The largest value for $\hat{w}_{\min}^{\text{AWGN}}$ found in the search is 42.9650 after taking approximately 100000 samples from the search space.

C. Exhaustive/Random Search Optimizing $d_{\min}$

We also performed an exhaustive/random search for pairs of QPPs with a quadratic inverse for selected values of K for unpunctured $R = 1/3$ 3D-TCs. For $K = 128$, 160, 192, and 208, the search was exhaustive, in the sense that each pair of interleavers was looked at. The objective function in the search was the $d_{\min}$, which was estimated using the triple impulse method [18]. The results are given in Table I for selected values of K, where $f(x) = f_1 x + f_2 x^2 \pmod{K}$ generates the TC interleaver, $\tilde{f}(x) = \tilde{f}_1 x + \tilde{f}_2 x^2 \pmod{N_c}$ generates the permutation in the patch, and $\hat{d}_{\min}$, $d_{\min}^{\text{LB}}$, and $\hat{w}_{\min}^{\text{AWGN}}$ denote the estimated $d_{\min}$, a lower bound on the $d_{\min}$ based on an ensemble analysis, and the estimated $w_{\min}^{\text{AWGN}}$, respectively. The minimum AWGN pseudoweight $w_{\min}^{\text{AWGN}}$ was estimated using the algorithm from [13], adapted to 3D-TCs, with SNR and number of evaluations of the algorithm given in the tenth and the eleventh column of the table, respectively.

TABLE I
RESULTS FROM AN EXHAUSTIVE/RANDOM SEARCH FOR PAIRS OF QPPs WITH $\lambda = 1/4$, BOTH WITH A QUADRATIC INVERSE, IN WHICH THE FIRST QPP $f(x)$ GENERATES THE TC INTERLEAVER AND THE SECOND QPP $\tilde{f}(x)$ GENERATES THE PERMUTATION IN THE PATCH.

K	f_1	f_2	N_c	$\tilde{f}_1$	$\tilde{f}_2$	$\hat{d}_{\min}$	$d_{\min}^{\mathrm{LB}}$	$\hat{w}_{\min}^{\mathrm{AWGN}}$	SNR	Evaluations
128 [a]	55	96	64	9	16	38 [b]	19	29.6042	2.0 dB	2000
160 [a]	131	60	80	9	20	42 [b]	20	30.0000	1.7 dB	500
192 [a]	35	24	96	11	12	46	22	32.9046	1.7 dB	500
208 [a]	165	182	104	37	26	49	22	36.3370	1.7 dB	500
256	239	192	128	37	32	52	24	42.7816	1.7 dB	500
320	183	280	160	57	20	58	26	41.3818	1.7 dB	500
512 [c]	175	192	256	15	192	67	33	45.5872	2.0 dB	500

[a] Exhaustive search.
[b] This is the exact $d_{\min}$.
[c] The QPPs are taken from [3].

Note that the number of evaluations of the algorithm indicates its reliability. The lower bound on the $d_{\min}$ given in the eight column of the table should be interpreted in the following manner. When picking random pairs of interleavers from the 3D-TC ensemble, the $d_{\min}$ will be at least equal to the value of the bound with probability $1/2$. Finally, we remark that the codes in the first and second rows, for $K = 128$ and 160, are $d_{\min}$-optimal, in the sense that there does not exist any pair of QPPs (with a quadratic inverse) giving a $d_{\min}$ strictly larger than 38 and 42, respectively, for the unpunctured 3D-TC.

D. $K = 512$ and Different Code Rates

In [3], a pair of QPPs for $K = 512$ giving an estimated $d_{\min}$ of 67 for $R = 1/3$ was tabulated in Table IV. Also, the $d_{\min}$ with optimized puncturing patterns, giving estimated minimum distances of 38, 17, and 15, for nominal code rates $R = 1/2$, $2/3$, and $4/5$, respectively, was tabulated. By using the algorithm from [13], adapted to 3D-TCs, we found pseudocodewords of AWGN pseudoweights 45.5872, 28.8257, 19.1668, and 10.1984, for nominal code rates $1/3$, $1/2$, $2/3$, and $4/5$, respectively.

VI. CONCLUSION

In this work, we considered pseudocodewords of (relaxed) LP decoding of 3D-TCs, generalizing the LP relaxation proposed by Feldman in his thesis for conventional TCs. We showed that the support set of any pseudocodeword is a stopping set, and presented a numerical study of small block length 3D-TCs, which showed that typically $w_{\min}^{\mathrm{AWGN}}$ is smaller than both the $d_{\min}$ and the $h_{\min}$ for these codes. It is expected that the $w_{\min}^{\mathrm{AWGN}}$ will dominate performance for high SNRs.

REFERENCES

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Geneva, Switzerland, May 1993, pp. 1064–1070.

[2] C. Berrou, A. Graell i Amat, Y. Ould-Cheikh-Mouhamedou, and Y. Saouter, "Improving the distance properties of turbo codes using a third component code: 3D turbo codes," *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2505–2509, Sep. 2009.

[3] E. Rosnes and A. Graell i Amat, "Performance analysis of 3-dimensional turbo codes," *IEEE Trans. Inf. Theory*, to appear.

[4] A. Graell i Amat and E. Rosnes, "Stopping set analysis of 3-dimensional turbo code ensembles," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Austin, TX, Jun. 2010, pp. 2013–2017.

[5] ——, "Analysis of 3-dimensional turbo code ensembles on the binary erasure channel," *IEEE Trans. Commun.*, submitted for publication.

[6] J. Feldman, M. J. Wainwright, and D. R. Karger, "Using linear programming to decode binary linear codes," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 954–972, Mar. 2005.

[7] J. Feldman, "Decoding error-correcting codes via linear programming," Ph.D. dissertation, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (MIT), Cambridge, MA, 2003.

[8] E. Rosnes, "On the connection between finite graph covers, pseudo-codewords, and linear programming decoding of turbo codes," in *Proc. 4th Int. Symp. Turbo Codes & Rel. Topics*, Munich, Germany, Apr. 2006.

[9] E. Rosnes and Ø. Ytrehus, "Turbo decoding on the binary erasure channel: Finite-length analysis and turbo stopping sets," *IEEE Trans. Inf. Theory*, vol. 53, no. 11, pp. 4059–4075, Nov. 2007.

[10] P. O. Vontobel and R. Koetter, "Graph-cover decoding and finite-length analysis of message-passing iterative decoding of LDPC codes," *IEEE Trans. Inf. Theory*, to appear. [Online]. Available: http://arxiv.org/abs/cs.IT/0512078/

[11] G. D. Forney, Jr., R. Koetter, F. R. Kschischang, and A. Reznik, "On the effective weights of pseudocodewords for codes defined on graphs with cycles," in *Codes, Systems, and Graphical Models*, B. Marcus and J. Rosenthal, Eds., vol. 123 of IMA Vol. Math. Appl. Springer Verlag, 2001, pp. 101–112.

[12] R. M. Tanner, D. Sridhara, and T. Fuja, "A class of group-structured LDPC codes," in *Proc. Int. Symp. Commun. Theory and Appl. (ISCTA)*, Ambleside, England, Jul. 2001.

[13] M. Chertkov and M. G. Stepanov, "An efficient pseudocodeword search algorithm for linear programming decoding of LDPC codes," *IEEE Trans. Inf. Theory*, vol. 54, no. 4, pp. 1514–1520, Apr. 2008.

[14] *Multiplexing and Channel Coding (FDD)*, 3rd Generation Partnership Project, Jun. 1999, 3G TS 25.212.

[15] J. Sun and O. Y. Takeshita, "Interleavers for turbo codes using permutation polynomials over integer rings," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 101–119, Jan. 2005.

[16] O. Y. Takeshita, "On maximum contention-free interleavers and permutation polynomials over integer rings," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1249–1253, Mar. 2006.

[17] "3rd generation partnership project; technical specification group radio access network; evolved universal terrestrial radio access (E-UTRA); multiplexing and channel coding (release 8)," Dec. 2008, 3GPP TS 36.212 v8.5.0.

[18] S. Crozier, P. Guinand, and A. Hunt, "Computing the minimum distance of turbo-codes using iterative decoding techniques," in *Proc. 22th Biennial Symp. Commun.*, Kingston, ON, Canada, May/Jun. 2004, pp. 306–308.